Safely Doubling your Block Ciphers for a Post-Quantum World *

Ritam Bhaumik[1], André Chailloux[1], Paul Frixons[1,2] and María Naya-Plasencia[1]

[1] Inria, Paris
[2] Orange Labs, Paris
ritam.bhaumik@epfl.ch andre.chailloux@inria.fr
paul.frixons@inria.fr maria.naya_plasencia@inria.fr

Abstract. In order to maintain a similar security level in a post-quantum setting, many symmetric primitives should have to double their keys and increase their state sizes. So far, no generic way for doing this is known that would provide convincing quantum security guarantees. In this paper we propose a new generic construction that allows to double the key and the state size of a block cipher. For this we have modified the ECB-Mix-ECB (EME) construction, as we have been able to mount a new type of superposition attack on EME, and we provide several classical and quantum security arguments and analyses for our new construction QuEME. We propose a concrete instantiation of this construction with variants of AES-128.

Keywords: block cipher, post-quantum security, superposition attacks, security proofs, AES-128, generic construction.

1 Introduction

For a long time, it was accepted that symmetric primitives only needed to double their key length in order to stay resistant to quantum attackers. Though new attacks in powerful models [KM12,KM10,KLLN16a] have shown that a more in-depth study is needed and that some particular scenarios are dangerous, most current symmetric primitives have best quantum attacks that achieve at most a square-root speed-up on the classical attack, showing that most attacks would indeed be infeasible with a double-sized key. Nevertheless, no generic, simple, and efficient way is known for doubling the key size of a primitive, and the best known candidate for this purpose—the FX construction [KR01]—was proven to be insecure with respect to quantum attacks in the superposition model [LM17], though it has been shown to fare better in weaker models [JST21]. Other key-extension modes like the two-key Even Mansour [ABKM22] could also be shown secure only in weaker models.

* This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement no. 714294 - acronym QUASYModo).

Additionally, in [CNS17], it was pointed out that attacks on modes exploiting internal collisions, that depend on the internal size of the primitives, might also render the primitives weaker against quantum adversaries, so that just increasing the key length might not be enough, and the internal size of the primitive should also be increased. A new post-quantum symmetric family of primitives—Saturnin [CDL+20]—was proposed to address this concern. The block-cipher which forms the core of this family has a state-size of 256 bits, allowing much more reasonable security claims regarding all types of quantum attacks.

A generic and provable way of extending secure classical constructions into new ones with doubled key as well as doubled state-size is still an interesting question that has been widely studied without a satisfactory answer being found so far. In [HI19] the authors show a way of proving that the 4-round Luby-Rackoff construction (LR4) is a qPRP. However, in order to build secure post-quantum constructions, we need also to take into account the decryption direction. In [IHM+19] it was shown that LR4 has a quantum attack when we allow both encryption and decryption queries, so a natural candidate for extending this attack would be LR5. Unfortunately, with the proof techniques available at present, proving the quantum security of LR5 is turning out to be very challenging. Using the same database technique as in the proof of [HI19] is not possible since there is no known way yet of generalising database-oracles to permutations, and the equations governing the internal variables are quite complex with many variables, making ad-hoc techniques difficult to apply.

The aim of our work is to provide new useful information in order to advance towards building a generic and provable way of easily extending secure classical constructions into new ones with doubled key and doubled state-size, while their post-quantum security remains comparable to the original classical one. For this, we consider a new approach, based on the Encrypt-Mix-Encrypt paradigm with five block-cipher calls, that would allow to extend the internal state as well as the key size of any classically secure block-cipher.

The ECB-Mix-ECB or EME construction [HR04] was proposed as a highly parallelisable mode of operation to extend the domain of a block-cipher to arbitrary lengths. The ECB layers above and below make it a suitable candidate for resisting quantum attacks, since most Simon-like attacks rely on some part of the input passing through only one block-cipher call or being XOR-ed directly to the state, and the ECB layers ensure that every part of the input passes through at least two block-cipher calls during both the encryption and decryption routines.

1.1 Our Contributions

The main contributions of our work are:

1. An original quantum superposition attack on the EME construction. It introduces a new family of attacks that exhibits a periodic property found in collisions;
2. A new construction based on Encrypt-Mix-Encrypt that resists this attack, by replacing the XOR-then-Encrypt mixing layer with a tweakable permu-

tation call, that we will call QuEME, and that is claimed to offer n bits of security in both the classical and the quantum setting;
3. Four classical proofs for this construction: an IND-CPA proof of security up to the width of the underlying block-cipher using mirror theory; an IND-CCA proof up to the same bound using a conjectured tighter version of mirror theory, for which we find numerical evidence through an original approach using some simulations; and direct proofs of IND-CPA and IND-CPA security up to 2/3-rd the width of the block-cipher;
4. An original distinguisher that applies in particular to our construction that runs in time $O(2^n)$, which matches our security bound;
5. The first quantum security arguments for Encrypt-Mix-Encrypt constructions, to show that QuEME has, at least, $n/6$ bits of quantum security, which is in par for instance with the quantum security of LR4 [HI19], and show in particular that there is no collapse in the quantum security as can happen in LR3 for instance [KM10];
6. A concrete instantiation of the QuEME scheme with AES-128 for building a block-cipher with 256-bit state and key, with concrete (quantum) security claims; and some variants with fewer rounds than 10 for the building block that we also believe should be resistant.

The paper is organised as follows: as there are several results of different flavours in this paper, we start Section 2 by providing an inclusive summary of the main conclusions and final results obtained in the paper. Next, in Section 3, the EME construction is introduced, as well as a quantum attack on it. Section 4 proposes our new construction, QuEME, and proves its classical security. Section 5 describes the simulation we have implemented in order to validate the conjecture from the previous section, in a quite innovative approach. Section 6 presents our new distinguisher on $O(2^n)$ that matches our bound. Section 7 provides some quantum arguments to support the quantum security of the construction. Section 8 provides variants of a concrete construction, Double-AES, combining QuEME with AES-128. A conclusion and discussion are provided in Section 9.

2 Summary of the Results

Our goal in this paper is to design a quantum-safe mode for doubling the width and key-length of a block-cipher. To the best of our knowledge, such a design has previously not been proposed. We propose a construction with five block-cipher calls that achieves good parallelisability, good classical security bounds, and some post-quantum security guarantees. Some side results of our work are a new type of superposition attack, that applies to the EME construction, an original distinguisher on our construction, matching our bound, and an new kind of approach using simulation for supporting the conjecture in the proof. In addition we propose a concrete instantiation using our construction combined with the AES block-cipher. In this section we provide a summary of each of the presented results in this paper.

2.1 First Attempt and Attack

The main aim is to propose a construction that would double the internal state and key, and simultaneously we considered building a parallelisable construction. Therefore our first idea was to begin with a two-block version of EME (ECB-Mix-ECB) [HR04], shown in Fig. 1, which is known to be classically secure to the birthday bound in the width of the underlying block-cipher:

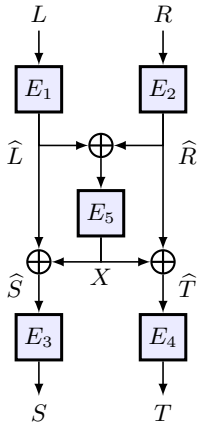


Fig. 1: The EME construction [HR04].

$$\mathsf{EME}(L,R) := E_3(E_1(L)\oplus E_5(E_1(L)\oplus E_2(R))), E_4(E_2(R)\oplus E_5(E_1(L)\oplus E_2(R))).$$

However, we show that this does not work, as we managed to build a new superposition key-recovery attack against this construction. This attack is one of its own kind as it is the first (to our knowledge) to combine collision search and Simon's algorithm. Indeed, our attack uses the uniform superposition of collisions to restrict a simple function, exposing a period property only for the correct key.

2.2 New Construction

We modify the EME construction by introducing a tweakable permutation call in the mixing layer, which can be implemented with a block-cipher by inserting the tweak as key. This we call QuEME, shown in Fig. 2.

2.3 Classical Security Proofs

We present several classical security proofs for the paper. When we use a well-accepted result of mirror theory, we can show IND-CPA security up to n bits. If

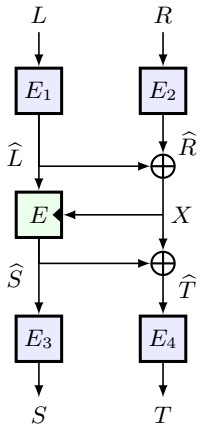


Fig. 2: The QuEME construction.

we assume a more general conjectured result of mirror theory, we can also show IND-CCA security up to n bits. Using an innovative approach, we also perform a simulation using small values of n, in order to obtain numerical evidence supporting both results of mirror theory that we use. Finally, without mirror theory, we can show security up to $2n/3$ bits in both IND-CPA and IND-CCA settings. Our results improve upon previously known security results of EME, which were only up to the birthday bound in the width of E. In support of our conjectured result from Mirror Theory, we show some evidence obtained from simulations, which to the best of our knowledge is a novel use of simulations in reduction proofs of this kind.

2.4 Distinguisher Matching the bound

We did not find any classical or quantum attack against the new mode of complexity lower than n bits, where n is the width of the underlying block-cipher. Even for less than $2n$ bits, mounting a key-recovery or message-recovery attack seems to be difficult. However, we found an information-theoretic distinguisher in 2^n queries which works by exhausting the entropy of the keyed block-cipher calls and solving the resulting equations. Our distinguisher uses the fact that any query $(x_1||x_2)$ done to our construction gives output $(y_1||y_2)$ satisfying

$$E_1(x_1) \oplus E_2(x_2) = E_3^{-1}(y_1) \oplus E_4^{-1}(y_2)$$

for some random permutations E_1, E_2, E_3, E_4. (Since this relation also holds in EME and other similar constructions, our attack also works there.) What we show using linear algebra arguments is that if we query a random permutation with $\Omega(2^n)$ queries then permutations E_1, E_2, E_3, E_4 st. the above equation is compatible with our queries don't exist anymore; which allows us to distinguish between our construction and a random permutation.

2.5 Quantum Security Proof

We also analyse the security of the construction against a quantum adversary, and obtain some basic bounds. We show that our proof has $n/6$ bits of quantum security. In order to prove this bound, we exploit the fact that the construction starts with two encryption layers and then relate the quantum security to the classical security using Zhandry's quantum lower bounds on small range functions. However, we don't know of any attack that would perform better than the classical distinguisher running in time $O(2^n)$. Therefore, we don't believe our results are tight and discuss potential improvements of our results. A natural way of doing so would be to use Zhandry's technique of recording with permutation, but it is still an active topic in the field to understand well the quantum query recording in this setting.

2.6 Instantiations

Finally, we propose some concrete instantiations of our construction when using as building blocks (reduced-round versions of) AES-128. We propose Double-AES, where the blocks are slightly-tweaked versions (constant-wise) of the full 10-round AES, and we also propose Double-AES-7, where the number of rounds is reduced to 7 in all the blocks, and Double-AES-5-MC, a variant with 5 rounds but that includes the last MC transformation in E_1, E_2 and E. Our security claims of n bit security, are, for the first time to the best of our knowledge, unified, as we claim a unique security against all adversaries, whether they are classical or quantum. We believe that this is a trend that might become predominant as the post-quantum future might approach.

3 Constructions based on Encrypt-Mix-Encrypt Paradigm: First Attempt and Attack

Our aim is to find a $2n$-bit-to-$2n$-bit encryption mode using an n-bit blockcipher, ideally with five or fewer calls to the blockcipher. Specifically, we studied the encrypt-mix-encrypt paradigm, where the plaintext blocks first pass through a (weak) encryption layer, then an invertible mixing layer with possibly non-linear components, and then another encryption layer. For the encryption layers we can begin with something very simple, like an ECB layer (with different keys). Then our generic encryption function becomes

$$(L, R) \mapsto (E_3(M(E_1(L), E_2(R))_\ell), E_4(M(E_1(L), E_2(R))_r)),$$

where $E_1, \ldots, E_4$ are blockciphers with independent keys, and M is a $2n$-bit-to-$2n$-bit mixing layer with $M(\cdot,\cdot)_\ell$ and $M(\cdot,\cdot)_r$ indicating the left and right halves of its output respectively (Fig. 3, left).

For a specific mixing function, we take a $2n$-bit-to-n-bit compressing function F and use

$$M(x, y) := (x \oplus F(x, y), y \oplus F(x, y))$$

as our mixing layer (Fig. 3, centre). Since M needs to be invertible, we have the condition on F that (x, y) should be recoverable from $(x \oplus F(x, y), y \oplus F(x, y))$. One easy way to achieve this is to take an n-bit-to-n-bit function f and define

$$F(x, y) := f(x \oplus y).$$

This was the first specific construction we considered (Fig. 3, right). We assumed f is a qprf, and $E_1, \ldots, E_4$ are qprp's.

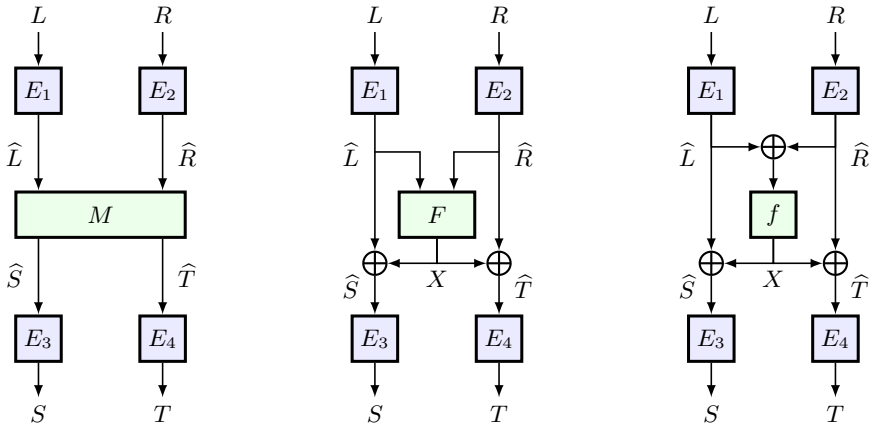


Fig. 3: **Left:** The generic construction, with an invertible mixing layer $M : \{0,1\}^{2n} \longrightarrow \{0,1\}^{2n}$. **Centre:** With a specific mixing layer, with a compressing function $F : \{0,1\}^{2n} \longrightarrow \{0,1\}^{n}$. **Right:** A more specific instantiation of F, with a prf $f : \{0,1\}^{n} \longrightarrow \{0,1\}^{n}$.

3.1 New superposition attack on the EME construction

While analyzing the quantum security of this construction with $F(x, y) := f(x \oplus y)$, we discovered a new kind of superposition attack.

The original idea of the attack is to build in the beginning a superposition of states that partially collide in the left output, and next, performing an exhaustive search of the key we are able to build a function that will be periodic in the subset of colliding states. This procedure is presented in Algorithm 1, and takes $\tilde{O}(2^{k/2} + 2^{n/3})$ computations for recovering the key of E_2. The same can be applied for recovering the keys of E_1, E_3 or E_4 (as the inverse has the same shape). This is a new kind of superposition attack, that combines for the first time BHT, Grover and Simon (introduced in subsection A.4), and where BHT is used to restraint the function to the interesting outputs that generate a partial collision, and that will next verify a certain property. We believe this attack might apply to more constructions, and it should be considered in further studies

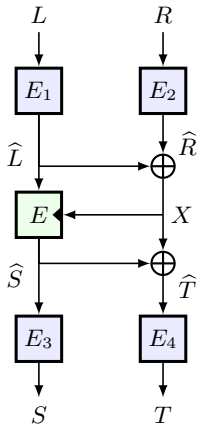


Fig. 4: The new construction QuEME; the E in the middle layer takes its key as an input from the right.

analyzing the security of symmetric primitives with respect to superposition attacks.

Description of the attack. Let $S(L, R)$ be the function corresponding to this construction for inputs L, R (see rightmost construction of Figure 3) so

$$S(L,R) = E_3\left(E_1(L) \oplus f(E_1(L) \oplus E_2(R))\right).$$

We start by fixing two distinct values L_0 and L_1 for left entries. Then, we consider the uniform superposition of claws between $F_0 : R \mapsto S(L_0, R)$ and $F_1 : R \mapsto S(L_1, R)$:

$$\frac{1}{\sqrt{|\{(R_0,R_1), S(L_0,R_0) = S(L_1,R_1)\}|}} \sum_{S(L_0,R_0)=S(L_1,R_1)} |R_0, R_1\rangle .$$

(We can obtain this with BHT algorithm with complexity $2^{n/3}$.)

We observe that $S(L_0, R_0) = S(L_1, R_1)$ is equivalent to

$$f(E_1(L_0) \oplus E_2(R_0)) \oplus f(E_1(L_1) \oplus E_2(R_1)) = E_1(L_0) \oplus E_1(L_1).$$

We add an external qubit $|b\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ to the state and apply the controlled exchange $(b, R_0, R_1) \mapsto (b, R_b, R_{1-b})$. The state becomes the uniform superposition of elements from

$$\{(b, R_0, R_1)/f(E_1(L_b) \oplus E_2(R_0) \oplus f(E_1(L_{1-b}) \oplus E_2(R_1) = E_1(L_0) \oplus E_1(L_1)\}.$$

This state is called $|\Phi\rangle$ for the rest of the attack.

Algorithm 1 Attack on construction

Input: superposition oracle access to E
Output: the key k_2 of E_2
Choose two values L_0 and L_1
Repeat $O(n)$ **times** (for confirmation)
  Search with BHT algorithm for claws on $F_b : R \mapsto S(L_b, R)$ (using $2^{n/3}$ turns)
    ▷ We get a uniform superposition $\sum |R_0, R_1\rangle$ of all elements of the set $\{(R, R')/f(E_1(L_0) \oplus E_2(R_0)) \oplus f(E_1(L_1) \oplus E_2(R_1)) = E_1(L_0) \oplus E_1(L_1)\}$.
EndRepeat
  ▷ We get $O(n)$ superpositions that we use as a database for the following Grover search
Grover search on k_2 with $2^{k/2}$ turns using the following oracle :
  ForEach superposition $\sum |R_0, R_1\rangle$
    Add an external qubit $|b\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$
    Apply $(b, R_0, R_1) \mapsto (b, E_{2,k_2}(R_b), E_{2,k_2}(R_{1-b}))$
      ▷ If we guessed right, we get a uniform superposition of the set $\{(b, R_0, R_1)/f(E_1(L_b) \oplus R_0) \oplus f(E_1(L_{1-b}) \oplus R_1) = E_1(L_0) \oplus E_1(L_1)\}$. This set admits the period $(1, E_1(L_0) \oplus E_1(L_1), E_1(L_0) \oplus E_1(L_1))$.
  EndFor
  Apply Simon's algorithm on the resulting superposition with the function $(b, R, R') \mapsto R' \oplus R$.
    ▷ If we guessed right, the function on this set admits the period $(1, E_1(L_0) \oplus E_1(L_1), E_1(L_0) \oplus E_1(L_1))$. Detecting a period on a wrong guess would mean a weakness of f.
  Uncompute to get back the superpositions $\sum |R_0, R_1\rangle$
EndGrover
Return k_2

Now, if we guess the key of E_2 right, we can apply $\mathcal{O}_{E_2}^{sup} : |x\rangle \to E_2(|x\rangle)$ on the 2 rightmost register of $|\Phi\rangle$[3] and get the superposition

$$\frac{1}{\sqrt{2|\{(R_0, R_1), S(L_0, R_0) = S(L_1, R_1)\}|}} \sum_{(b,R_0,R_1)\in A} |b, R_0, R_1\rangle$$

where $A = \{(b, R_0, R_1)/f(E_1(L_b)\oplus R_0)\oplus f(E_1(L_{1-b})\oplus R_1) = E_1(L_0)\oplus E_1(L_1)\}$.

This set admits the period $(1, E_1(L_0) \oplus E_1(L_1), E_1(L_0) \oplus E_1(L_1))$, i.e. if (b, R_0, R_1) is in the set then $(b \oplus 1, R_0 \oplus E_1(L_0) \oplus E_1(L_1), R_1 \oplus E_1(L_0) \oplus E_1(L_1))$ is also in the set.

Then, we apply Simon's algorithm on $(b, R_0, R_1) \mapsto R_0 \oplus R_1$ to recover the existence of this period and uncompute the last steps to recover the states $|\Phi\rangle$.

This part is the execution of Theorem 1 below with $g = 0$, $A = \{(b, R_0, R_1)|S(L_0, R_0) = S(L_1, R_1)\}$, $f_i' : (b, R_0, R_1) \mapsto (b, E_{2,k_2}(R_b), E_{2,k_2}(R_{1-b}))$, $f_i : (b, R, R') \mapsto R' \oplus R$ and $s = (1, E_1(L_0) \oplus E_1(L_1), E_1(L_0) \oplus E_1(L_1))$.

[3] This is possible since from the key of E_2, we can compute E_2 and E_2^{-1} efficiently. We then compute $|x\rangle|0\rangle \xrightarrow{O_{E_2}} |x\rangle|E_2(x)\rangle \xrightarrow{Swap} |E_2(x)\rangle|x\rangle \xrightarrow{O_{E_2^{-1}}} |E_2(x)\rangle|0\rangle$.

Theorem 1. *Suppose that $m = O(n)$, $\{f_i\}$ ia a family of public functions from $\{0,1\}^n \to \{0,1\}^l$, $\{f'_i\}$ ia a family of public permutations from $\{0,1\}^n \to \{0,1\}^l$ and $g : A \subseteq \{0,1\}^n \to \{0,1\}^l$ on which we only get some databases $|\phi_g\rangle$ and there is a unique i_0 such that $f_{i_0} \oplus g \circ f'_{i_0}$ has a period s and*

$$\max_{i,t \notin \{0,1\}^m \times \{0\} \cup \{i_0,s\}} \mathbb{P}_{x \in f_i'^{-1}(A)}((f_i \oplus \tilde{g} \circ f'_i)(x \oplus s) = (f_i \oplus g \circ f'_i)(x)) \leq \frac{1}{2}$$

With $O(n)$ databases $|\phi_g\rangle = \sum_{x \in A} \frac{1}{\sqrt{|A|}} |x\rangle |g(x)\rangle$, we can recover i_0 with a probability in $\Theta(1)$. The running time is $O(n^3 2^{m/2})$.

This can be deduced as a modification of the Offline-Simon algorithm [BHN+19], which is explained in more detail in section B from the supplementary material.

Similar Mixing Layers. While the previous description of our attack targets the specific mixing layer from figure 3-right, our attack can be adapted and impact also the other mixing layers $M(x, y)$ the following way. We describe the mixing layer the following way:

$$M(x, y) = \Pi_2(f(\Pi_1(x, y)), x, y)$$

By linearity, we write $\Pi_1(x, y) = \Pi_{1,L}(x) \oplus \Pi_{1,R}(y)$ and $\Pi_2(f, x, y) = f \oplus \Pi_{2,L}(x) \oplus \Pi_{2,R}(y)$ (even if it means rewriting the function f). The collision equation $S(L_0, R_0) = S(L_1, R_1)$ is then equivalent to

$$\begin{aligned} f(\Pi_{1,L} \circ E_1(L_0) \oplus \Pi_{1,R} \circ E_2(R_0)) \oplus f(\Pi_{1,L} \circ E_1(L_1) \oplus \Pi_{1,R} \circ E_2(R_1)) = \\ \Pi_{2,L} \circ E_1(L_0) \oplus \Pi_{2,L} \circ E_1(L_1) \oplus \Pi_{2,R} \circ E_2(R_0) \oplus \Pi_{2,R} \circ E_2(R_1). \end{aligned}$$

With a good guess and the controlled exchange, the equation of collisions becomes

$$\begin{aligned} f(\Pi_{1,L} \circ E_1(L_b) \oplus \Pi_{1,R}(R_0)) \oplus f(\Pi_{1,L} \circ E_1(L_{1-b}) \oplus \Pi_{1,R}(R_1)) = \\ \Pi_{2,L} \circ E_1(L_0) \oplus \Pi_{2,L} \circ E_1(L_1) \oplus \Pi_{2,R}(R_0) \oplus \Pi_{2,R}(R_1). \end{aligned}$$

Then, we discuss what happens whether $\Pi_{1,R}$ is reversible or not.

First case: $\Pi_{1,R}$ is not reversible. Then there exists $t \neq 0$ such that $\Pi_{1,R}(t) = 0$. and the set of collisions admits the period $s = (0, t, t)$.

Second case: $\Pi_{1,R}$ is reversible. Then we note $t = \Pi_{1,R}^{-1} \circ \Pi_{1,L}(E_1(L_0) \oplus E_1(L_1))$ and the set of collisions admits the period $s = (1, t, t)$.

Overall, the attack works the same way but the recovered period will be different.

4 New Construction and Classical Security Proofs

In this paper we propose the new construction QuEME shown in shown in Fig. 4. We define $\mathsf{QuEME}^E : \{0,1\}^{2n} \longrightarrow \{0,1\}^{2n}$ as

$$\mathsf{QuEME}^E(L,R) := (S,T),$$

where

$$\begin{aligned}
&\widehat{L} = E(K_1, L), \quad \widehat{R} = E(K_2, R),\\
&X = \widehat{L} \oplus \widehat{R},\\
&\widehat{S} = E(X, \widehat{L}), \quad \widehat{T} = X \oplus \widehat{S},\\
&S = E(K_3, \widehat{S}), \quad T = E(K_4, \widehat{T}),
\end{aligned}$$

for a block-cipher $E(\cdot\,,\cdot)$ and four keys $K_1, \ldots, K_4$. For each $i \in [1..4]$ we'll use the notation write $E_i := E(K_i, \cdot)$.

4.1 Proof of Classical Security

First we analyse the information-theoretic security of QuEME against a classical adversary. To move to the information-theoretic setting, we replace E_1, E_2, E_3, E_4 with independent random permutations π_1, π_2, π_3, π_4, and $E(\cdot,\cdot)$ with a tweakable random permutation $\widetilde{\pi}$. We call this modified construction QuEME^{π}.

Since the inner call to E uses a dynamic key derived from the internal state, we consider the possibility that $\mathcal{A}$ tries to guess the internal key with offline queries to E. However, we note that for a good block-cipher, the only way this can provide $\mathcal{A}$ additional information about the game is if they can guess a key-message pair or a key-ciphertext pair. Since there are $O(q)$ internal input and output pairs for E, this gives a $O(qq'/2^{2n})$ probability of guessing one such pair correctly with q' offline queries to E. In addition, let $\alpha^E(q,q')$ be an upper-bound on the advantage of an adversary who has learnt q' key-plaintext-ciphertext triples for E and is trying to distinguish q other key-plaintext-ciphertext triples from random. In our security bounds, whenever we switch from QuEME^{π} to QuEME^E, we add the terms $O(qq'/2^{2n})$ and $\alpha^E(q,q')$ to $\mathcal{A}$'s advantage as the cost of replacing the inner call to E with $\widetilde{\pi}$.

Settings of the Game. For convenience, we first make a few standard stipulations and modifications to the sprp game. We do not allow $\mathcal{A}$ to make any *redundant queries*, which can either be repetitions of earlier queries or the feeding back in the opposite direction of an earlier response. Since such queries can give $\mathcal{A}$ no extra information, this constraint cannot decrease the advantage of $\mathcal{A}$.

As a final modification, we consider the extended transcript game defined in Sec. A.3, with $\tau = \{(L^i, R^i, S^i, T^i) \mid i \in [1..q]\}$ and $\tau^* = \{(\widehat{L}^i, \widehat{R}^i, X^i, \widehat{S}^i, \widehat{T}^i) \mid i \in [1..q]\}$. Defining the sampler $\mathscr{S}$ will be a critical part of the proof.

Transcript Graphs. As preparation for sampling the internal transcript τ^*, we first define two undirected bipartite graphs G and H on the external transcript τ. The vertices of G are the q_1 distinct values $L_1, \ldots, L_{q_1}$ in the set $\{L^i \mid i \in [1..q]\}$ and the q_2 distinct values $R_1, \ldots, R_{q_2}$ in the set $\{R^i \mid i \in [1..q]\}$ (we'll soon specify how we pick these labels); we put an edge between L_j and R_k if they appear together in some query, i.e., there is a query $i \in [1..q]$ with $(L^i, R^i) = (L_j, R_k)$. H is defined identically except with the ciphertexts $\{(S^i, T^i) \mid i \in [1..q]\}$ replacing the plaintexts in the above definition of G.

Let α (resp. β) be the number of components in G (resp. H). We label these components $G^{(1)}, \ldots, G^{(\alpha)}$ and $H^{(1)}, \ldots, H^{(\beta)}$. For $t \in [1..\alpha]$ let $q_1^{(t)}$ (resp. $q_2^{(t)}$) be the number of L-nodes (resp. R-nodes) in $G^{(t)}$. Similarly, for $t \in [1..\beta]$ let $q_3^{(t)}$ (resp. $q_4^{(t)}$) be the number of S-nodes (resp. T-nodes) in $H^{(t)}$. Define the cumulative sums

$$Q_b^{(j)} = \sum_{i=1}^{j-1} q_b^{(i)}$$

for each $j \in [1..\alpha]$ when $b \in \{1, 2\}$ and each $j \in [1..\beta]$ when $b \in \{3, 4\}$. We assume the labelling of the L-nodes in G is such that the nodes $\{L_k \mid Q_1^{(j)} + 1 \leq k \leq Q_1^{(j+1)}\}$ are in $G^{(j)}$, and likewise for the R-nodes, S-nodes, and T-nodes.

Classical Security Claim. We claim the following security bound for QuEME^π.

Theorem 2. *For any classical adversary $\mathcal{A}$ playing a q-query IND-CCA game against QuEME^π, we have*

$$\boldsymbol{Adv}_{sprp}^{\mathsf{QuEME}^\pi}(\mathcal{A}) = O\left(\frac{q}{2^n}\right).$$

For any classical adversary $\mathcal{A}'$ playing a q-query IND-CCA game against QuEME^E with q' offline queries to E, we have

$$\boldsymbol{Adv}_{sprp}^{\mathsf{QuEME}^E}(\mathcal{A}') \leq O\left(\frac{q}{2^n}\right) + O\left(\frac{qq'}{2^{2n}}\right) + \alpha^E(q, q').$$

The proof of this relies on the conjectured Tight Mirror Theorem (Theorem 6), and is found in App. C.2 in the Supplementary Material. A simulation supporting the conjecture following an innovative approach can be found in section 5.

4.2 IND-CCA security proof of $(2n/3)$-bit Security

In this subsection we instead prove the following weaker result without relying on any conjectured bound.

Theorem 3. *For any classical adversary $\mathcal{A}$ playing a q-query IND-CCA game against QuEME^π, we have*

$$\boldsymbol{Adv}_{sprp}^{\mathsf{QuEME}^\pi}(\mathcal{A}) = O\left(\frac{q^3}{2^{2n}}\right).$$

For any classical adversary $\mathcal{A}'$ playing a q-query IND-CCA game against QuEME^E with q' offline queries to E, we have

$$\boldsymbol{Adv}^{\mathsf{QuEME}^E}_{sprp}(\mathcal{A}') \leq O\left(\frac{q^3}{2^{2n}}\right) + O\left(\frac{qq'}{2^{2n}}\right) + \alpha^E(q, q').$$

When decryption queries are allowed, we partition the queries into two sets: let $\mathcal{I}^*$ contain the queries where both output blocks are *fresh*, and $\mathcal{I}$ contain the queries where one of the output blocks collides with an earlier block at the same position.

Sampler of Internal Transcripts. Here we'll define the sampler $\mathscr{S}$ which takes $\tau = \{(L^i, R^i), (S^i, T^i) \mid i \in [1..q]\}$ as input and samples a $\tau^* = \{(\widehat{L}^i, \widehat{R}^i, \widehat{S}^i, \widehat{T}^i) \mid i \in [1..q]\}$. The sampling proceeds as follows:

1. $\mathscr{S}$ initialises four tables $D_{\widehat{L}}$, $D_{\widehat{R}}$, $D_{\widehat{S}}$, and $D_{\widehat{T}}$ as empty.
2. For an encryption query i, $\mathscr{S}$ first checks the tables $D_{\widehat{L}}$ and $D_{\widehat{R}}$ to see if $\widehat{L}^i$ or $\widehat{R}^i$ has already been sampled; whichever is not found in the table is freshly sampled from the set of unsampled values. X^i is set to be $\widehat{L}^i \oplus \widehat{R}^i$.
3. If $i \in \mathcal{I}$, one of $\widehat{S}^i$ and $\widehat{T}^i$ is already sampled, so the other one is set as the sum of the sampled one and X^i.
4. If $i \in \mathcal{I}^*$, both $\widehat{S}^i$ and $\widehat{T}^i$ are fresh, so $\widehat{S}^i$ is sampled from outside $D_{\widehat{S}}$, such that $\widehat{T}^i = \widehat{S}^i \oplus X^i$ is also outside $D_{\widehat{T}}$.
5. For a decryption query i, $\mathscr{S}$ first checks the tables $D_{\widehat{S}}$ and $D_{\widehat{T}}$ to see if $\widehat{S}^i$ or $\widehat{T}^i$ has already been sampled; whichever is not found in the table is freshly sampled from the set of unsampled values. X^i is set to be $\widehat{S}^i \oplus \widehat{T}^i$.
6. If $i \in \mathcal{I}$, one of $\widehat{L}^i$ and $\widehat{R}^i$ is already sampled, so the other one is set as the sum of the sampled one and X^i.
7. If $i \in \mathcal{I}$, both $\widehat{L}^i$ and $\widehat{R}^i$ are fresh, so $\widehat{L}^i$ is sampled from outside $D_{\widehat{L}}$, such that $\widehat{R}^i = \widehat{L}^i \oplus X^i$ is also outside $D_{\widehat{R}}$.

Bad Events. We define the following bad events on the random coins of f and $\mathscr{S}$:

bad_0: For some $i, i', i'' \in [1..q]$ with $i > i'$ and $i > i''$, $S^i = S^{i'}$ and $T^i = T^{i''}$;
bad_1: In an encryption query $i \in \mathcal{I}$, a previously unsampled $\widehat{S}^i$ or $\widehat{T}^i$ is set to be equal to a previously sampled value at the same position;
bad_2: In a decryption query $i \in \mathcal{I}$, a previously unsampled $\widehat{L}^i$ or $\widehat{R}^i$ is set to be equal to a previously sampled value at the same position.

In bad_1 and bad_2 we assume bad_0 has not happened.

Bad Probabilities. For bad_0 the two collisions have a joint probability of $1/N^2$, with choice of the three indices i, i', i''. Thus,

$$\Pr[\mathsf{bad}_0] \leq \frac{q^3}{N^2}. \tag{1}$$

For bad_1 we need one collision with a previous i' for $i \in \mathcal{I}$, and one collision with a previously sampled value at some i''. Again they have a joint probability of $1/N^2$, with choice of the three indices i, i', i''. Thus,

$$\Pr[\mathsf{bad}_1] \leq \frac{q^3}{N^2}. \tag{2}$$

Similarly we can show that

$$\Pr[\mathsf{bad}_2] \leq \frac{q^3}{N^2}. \tag{3}$$

Ratio of Good Probabilities. As before, in the real world, we have

$$\Pr_1[(\tau, \tau^*)] = \frac{1}{(N)_{q_1} \dots (N)_{q_4} (N)_{t_1} \dots (N)_{t_r}}. \tag{4}$$

In the ideal world, we have a term $1/N^{2q}$ that comes from the sampling of the outputs in the online phase. As before, we use one N^q to cancel out $(N)_{t_1} \dots (N)_{t_r}$. If the j-th unique $\widehat{L}$ or $\widehat{R}$ first appears in an encryption query, it will contribute a term $1/(N-j-1)$ to the probability, and the resulting $(N-j-1)$ in the numerator will cancel out the same term in $(N)_{q_1}$ or $(N)_{q_2}$. Similarly, if the j-th unique $\widehat{S}$ or $\widehat{T}$ first appears in an encryption query, it will contribute a term $1/(N-j+1)$ to the probability, and the resulting $(N-j+1)$ in the numerator will cancel out the same term in $(N)_{q_3}$ or $(N)_{q_4}$. For each encryption query in $\mathcal{I}$, no sampling is done for $\widehat{S}$ or $\widehat{T}$, and for each decryption query in $\mathcal{I}$, no sampling is done for $\widehat{L}$ or $\widehat{R}$, so these do not contribute anything to the probability.

Finally, consider an encryption query in $\mathcal{I}^*$ that contains the j-th unique $\widehat{S}$ and j'-th unique $\widehat{T}$. Thus, when sampling $\widehat{S}$ we need to avoid $j + j' - 2$ values, so this contributes a term $1/(N-j-j'+2)$ to the probability. We combine this $(N-j-j'+2)$ in the numerator with one N-term in the numerator and the terms $(N-j+1)$ and $(N-j'+1)$ in the denominator (from $(N)_{q_3}$ or $(N)_{q_4}$ respectively), to get

$$\frac{N(N-j-j'+2)}{(N-j+1)(N-j'+1)} = 1 - \frac{(j-1)(j'-1)}{(N-j+1)(N-j'+1)} \geq 1 - \frac{2jj'}{N^2}, \tag{5}$$

where we use the inequalities $j, j' \leq N(1-1/\sqrt{2})$. This uses up $N^{|\mathcal{I}^*|}$, and the remaining $N^{|\mathcal{I}|}$ is used to cancel out the remaining terms in the denominator.

Thus we have

$$\rho \geq \prod_{j,j'} \left(1 - \frac{2jj'}{N^2}\right) \geq \prod_{j \in [1..q]} \left(1 - \frac{2j^2}{N^2}\right) \geq 1 - \frac{q^3}{N^2}, \tag{6}$$

since we can replace the smaller of j and j' with the bigger one without breaking the inequality. This completes the proof with $\epsilon_2 = q^3/N^2$.

5 Simulation of the Mirror Theory

In the proof of Theorem 2 we use a conjectured result from a line of research that focuses on finding tight approximations on the number of solutions to systems of bi-variate equations, which goes by the name Mirror Theory [Pat03] (see App. A.2). The particular conjecture we use can be summarised as follows: Consider two sequences of n-bit variables $Y_1, \ldots, Y_{q_1}$ and $Z_1, \ldots, Z_{q_2}$, with $q_1 < N, q_2 < N$, where N denote 2^n. For some $q < q_1 + q_2$ suppose there are q bi-variate equations of the form

$$Y_i \oplus Z_j = \delta_{ij}.$$

Consider the graph where this equation is represented by an edge between Y_i and Z_j. We assume none of the equations is redundant and the system of equations is consistent, so there are no cycles in this graph. Let $C^{(1)}, \ldots, C^{(t)}$ be the connected components, where $t = q_1 + q_2 - q$. For each $j \in [1..t]$ let $q_1^{(j)}$ (resp. $q_2^{(j)}$) be the number of Y_i's (resp. Z_i's) that appear in $C^{(j)}$. Finally, define the cumulative sums

$$Q_b^{(j)} = \sum_{i=1}^{j-1} q_b^{(i)}$$

for each $b \in \{1, 2\}$ and each $j \in [1..t]$. Then the conjectured Tight Mirror Theorem (Theorem 6) claims that the number of solutions to this system such that Y_i's are all distinct and Z_i's are all distinct is at least

$$\frac{1}{N^q} \cdot \prod_{j=1}^{t} \left[\left(N - Q_1^{(j)}\right)^{q_1^{(j)}} \left(N - Q_2^{(j)}\right)^{q_2^{(j)}} \right] \cdot (1 - \epsilon),$$

where $\epsilon = O(q/N)$.

In order to verify this conjecture, we have implemented a simulation that allow us to predict the number of possible internal transcripts. In this section, we describe these simulations that are, to the best of our knowledge, an innovative approach that hasn't been used before in this context.

First step: getting the sets of connected components. The first step for comparing experiments to the conjecture is computing the set of the connected components of two or more variables. We recall that there is an edge between the variables X_i and Y_j if and only if there is the equation $X_i \oplus Y_j = \delta_{i,j}$ and $X_{i'} \oplus Y_{j'} = \delta_{i',j'}$. First we make a list of equations sorted by index i and one sorted by index j for retrieving the edges quickly, then we apply a classic Breadth-First Traversal of the graph. The procedure to get it is described in Algorithm 2 below:

Naive approach. The most natural way of doing once obtained the connected components is to generate the solutions. This can be done by initializing two sets of remaining places to $\{0,1\}^n$ (S_X for the variables X and S_Y for the variables Y). We take the largest component, find a place α for its root and for every

Algorithm 2 Retrieving the sets on connected components

Input: List of equations of the form $X_i \oplus Y_j = \delta_{i,j}$
Output: The list of connected components
Sort the equations $X_i \oplus Y_j = \delta_{i,j}$ by i the result is named L_X
▷ There needs to have a place to mark the different i.
Sort the equations $X_i \oplus Y_j = \delta_{i,j}$ by j the result is named L_Y
▷ There needs to have a place to mark the different j.
for all i **do**
    Start a pile with the element $(X, i, 0)$
    Start a list for recording the elements of the current connected component with the root element $(X, i, 0)$
    while the pile is not empty **do**
        Pop the first element (Z, l, Δ)
        if l is not marked in the list L_Z **then**
            for all $X_i \oplus Y_j = \delta_{i,j}$ in L_Z with $i = l$ if $X = Z$ and $j = l$ otherwise **do**
                Add $(Y, j, \Delta \oplus \delta_{i,j})$ to the pile and to the component if $X = Z$ and $(X, i, \Delta \oplus \delta_{i,j})$ otherwise
            end for
            Mark l in the list L_Z
        end if
    end while
    Register the list if it contains more than one element
        ▷ This connected component has either no elements or is a isolated point.
end for
Return the lists of connected components

points (Z, l, Δ) of this component ruling out $\alpha \oplus \Delta$ from the the corresponding variable (S_X if $Z = X$ and S_Y otherwise). We take the second largest, find a place for the second root β such that for every point (Z, l, Δ) of this second component $\beta \oplus \Delta$ is in the set of remaining places of corresponding variable and once a β is found we rule them out. We continue for the remaining components until there is either no more component (making a solution) or there is no viable place for a root (the earlier choices did not lead to a solution). While this method is not practical as it generates every solution which is double exponential on the size of the input, it is the only one (to our knowledge) to give the exact number of solutions.

Approximation. In this paragraph, we describe a method that allows to propose an approximation of the number of solutions of the system given by the equations $X_i \oplus Y_j = \delta_{i,j}$. For a given set of equations we want to obtain the connected components (C_i) as described before and m their number and their size. We note $\Delta_{i,j}$ the set of placements of the connected components such that the components C_i and C_j collide. Then by the formula of the cardinal of a union, we get

$$2^{nm} - |\text{solutions}| = \sum_{k=1}^{m} (-1)^{k-1} \sum_{\{i_1,j_1\}>\ldots>\{i_k,j_k\}} |\Delta_{i_1,j_1} \cap \ldots \cap \Delta_{i_k,j_k}|$$

where $>$ is an order.

A first observation is that for all $\{i_1, j_1\} \neq \{i_2, j_2\}$,

$$|\Delta_{i_1,j_1} \cap \Delta_{i_2,j_2}| \times 2^{nm} = |\Delta_{i_1,j_1}| \times |\Delta_{i_2,j_2}|$$

as the difference between the value of the roots of C_{i_1} and C_{j_1} and between C_{i_2} and C_{j_2} are independent even for cases $\{i_1, j_1\} \cap \{i_2, j_2\} \neq \emptyset$. For further use, for a set $\{i_1, j_1\} > ... > \{i_k, j_k\}$, we define the graph $G_{\{i_1,j_1\}>...>\{i_k,j_k\}}$ with vertices $1, ..., m$ and there is an edge between vertices a and b if and only if there exist l such that $\{i_l, j_l\} = \{a, b\}$. For a graph G on vertices $1, ..., m$, we define

$$S_G = \frac{1}{2^{nm}} \sum_{G' \cong G} \Big| \bigcap_{i,j \in G'} \Delta_{i,j} \Big|,$$

where $\cong$ denotes graph isomorphism. We let C_G be the number of connected components of G and P_G the number of non-isolated points.

This first observation extents to the computation of any S_G where G has no cycle and to unions of not connected sub-graphs. S_G can be bounded by $(\sum_i |C_i|^2)^{P_G}/2^{n(m-C_G)}$. This means that this method of approximation is suited for cases where the value $\sum_i |C_i|^2$ is controlled. For random systems, $\sum_i |C_i|^2 = O(q)$. Then a first approximation can be made by taking

$$\frac{|\text{solutions}|}{2^{nm}} = \prod_{i,j} \left(1 - \frac{|\Delta_{i,j}|}{2^n}\right) + O\left(\frac{q^3}{2^{2n}}\right).$$

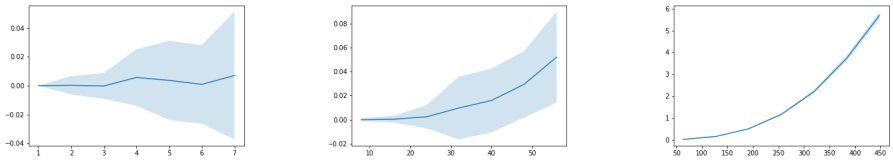
More advanced approximations can be made by considering cycles of successively bigger sizes. (For example, by considering the cycles of size 3, we get a better approximation with an error in $O(q^4/2^{3n})$.)

The figure 5 shows the different simulations of the different approximations. It took 10 hours on an Intel i5-6500U CPU to compute. The approximations tend to get more solutions than the conjecture, but remaining in the expected error.

Conclusion. Overall our simulations confirm that the tighter version of the mirror theory holds for small bit sizes ($n \leq 5$ for the exact approach, $n \leq 8$ for the better approximation, and $n \leq 11$ for the first approximation). While it becomes infeasible to run the simulations for usable values of n, since we did not use any special properties of small n-values, our results seem to suggest that it should also hold for larger values of n, making our conjecture a reasonable one.

6 An $O(2^n)$-distinguisher on our scheme

We have black box access to a function f st. $f \xleftarrow{\$} P_{2n}$ or $f \leftarrow \mathsf{QuEME}^{\pi}$, and we want to distinguish in which case we are. We present an distinguisher that performs only forward queries to f. Assume on query $(x_1||x_2)$ we get output



(a) exact approach ($n = 5$) (b) better approximation ($n = 8$) (c) first approximation ($n = 11$)

Fig. 5: Results of simulations: logarithmic difference between simulation and conjectural prediction by number of equations (line is mean over 100 tries, area is variability)

$(y_1||y_2)$. In the first case, $(y_1||y_2)$ is a uniformly random string that hasn't been outputted before for any $(x_1||x_2)$ that hasn't bee queried before. In the second case, there exists permutations $\pi_1, \pi_2, \pi_3, \pi_4$ such

$$\pi_1(x_1) \oplus \pi_2(x_2) = \pi_3^{-1}(y_1) \oplus \pi_4^{-1}(y_2). \tag{7}$$

Our proof will use linear algebra techniques. Let $N = 4 \cdot 2^n$. Let $x_1, x_2, y_1, y_2 \in \{0,1\}^n$ that we interpret as integers in $[0, 2^n - 1]$, and let $\mathbf{e}_{x_1 x_2 y_1 y_2} \in \mathbb{F}_2^N$ be the binary column vector where the i^{th} coordinate of $\mathbf{e}_{x_1 x_2 y_1 y_2}$ is equal to 1 iff. $i = x_1$, $i = x_2 + 2^n$, $i = y_1 + 2 \cdot 2^n$ or $t = y_2 + 3 \cdot 2^n$ and is equal to 0 otherwise. This means each $\mathbf{e}_{x_1 x_2 y_1 y_2} \in \mathbb{F}_2^N$ has weight 4, meaning four non-zero coordinates.

The idea of the distinguisher is that if the following: perform M queries $\{x_1^i x_2^i || y_1^i y_2^i\}_{i \in [M]}$, let $H = span\{\vec{e}_{x_1^i x_2^i y_1^i y_2^i}\}_{i \in [M]}$. For M large enough (but linear in N), we will show that if we queried our QuEME^π construction, we have $dim(H) \leq N - 2$ with overwhelming probability, which we prove using Equation 7. On the other hand, if we start from a random permutation, then we can show that $dim(H) \geq N - 1$ since the $\vec{e}_{x_1^i x_2^i y_1^i y_2^i}$ will essentially be random vectors of $\mathbb{F}_2^N$ of weight 4.

We consider the following adversary

Adversary $\mathcal{A}$ for distinguishing the QuEME^π construction from a random permutation

- Perform $M = 4N$ random different queries $(x_1^i||x_2^i)$ for $i \in [M]$ and get respective outputs $(y_1^i||y_2^i)$.
- Let $H = span\{\vec{e}_{x_1^i x_2^i y_1^i y_2^i}\}_{i \in [M]}$. Compute $dim(H)$.
- If $dim(H) \leq N - 2$, return "EME" else return "random permutation".

Proposition 1. *If the adversary queries a function $f \leftarrow \mathsf{QuEME}^{\pi}$, we have $dim(H) \leq N-2$ with overwhelming probability.*

Proof. Take $f \leftarrow \mathsf{QuEME}^{\pi}$. This means in particular we choose random permutations $\pi_1, \pi_2, \pi_3, \pi_4$ and for each query x_1^i, x_2^i that gives output y_1^i, y_2^i, we have

$$\pi_1(x_1^i) \oplus \pi_2(x_2^i) = \pi_3^{-1}(y_1^i) \oplus \pi_4^{-1}(y_2^i).$$

Consider the following matrix $M \in F_2^{n \times 4 \cdot 2^n}$: the first 2^n columns of M are the columns $\begin{pmatrix} [\pi_1(x)]_1 \\ \vdots \\ [\pi_1(x)]_n \end{pmatrix}$ for $x \in \{0,1\}^n$. Then, the next 2^n columns are the same but we replace π_1 with π_2, and similarly with the third and last where we have π_3^{-1} and π_4^{-1} respectively instead of π_1. So we can write

$$M = \left(\begin{pmatrix} [\pi_1(0)]_1 \\ \vdots \\ [\pi_1(0)]_n \end{pmatrix} \cdots \begin{pmatrix} [\pi_2(0)]_1 \\ \vdots \\ [\pi_2(0)]_n \end{pmatrix} \cdots \begin{pmatrix} [\pi_3^{-1}(0)]_1 \\ \vdots \\ [\pi_3^{-1}(0)]_n \end{pmatrix} \cdots \begin{pmatrix} [\pi_4^{-1}(0)]_1 \\ \vdots \\ [\pi_4^{-1}(0)]_n \end{pmatrix} \cdots \right)$$

Because $\pi_1, \pi_2, \pi_3, \pi_4$ are permutations, the matrix M contains at least 2 different non-zero lines, therefore $dim(M) \geq 2$. Also notice that

$$M \cdot \vec{e}_{x_1 x_2 y_1 y_2} = \pi_1(x_1) \oplus \pi_2(x_2) \oplus \pi_3^{-1}(y_1) \oplus \pi_4^{-1}(y_2).$$

so $H \subseteq Ker(M)$ and $dim(Ker(M)) = N - dim(M) \leq N-2$ from which we conclude $dim(H) \leq N-2$.

Proposition 2. *If the adversary queries a random permutation $f \xleftarrow{\$} P_{2n}$, we have $dim(H) = 4 \cdot 2^n - 1$ with overwhelming probability.*

Proof. Let $H_j = span\{\vec{e}_{x_1^i x_2^i y_1^i y_2^i}\}_{i \in [j]}$. We will show that if $dim(H_j) \leq N-2$ then with constant probability, $dim(H_{j+1}) = dim(H_j) + 1$. Let $H_j^{\perp}$ be the dual of H_j, so

$$x \in H_j \Leftrightarrow \forall y \in H_j^{\perp}, \langle x, y \rangle = 0.$$

We have $dim(H_j) + dim(H_j^{\perp}) = N$ so in particular $dim(H_j^{\perp}) \geq 2$. This means in particular that there exists two distinct non-zero vectors $z_1, z_2 \in H_j^{\perp}$. This again implies that there exists $z^* \in H_j^{\perp}$ st. $|z^*| \leq 2N/3$. One can indeed easily check that if $|z_1|, |z_2| > 2N/3$ then $|z_1 + z_2| \leq 2N/3$. For a random tuple x_1, x_2, y_1, y_2, we have

$$\begin{aligned} \Pr[\vec{e}_{x_1 x_2 y_1 y_2} \in H_j] &\leq \Pr[\langle \vec{e}_{x_1 x_2 y_1 y_2}, z^* \rangle = 0] \\ &\leq \left(\frac{1}{3}\right)^4 + 6\left(\frac{1}{3}\right)^2\left(\frac{1}{3}\right)^2 + \left(\frac{2}{3}\right)^4 = \frac{41}{81}. \end{aligned}$$

This gives $\Pr[\vec{e}_{x_1 x_2 y_1 y_2} \notin H_j] \geq 40/81$. In reality, the tuple $(x_1^{j+1} x_2^{j+1} y_1^{j+1} y_2^{j+1})$ is not entirely random. Indeed, while x_1^{j+1}, x_2^{j+1} are chosen uniformly at random, the outputs must satisfy the permutation constraints, meaning that if

$(x_1^{j+1}||x_2^{j+1})$ hasn't been queried then the output $(y_1^{j+1}||y_2^{j+1})$ must be different from the previous outputs. For a fixed query, this changes the output distribution by at most $O(j/2^{2n}) = O(1/2^n)$ (since there are $O(N) = O(2^n)$ queries in total, so $j \leq O(2^n)$). From there, we get

$$\Pr[\vec{e}_{x_1^{j+1}x_2^{j+1}y_1^{j+1}y_2^{j+1}} \notin H_j] \geq \frac{40}{81} - O\left(\frac{1}{2^n}\right).$$

when the above holds, this immediately implies that $dim(H_{j+1}) = dim(H_j) + 1$. Since $M = 4N$, this then implies that with overwhelming probability $dim(H_M) \geq N - 1$.

Quantising the distinguisher. So far we have not found any quantum versions improving the complexity, and we do not believe that this distinguisher can benefit of any speed-up in the quantum setting.

7 Quantum security

In this section, we study the quantum security of our constructions. We will show that the QuEME^π has $n/6$ bits of quantum security. While this isn't an as strong statement as in the classical setting, it implies at least that the security doesn't totally collapse when we consider quantum adversaries as is the case for the 3-round Luby-Rackoff construction [KM10].

To prove this statement, we will actually reduce the quantum security to classical security (in a non-tight way) using Zhandry's lower bound on small range functions. We then show how to prove our quantum security statement in this framework. Finally, we discuss our results and ways of improving these quantum security claims.

7.1 Hardness of distinguishing a random permutation from a random function with small range

Our proof will use a quantum lower bound on distinguishing a random permutation from a random function with small range proven in [Zha15]. We first define a distribution $S_n(r)$ of small range functions:

Definition 1. *$S_n(r)$ is a distribution on functions from $\{0,1\}^n$ to $\{0,1\}^n$ sampled as follows:*

- *Draw a random function g from $\{0,1\}^n \to [r]$.*
- *Draw a random injective function h from $[r] \to \{0,1\}^n$.*
- *Output the composition $h \circ g$.*

Notice that any function f drawn from $S_n(r)$ satisfies $|Im(f)| \leq r$. Let also P_n be the uniform distribution on permutations on $\{0,1\}^n$. Zhandry's lower bound can be stated as follows:

Proposition 3 ([Zha15]). *For any r, for any quantum adversary $\mathcal{A}^f$ performing q queries to f, we have*

$$\boldsymbol{Adv}_{qprp}^{S_n(r)}(\mathcal{A}) = O\left(\frac{q^3}{r}\right).$$

7.2 Quantum Security Statement

As in the classical setting, we consider the prp advantage against quantum adversaries that can query the whole function QuEME^E but also the inner function E. In the quantum setting, both these queries can be quantum queries.

Theorem 4. *For any quantum adversary $\mathcal{A}$ playing a q-query IND-CPA game against QuEME^E with q' offline (quantum) queries to E, we have*

$$\boldsymbol{Adv}_{prp}^{\mathsf{QuEME}^E}(\mathcal{A}) \leq O\left(\frac{(q+q')^2}{2^{n/3}}\right) + O\left(\left(\frac{(q+q')^2}{2^{n/3}}\right)^2\right) + \alpha^E(r^2, r^2).$$

with $r = O((q+q')2^{n/3})$, where recall that $\alpha^E(x, x')$ be an upper-bound on the advantage of an adversary who has learned x' key-plaintext-ciphertext triples for E and is trying to distinguish x other key-plaintext-ciphertext triples from random. This implies in particular that the advantage is small essentially up to $(q+q') = 2^{n/6}$.

Proof. We consider a quantum adversary $\mathcal{A}$ that performs q quantum queries to QuEME^E and q' quantum queries to E. We perform a game based proof to prove our statement. We start from Game 1 which corresponds to the prp-advantage.

Game1 → Game2. We transform Game1 into Game2 by adding two random permutations j_1, j_2 in the key and in the input of E.

Game1: prp-game($\mathcal{A}$)
$b \xleftarrow{\$} \{0,1\}$
$E \xleftarrow{\$} TBC_n$
$f \leftarrow \begin{cases} \mathsf{QuEME}^{\pi} & \text{if } b = 0 \\ P_{2n} & \text{if } b = 1 \end{cases}$
$b' \leftarrow A^{f,E}(\cdot)$
Win if $b = b'$

Game2: adding permutations to E
$b \xleftarrow{\$} \{0,1\}$
$E \xleftarrow{\$} TBC_n$
$j_1, j_2 \xleftarrow{\$} P_n$
Let E' st. $E'_k(x) = E_{j_1(k)}(j_2(x))$.
$f \leftarrow \begin{cases} \mathsf{QuEME}^{E'} & \text{if } b = 0 \\ P_{2n} & \text{if } b = 1 \end{cases}$
$b' \leftarrow A^{f,E'}(\cdot)$
Win if $b = b'$

Transforming E into E' doesn't change its distribution. It is still a random TBC_n. Therefore, this doesn't change the value of the game.

$$\Pr[\mathcal{A} \text{ wins Game1}] = \Pr[\mathcal{A} \text{ wins Game2}].$$

Game2 → Game3. We transform Game2 into Game3 by adding two random permutations on the top of $\mathsf{QuEME}^{E'}$.

Game2: adding permutations to E
$b \xleftarrow{\$} \{0,1\}$
$E \xleftarrow{\$} TBC_n$
$j_1, j_2 \xleftarrow{\$} P_n$
Let E' st. $E'_k(x) = E_{j_1(k)}(j_2(x))$.
$f \leftarrow \begin{cases} \mathsf{QuEME}^{E'} & \text{if } b = 0 \\ P_{2n} & \text{if } b = 1 \end{cases}$
$b' \leftarrow A^{f,E'}(\cdot)$
Win if $b = b'$

Game3: adding permutations to $\mathsf{QuEME}^{E'}$
$b \xleftarrow{\$} \{0,1\}$
$E \xleftarrow{\$} TBC_n$
$j_1, j_2 \xleftarrow{\$} P_n$
Let E' st. $E'_k(x) = E_{j_1(k)}(j_2(x))$.
$f \leftarrow \begin{cases} \mathsf{QuEME}^{E'} & \text{if } b = 0 \\ P_{2n} & \text{if } b = 1 \end{cases}$
$h_L, h_R \xleftarrow{\$} P_n$
$f' := f \circ (h_L \| h_R)$.
$b' \leftarrow A^{f',E'}(\cdot)$
Win if $b = b'$

In our encrypt then mix construction, we start already by 2 random permutations so adding an extra layer of permutations won't change the distribution $\mathsf{QuEME}^{E'}$. This of course doesn't change the distribution of random permutation as well hence

$$\Pr[\mathcal{A} \text{ wins Game2}] = \Pr[\mathcal{A} \text{ wins Game3}].$$

Game3 → Game4. We now replace all the permutations with random small-range functions.

Game3: adding permutations to $\mathsf{QuEME}^{E'}$
$b \xleftarrow{\$} \{0,1\}$
$E \xleftarrow{\$} TBC_n$
$j_1, j_2 \xleftarrow{\$} P_n$
Let E' st. $E'_k(x) = E_{j_1(k)}(j_2(x))$.
$f \leftarrow \begin{cases} \mathsf{QuEME}^{E'} & \text{if } b = 0 \\ P_{2n} & \text{if } b = 1 \end{cases}$
$h_L, h_R \xleftarrow{\$} P_n$
$f' := f \circ (h_L \| h_R)$.
$b' \leftarrow A^{f',E'}(\cdot)$
Win if $b = b'$

Game4: adding permutations to $\mathsf{QuEME}^{E'}$
$b \xleftarrow{\$} \{0,1\}$
$E \xleftarrow{\$} TBC_n$
$j_1, j_2 \xleftarrow{\$} S_n(r)$
Let E' st. $E'_k(x) = E_{j_1(k)}(j_2(x))$.
$f \leftarrow \begin{cases} \mathsf{QuEME}^{E'} & \text{if } b = 0 \\ P_{2n} & \text{if } b = 1 \end{cases}$
$h_L, h_R \xleftarrow{\$} S_n(r)$
$f' := f \circ (h_L \| h_R)$.
$b' \leftarrow A^{f',E'}(\cdot)$
Win if $b = b'$

From Zhandry's bound on small range functions, we have

$$\begin{aligned} \Pr[\mathcal{A} \text{ wins Game3}] &\leq \Pr[\mathcal{A} \text{ wins Game4}] + O\left(\frac{q^3}{r}\right) + O\left(\frac{q'^3}{r}\right) \\ &\leq \Pr[\mathcal{A} \text{ wins Game4}] + O\left(\frac{(q+q')^3}{r}\right). \end{aligned}$$

Game4: classical emulation. We consider the following classical adversary.

Classical adversary $\mathcal{B}^{f,E}$
1. Pick $j_1, j_2, h_L, h_R \xleftarrow{\$} S_n(r)$. Let Y_1, Y_2 be the ranges of j_1, j_2 respectively, and Z_L, Z_R be the ranges of h_L, h_R respectively, so they are each subsets of $\{0,1\}^n$ of size r. **2**. Define E' st. $E'_k(x) = E_{j_1(k)}(j_2(x))$ and $f' := f \circ (h_L \| h_R)$. **3**. Query $f(x,y)$ for each $(x,y) \in Z_L \times Z_R$ and query $E_k(x)$ for each $(k,x) \in Y_1 \times Y_2$, for a total of $2r^2$ queries. From these queries, recover the truth table of f' and E'. **4**. Emulate the quantum circuit $\mathcal{A}^{f',E'}(\cdot)$ and output $b' \leftarrow \mathcal{A}^{f',E'}(\cdot)$.

$B^{f,E}$ outputs exactly the same output as $A^{f',E'}$ so by definition, we have

$$\Pr[\mathcal{B} \text{ wins Game1}] = \Pr[\mathcal{A} \text{ wins Game4}].$$

and moreover, $\mathcal{B}$ is a classical algorithm that performs $2r^2$ queries to f. We still add a few remarks on the adversary $\mathcal{B}$. Notice that we limit the number of queries of $\mathcal{B}$ but not it's running time so we don't need to perform the different steps of the algorithm efficiently. From this, it makes it much easier to see how $\mathcal{B}$ performs the different steps of the algorithm. In particular, he chooses h_L, h_R, j_1, j_2 at random and knows the full description of these functions, so he knows Y_1, Y_2, Z_L, Z_R. Also, after his $2r^2$ queries, he can know the full truth table of f' and E', so he knows the full description of $A^{f',E'}$. A quantum algorithm on n qubits with m gates, can be emulated by a classical algorithm running in time exponential in n, m. However, he doesn't need any extra queries to f', E' since he already knows the full description of f' and E'. What is important here is that even though the running time is large, the amount of queries done to f and E is limited to r^2.

We can now conclude

$$\begin{aligned}
\mathbf{Adv}_{prp}^{\mathsf{QuEME}^E}(\mathcal{A}) &= \Pr[\mathcal{A} \text{ wins Game1}] \\
&\leq \Pr[\mathcal{A} \text{ wins Game4}] + O\left(\frac{(q+q')^3}{r}\right) \\
&= \Pr[\mathcal{B} \text{ wins Game1}] + O\left(\frac{(q+q')^3}{r}\right) \\
&= \mathbf{Adv}_{prp}^{\mathsf{QuEME}^E}(\mathcal{B}) + O\left(\frac{(q+q')^3}{r}\right).
\end{aligned}$$

In order to conclude, take $r = O\left((q+q')2^{n/3}\right)$. From Theorem 10, we have

$$\begin{aligned}
\mathbf{Adv}_{prp}^{\mathsf{QuEME}^E}(\mathcal{B}) &\leq O\left(\frac{r^2}{2^n}\right) + O\left(\frac{r^4}{2^{2n}}\right) + \alpha^E(r^2, r^2) \\
&= O\left(\frac{(q+q')^2}{2^{n/3}}\right) + O\left(\left(\frac{(q+q')^2}{2^{n/3}}\right)^2\right) + \alpha^E(r^2, r^2).
\end{aligned}$$

and from the choice of r, we have $O\left((q+q')^3/r\right) = O\left((q+q')^2/2^{n/3}\right)$. From there, we conclude

$$\begin{aligned}\mathbf{Adv}_{qprp}^{\mathsf{QuEME}^E}(\mathcal{A}) &\leq \mathbf{Adv}_{prp}^{\mathsf{QuEME}^E}(\mathcal{B}) + O\left(\frac{(q+q')^3}{r}\right) \\ &\leq O\left(\frac{(q+q')^2}{2^{n/3}}\right) + O\left(\left(\frac{(q+q')^2}{2^{n/3}}\right)^2\right) + \alpha^E(r^2, r^2).\end{aligned}$$

7.3 Discussion

Our proof is very generic and we can relate the quantum security to the classical security for any construction that start by encrypting the left and right halves of the input. The drawback of this strategy is that it seems to be far from tight. Indeed, when looking at our construction and the attack running with $O(2^n)$ queries it's not clear how to use quantum queries to improve this attack. We expect our construction to have much more than $n/6$ bits of quantum security, maybe its quantum security is actually n bits.

In order to improve these bounds, one natural path would be to look at Zhandry's quantum query recording technique. However, in our case, we need to consider random permutation and not random functions and this is notoriously hard, as some of the proposals for this turned out to be incorrect (see for instance [Unr21]). As this topic becomes more mature, we hope that this tool will be available for proving tight quantum security bounds for our construction.

8 Proposing a concrete instance: Double-AES

The aim of this section is to propose a concrete instance that would motivate cryptanalysis on these constructions. For this, we propose some variants that claim the same quantum security as the Saturnin block cipher [CDL+20], that was conceived with the objective of proposing resistance against quantum-attackers. In particular, we also claim that:

There exists no quantum attack in the single-key setting with $T^2/p < 2^{224}$, where T is the time complexity and p the success probability. We do not provide security against related-key superposition attacks (as is the case of all known block ciphers).

Therefore, in this section we propose a concrete construction based on AES-128 [DR00], showing how to use $256-$bit keys and domain extensions in the different blocks, and proposing several possible variants depending on how many AES rounds are considered in each block, and we announce their corresponding (quantum) security claims. We have chosen the AES-128 block cipher to profit from it's large amount of cryptanalysis on reduced round versions, that will simplify analyzing the instantiation of the new construction. We also briefly describe the best attacks we have found so far in these family of constructions. As we will see, these attacks motivated us to include the last MC transformation

on each block cipher call, but on the two ones from the final layer. A brief description of AES-128 and of the best known attacks on it can be found in section D.

The main aim of these instantiations is to motivate cryptanalysis and comparison with other constructions as well as further research.

8.1 How to Extend the keys?

Independently of the blocks ciphers used, we can decide how to choose the keys used in each block. As we expect that a $2n$ key can have the same security as a $4n$ one, we will choose the keys for E_3 and E_4 to be dependent on the ones from E_1 and E_2 for the sake of simplicity. For this, we propose to have a $2n$ key formed by $(k_1||k_2)$. These keys will be used as the key of the two blocks applied in the input of the construction: k_1 for E_1 and k_2 for E_2. We have to define now the keys k_3 and k_4 for the two output blocks E_3 and E_4. The idea is that, from the knowledge from one input key we should not be able to retrieve any information from any other key. For this, we propose to define:

$$k_3 = k_1 \oplus k_2 \text{ and } k_4 = k_1 \oplus (k_2 \lll 1).$$

Though more robust and complicated key-extensions could be proposed, like for instance $K_1, K_2, E_5^{K_2}(K_1), E_6^{K_1}(K_2)$, we believe ours has the advantage of being simpler and very efficient, and should have an equivalent security as long as the used block cipher is secure.

Property : guessing one full key (k_1 for instance) and x bits of any other key (*i.e.* k_2), allows to compute x and $x-1$ bits of the other two keys (k_3 and k_4 in the example). We could have choosen other simple configurations where the bits determined in the two last subkeys wouldn't belong mainly to the same bytes, but we believe this configuration is interesting, for the sake of simplicty of implementation and analysis.

8.2 Introducing a domain in AES for defining E_j

We present in Section D.1 the specifications of AES-128. For defining 4 additional different instances E_1 to E_4, we propose to use a different constant for each block. All the remaining parts stay the same and we only change the constant, as: with

$$rc_{i,j} = \begin{pmatrix} X^i \bmod X^8 + X^4 + X^3 + X + 1 \\ j \\ 0 \\ 0 \end{pmatrix}.$$

For the middle call to the block cipher we consider the original AES definition, where the input arriving as a parameter will have the role of the secret key.

8.3 Double-AES concrete proposals

Double-AES-10. A very conservative approach would be to consider a full AES encription inside each block.

Conjecture: the 7-round attacks cannot be exploited when using AES in our construction. We have tried to build attacks in our construction exploiting the best known attacks on 7-rounds when we consider the blocks used in the instantiation reduced to 7 rounds and we have not been able to do so. We conjecture that these attacks cannot apply when using our construction, and we propose therefore Double-AES-7, that instead of 10 rounds in each block only used 7. We also believe that it is a quite conservative approach, given the bext found attacks in the next section.

Variants including last MC: Double-AES-6-MC. We propose a variant where the last MC operation is consider in the block call that do not belong to the last layer (E_3 and E_4). We encourage the cryptanalysis of Double-AES-5-MC, for which we think an attack might exist, and we conjecture that Double-AES-6-MC should provide a similar security than Double-AES-10.

Best attacks. In section E of supplementary material we present the best attacks we have found. In order to reflect that different number of rounds can be considered per block, we call an attack on r_1-r_2-r_3 when it covers r_1 rounds for E_1 and E_2, r_2 rounds for E and r_3 rounds for E_3 and E_4. Our best attacks cover X-3-3 and X-2-X, without the last MC in E_1, E_2 and E.

8.4 Security Claims

The final construction has a 256-bit state and key. We provide an unique security claim, not distinguishing between classical and quantum attacks: we claim that our extended block cipher Double-AES, with versions Double-AES-10, Double-AES-7 and Double-AES-5-MC provide around 128 bits of security against any type of attacker. [4]

In addition, we claim that when plugged in secure modes, any attack that requires a collision on the state would require at least the generic complexity for generating a collision, that is, no attack significantly better than $\mathcal{T}^5 \times M_q = 2^{512}$ exists, where $\mathcal{T}$ represents the time complexity, and M_q the quantum memory (that includes the classical memory).

8.5 Discussion

As already said, we considered Saturnin [CDL+20], the first conceived block cipher aiming at security against all quantum attackers, as a model for our

[4] As all known block ciphers, we cannot provide security against superposition related-key attacks.

quantum security claims. Given that that construction also inherits a lot from the AES one, we consider the comparison interesting. The same goes for the cipher AES-256 [DR02]. In particular, if we consider the Double-AES-7, as the two upper and the two lowers can be done in parallel, the time of one encryption should take an equivalent of $3 \times 7 = 21$ AES rounds, while AES-256 takes 14 rounds for encrypting half of the state. Rijndael-256 [DR00] is a similar case than the AES-256, but with a state of 256 bits this time, as Double-AES. Nevertheless, this constructions, that was not standardized, has been much less studied by the community and benifits less from the cryptanalysis knowledge of the community.

In all the three cases, our construction has the advantage of being a generic way for doubling block ciphers, and therefore, we belive further study would be useful for understanding the properties of the underlying block cipher that can generate an attack or that can not.

We expect further cryptanalysis to show if we can reduce the number of rounds in Double-AES-7 or in Double-AES-5-MC further in order to gain in interest with respect to these previous constructions while staying secure.

9 Conclusion

In this paper we provide the first proposal of a generic way for extending both the key and the state size, with quantum security arguments and significant classical proofs. In addition we have proposed a new type of superposition attacks on the EME construction, an original distinguisher matching the bound of our construction, and a method considering simulations for supporting conjectures from proofs.

On concrete instances and cryptanalysis. Concrete instances of our construction combined with AES-128 have been proposed, with the aim of motivating its study, with a unified security claim regarding classical and quantum attackers. We believe the number of rounds of AES-128 considered in the building blocks can be reduced. We propose 7 rounds and 5 rounds if the final MC is not omitted in E_1, E_2 and E, where we claim equal security than with 10 rounds, but we believe an interesting question would be whether we manage to attack a variant with less rounds, as our best attack reaches X-3-3 rounds (so any number of rounds in the first layer, and three in the middle and final layers), or 2 rounds in the middle one, for any number of rounds in the upper and lower instances. Related-key attacks on AES might also have an interesting application in this case because of the middle block. We leave this as an interesting open question to break a variant with 5-5-5 rounds, or 6-6-6, that should be harder, as the structural distinguishers could be exploited. Another option would be to consider variants with less rounds in the middle block than in the four external ones. How low could we go? We know attacks exists with only 2 rounds in the middle, for any number of rounds in the external applications.

Open problems regarding quantum security arguments. It would be nice to find a quantum reduction proof similar to that in [HI19], based on a recording oracle.

However, there is no known way to lazy-sample a permutation or respond to inverse queries using a quantum recording oracle, though several research groups are working on this. Once a suitable proof technique using databases is discovered, it will be interesting to revisit the quantum security of EME and see what more we can say about it.

References

ABKM22. Gorjan Alagic, Chen Bai, Jonathan Katz, and Christian Majenz. Post-quantum security of the even-mansour cipher. In Orr Dunkelman and Stefan Dziembowski, editors, *Advances in Cryptology – EUROCRYPT 2022*, pages 458–487, Cham, 2022. Springer International Publishing.

Amb07. Andris Ambainis. Quantum walk algorithm for element distinctness. *SIAM Journal on Computing*, 37(1):210–239, 2007.

BDK+18. Achiya Bar-On, Orr Dunkelman, Nathan Keller, Eyal Ronen, and Adi Shamir. Improved key recovery attacks on reduced-round AES with practical data and memory complexities. In *CRYPTO 2018, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part II*, 2018.

BGL20. Zhenzhen Bao, Jian Guo, and Eik List. Extended truncated-differential distinguishers on round-reduced AES. *IACR Trans. Symmetric Cryptol.*, 2020.

BHN+19. Xavier Bonnetain, Akinori Hosoyamada, María Naya-Plasencia, Yu Sasaki, and André Schrottenloher. Quantum attacks without superposition queries: The offline simon's algorithm. In *ASIACRYPT 2019, Kobe, Japan, December 8-12, 2019, Proceedings, Part I*, 2019.

BHT98. Gilles Brassard, Peter Høyer, and Alain Tapp. Quantum cryptanalysis of hash and claw-free functions. In *LATIN '98: Theoretical Informatics, Third Latin American Symposium, Campinas, Brazil, April, 20-24, 1998, Proceedings*, 1998.

BLSNP18. Christina Boura, Virginie Lallemand, Valentin Suder, and María Naya-Plasencia. Making the Impossible Possible. *Journal of Cryptology*, 2018.

BN10. Alex Biryukov and Ivica Nikolic. Automatic search for related-key differential characteristics in byte-oriented block ciphers: Application to aes, camellia, khazad and others. In *Advances in Cryptology - EUROCRYPT 2010, Monaco / French Riviera, May 30 - June 3, 2010. Proceedings*, 2010.

CDL+20. Anne Canteaut, Sébastien Duval, Gaëtan Leurent, María Naya-Plasencia, Léo Perrin, Thomas Pornin, and André Schrottenloher. Saturnin: a suite of lightweight symmetric algorithms for post-quantum security. *IACR Trans. Symmetric Cryptol.*, 2020.

CLP15. Benoit Cogliati, Rodolphe Lampe, and Jacques Patarin. The indistinguishability of the xor of $$k$$permutations. In *Fast Software Encryption*, 2015.

CNS17. André Chailloux, María Naya-Plasencia, and André Schrottenloher. An efficient quantum collision search algorithm and implications on symmetric cryptography. In *ASIACRYPT 2017, Hong Kong, China, December 3-7, 2017, Proceedings, Part II*, 2017.

DFJ13. Patrick Derbez, Pierre-Alain Fouque, and Jérémy Jean. Improved key recovery attacks on reduced-round AES in the single-key setting. In *Advances in Cryptology - EUROCRYPT 2013, Athens, Greece, May 26-30, 2013. Proceedings*, 2013.

DKRS20. Orr Dunkelman, Nathan Keller, Eyal Ronen, and Adi Shamir. The retracing boomerang attack. In *EUROCRYPT 2020, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part I*, 2020.

DKS10. Orr Dunkelman, Nathan Keller, and Adi Shamir. Improved single-key attacks on 8-round AES-192 and AES-256. In *ASIACRYPT 2010 Singapore, December 5-9, 2010. Proceedings*, 2010.

DNS22. Avijit Dutta, Mridul Nandi, and Abishanka Saha. Proof of mirror theory for $\xi_{\max} = 2$. *IEEE Transactions on Information Theory*, 2022.

DR00. Joan Daemen and Vincent Rijmen. Rijndael for AES. In *The Third Advanced Encryption Standard Candidate Conference, April 13-14, 2000, New York, New York, USA*, 2000.

DR02. Joan Daemen and Vincent Rijmen. *The Design of Rijndael: AES - The Advanced Encryption Standard*. Information Security and Cryptography. Springer, 2002.

FJP13. Pierre-Alain Fouque, Jérémy Jean, and Thomas Peyrin. Structural evaluation of AES and chosen-key distinguisher of 9-round AES-128. In *Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I*, 2013.

FKL$^+$00. Niels Ferguson, John Kelsey, Stefan Lucks, Bruce Schneier, Michael Stay, David A. Wagner, and Doug Whiting. Improved cryptanalysis of rijndael. In Bruce Schneier, editor, *FSE 2000*, volume 1978 of *Lecture Notes in Computer Science*, pages 213–230. Springer, 2000.

Gil14. Henri Gilbert. A simplified representation of AES. In *Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I*, 2014.

GLR$^+$20. Lorenzo Grassi, Gregor Leander, Christian Rechberger, Cihangir Tezcan, and Friedrich Wiemer. Weak-key distinguishers for AES. In *SAC 2020, Halifax, NS, Canada (Virtual Event), October 21-23, 2020, Revised Selected Papers*, 2020.

GR20. Lorenzo Grassi and Christian Rechberger. Revisiting gilbert's known-key distinguisher. *Des. Codes Cryptogr.*, 2020.

Gro94. Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24*, 1994.

HI19. Akinori Hosoyamada and Tetsu Iwata. 4-round luby-rackoff construction is a qprp. In Steven D. Galbraith and Shiho Moriai, editors, *Advances in Cryptology – ASIACRYPT 2019*, 2019.

HR04. Shai Halevi and Phillip Rogaway. A parallelizable enciphering mode. In *Topics in Cryptology – CT-RSA 2004*, 2004.

IHM$^+$18. Gembu Ito, Akinori Hosoyamada, Ryutaroh Matsumoto, Yu Sasaki, and Tetsu Iwata. Quantum chosen-ciphertext attacks against feistel ciphers. Cryptology ePrint Archive, Report 2018/1193, 2018.

IHM$^+$19. Gembu Ito, Akinori Hosoyamada, Ryutaroh Matsumoto, Yu Sasaki, and Tetsu Iwata. Quantum chosen-ciphertext attacks against feistel ciphers. In *CT-RSA 2019, San Francisco, CA, USA, March 4-8, 2019, Proceedings*, volume 11405 of *Lecture Notes in Computer Science*, pages 391–411. Springer, 2019.

JST21. Joseph Jaeger, Fang Song, and Stefano Tessaro. Quantum key-length extension. In Kobbi Nissim and Brent Waters, editors, *Theory of Cryptography*, pages 209–239, Cham, 2021. Springer International Publishing.

KLLN16a. Marc Kaplan, Gaëtan Leurent, Anthony Leverrier, and María Naya-Plasencia. Breaking symmetric cryptosystems using quantum period finding. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, CA, USA, August 14-18, 2016, Proceedings, Part II*, 2016.

KLLN16b. Marc Kaplan, Gaëtan Leurent, Anthony Leverrier, and María Naya-Plasencia. Quantum differential and linear cryptanalysis. *IACR Trans. Symmetric Cryptol.*, 2016.

KM10. Hidenori Kuwakado and Masakatu Morii. Quantum distinguisher between the 3-round feistel cipher and the random permutation. In *IEEE International Symposium on Information Theory, ISIT 2010, June 13-18, 2010, Austin, Texas, USA, Proceedings*, 2010.

KM12. Hidenori Kuwakado and Masakatu Morii. Security on the quantum-type even-mansour cipher. In *Proceedings of the International Symposium on Information Theory and its Applications, ISITA 2012, Honolulu, HI, USA, October 28-31*, 2012.

KR01. Joe Kilian and Phillip Rogaway. How to protect des against exhaustive key search (an analysis of desx). *Journal of Cryptology*, 14(1):17–35, 2001.

LDKK08. Jiqiang Lu, Orr Dunkelman, Nathan Keller, and Jongsung Kim. New impossible differential attacks on AES. In *Progress in Cryptology - INDOCRYPT 2008, 9th International Conference on Cryptology in India, Kharagpur, India, December 14-17, 2008. Proceedings*, 2008.

LM17. Gregor Leander and Alexander May. Grover meets simon – quantumly attacking the fx-construction. In Tsuyoshi Takagi and Thomas Peyrin, editors, *Advances in Cryptology – ASIACRYPT 2017*, 2017.

LP21. Gaëtan Leurent and Clara Pernot. New representations of the AES key schedule. In *Advances in Cryptology - EUROCRYPT 2021 - Zagreb, Croatia, October 17-21, 2021, Proceedings, Part I*, 2021.

Pat03. Jacques Patarin. Luby-rackoff: 7 rounds are enough for $2^{n(1-\epsilon)}$) security. In *Advances in Cryptology - CRYPTO 2003*, 2003.

Pat08. Jacques Patarin. A proof of security in o(2n) for the xor of two random permutations. In *Information Theoretic Security*, 2008.

Pat09. Jacques Patarin. The "coefficients h" technique. In *Selected Areas in Cryptography*, 2009.

Pat10. Jacques Patarin. Security of balanced and unbalanced feistel schemes with linear non equalities. Cryptology ePrint Archive, Paper 2010/293, 2010.

RBH17. Sondre Rønjom, Navid Ghaedi Bardeh, and Tor Helleseth. Yoyo tricks with AES. In *ASIACRYPT 2017, Hong Kong, China, December 3-7, 2017, Proceedings, Part I*, 2017.

RSP21. Mostafizar Rahman, Dhiman Saha, and Goutam Paul. Boomeyong: Embedding yoyo within boomerang and its applications to key recovery attacks on AES and pholkos. *TOSC*, 2021.

Sim97. Daniel R Simon. On the power of quantum computation. *SIAM journal on computing*, 26(5):1474–1483, 1997.

Tun12. Michael Tunstall. Improved "partial sums"-based square attack on AES. In *SECRYPT 2012 - Proceedings of the International Conference on Security and Cryptography, Rome, Italy, 24-27 July, 2012, SECRYPT*, 2012.

Unr21. Dominique Unruh. Compressed permutation oracles (and the collision-resistance of sponge/sha3). Cryptology ePrint Archive, Report 2021/062, 2021.

Zha15. Mark Zhandry. A note on the quantum collision and set equality problems. *Quantum Info. Comput.*, 15(7?8):557?567, may 2015.

Auxiliary Supporting Material

A General related concepts

A.1 Common definitions and notations

We first define the prp advantage of an adversary, that tries to distinguish a function f from a random permutation, performing only queries to f.

Definition 2. *Let D be a distribution over the set $\{f : \{0,1\}^n \to \{0,1\}^n\}$ for a fixed $n \in \mathbb{N}$. We write*

$$\boldsymbol{Adv}_{prp}D(\mathcal{A}) = \left| \Pr_{f \leftarrow D}[\mathcal{A}^f(\cdot) = 1] - \Pr_{f \xleftarrow{\$} P}[\mathcal{A}^f(\cdot) = 1] \right|$$

where P is the set of permutations in $\{0,1\}^n$.

Depending on the context, $\mathcal{A}^f$ can be a classical or quantum algorithm that does classical or quantum queries to f. We also say that an adversary performing queries to f plays the IND-CPA game. We now define the strong prp advantage, where $\mathcal{A}$ can also perform queries to f^{-1}.

Definition 3. *Let D be a distribution over the set P_n where P_n is the set of permutations in $\{0,1\}^n$. We write*

$$\boldsymbol{Adv}^D_{sprp}(\mathcal{A}) = \left| \Pr_{f \leftarrow D}[\mathcal{A}^{f,f^{-1}}(\cdot) = 1] - \Pr_{f \xleftarrow{\$} P_n}[\mathcal{A}^{f,f^{-1}}(\cdot) = 1] \right|$$

In this case, we say $\mathcal{A}$ plays the IND-CCA game.

A.2 Mirror Theory

We'll use some results from a line of research that focuses on finding tight approximations on the number of solutions to systems of bi-variate equations, which goes by the name Mirror Theory [Pat03]. Some of these results have been proved [DNS22,CLP15,Pat08], while others are still conjectural [Pat10], but generally accepted in the community.

We'll use the Pochhammer falling factorial power notation

$$(a)_b := a(a-1)\dots(a-b+1).$$

Let N denote 2^n. Consider two sequences of n-bit variables $Y_1, \dots, Y_{q_1}$ and Z_1, $\dots, Z_{q_2}$, with $q_1 < N, q_2 < N$. For some $q < q_1 + q_2$ suppose there are q bi-variate equations of the form

$$Y_i \oplus Z_j = \delta_{ij}.$$

Then the *Mirror Theorem* states the following.

Theorem 5 (Mirror Theorem (conjectural)). *The number of solutions to the system described above such that Y_i's are all distinct and Z_i's are all distinct is at least*

$$\frac{(N)_{q_1}(N)_{q_2}}{N^q} \cdot (1-\epsilon),$$

where $\epsilon = O(q/N)$.

The intuition behind this is that the numerator in the above expression is the total number of solutions satisfying just the distinctness constraint, and any randomly chosen solution has a probability of about $1/N^q$ of satisfying all q bi-variate equations. However, the exact calculations needed to bound ϵ can be very complicated, and so far has only been completed for the special case with $q_1 = q_2 = q$ where each variable appears in exactly one equation. In spite of this, the result has been claimed earlier without complete proofs, first by Patarin and then by others citing him, and is generally accepted in the community. In this paper, we'll use the mirror theorem as a black box.

Tighter version. When many of the variables appear multiple times, the above bound can be made tighter. Consider the graph where a bi-variate equation involving Y_i and Z_j is represented by an edge between Y_i and Z_j. We assume none of the equations is redundant and the system of equations is consistent, so there are no cycles in this graph. Let $C^{(1)}, \ldots, C^{(t)}$ be the connected components, where $t = q_1 + q_2 - q$. For each $j \in [1..t]$ let $q_1^{(j)}$ (resp. $q_2^{(j)}$) be the number of Y_i's (resp. Z_i's) that appear in $C^{(j)}$. Finally, define the cumulative sums

$$Q_b^{(j)} = \sum_{i=1}^{j-1} q_b^{(i)}$$

for each $b \in \{1,2\}$ and each $j \in [1..t]$. Then the *Tight Mirror Theorem* states the following.

Theorem 6 (Tight Mirror Theorem (conjectural)). *The number of solutions to this system such that Y_i's are all distinct and Z_i's are all distinct is at least*

$$\frac{1}{N^q} \cdot \prod_{j=1}^{t} \left[\left(N - Q_1^{(j)}\right)^{q_1^{(j)}} \left(N - Q_2^{(j)}\right)^{q_2^{(j)}} \right] \cdot (1-\epsilon),$$

where $\epsilon = O(q/N)$.

The intuition behind this is an extension of the intuition behind the Mirror Theorem. As before we randomly choose a valid solution for the $\{Y_i\}$ and the $\{Z_i\}$, and it satisfies the equations with (roughly) a probability $1/N^q$. However, in this case, the key additional observation is that when choosing the valid solution, instead of ensuring distinctness among all $\{Y_i\}$ and all $\{Z_i\}$, we just need to ensure that there are no collisions between components; since our system of equations is consistent, for any solution that satisfies the equations, within-component distinctness is automatically ensured. Thus when choosing the $q_1^{(j)}$

Y_i's from the j-th component, we just choose them randomly from all the $N - Q_1^{(j)}$ unsampled values, and similarly for the Z_i's.

When $q_1^{(j)} = q_2^{(j)} = 1$ for each j, this reduces to the Mirror Theorem bound, and is strictly tighter in other cases, the gap increasing as the component sizes increase. We'll use this result as a black box too, to deal with cases where the component sizes are difficult to bound. In Sec. 5, we discuss some simulations we ran that suggest that this conjecture is not unreasonable.

A.3 H-Coefficient Technique

Suppose an adversary $\mathcal{A}$ is playing a distinguishing game against two oracles, one representing an ideal cryptographic object f, and the other representing an actual cryptographic construction $\mathcal{C}$. We'll use the standard terminology where the oracle representing f is called the *ideal oracle*, denoted $\mathcal{O}_0$, and the oracle representing $\mathcal{C}$ is called the *real oracle*, denoted $\mathcal{O}_1$. Formally, the challenger samples a secret bit b at the beginning, and gives $\mathcal{A}$ the oracle $\mathcal{O}_b$; $\mathcal{A}$ makes q queries $\{u^i \mid i \in [1..q]\}$ and receives the corresponding responses $v^i = \mathcal{O}_b(u^i)$; at the end of the game, $\mathcal{A}$ outputs a response bit b', and wins if $b' = b$. The scenario where $\mathcal{A}$ interacts with the ideal (resp. real) oracle will interchangeable be called the ideal (resp. real) world.

To bound the advantage of $\mathcal{A}$ we use the *Coefficient H Technique*. Let τ be the *transcript* of a q-query game played by $\mathcal{A}$, i.e., $\tau = \{(u^i, v^i) \mid i \in [1..q]\}$. In addition, when $\mathcal{A}$ interacts with $\mathcal{C}$, let τ^* denote the *internal transcript*, denoted $\tau^* = \{w^i \mid i \in [1..q]\}$; these are the intermediate variables computed when computing the responses to $\mathcal{A}$'s queries. Note that here u^i, v^i, w^i are of unspecified length—the first two will depend on the oracle interface and the last on the complexity of the internal computations.

Extended Transcripts. Let $\mathcal{S}$ be a sampler that takes τ as input and simulates an internal transcript τ^* when $\mathcal{A}$ interacts with f. We consider a modified game where at the end of the game τ^* is released to $\mathcal{A}$, which can be used to compute the final response bit; as this is extra information $\mathcal{A}$ is free not to use, this can only increase the advantage of $\mathcal{A}$, so any upper bound we derive here must hold for the original game as well. (τ, τ^*) together will be called the *extended transcript*. (We'll simply call it a transcript when the context is unambiguous.)

Good Transcripts. We'll define one or several *bad events* in the ideal world, based on the internal random coins of f and $\mathcal{S}$. (Note that for $\mathcal{A}$ we only consider deterministic adversaries.) In the first step of the proof, we'll need to show that for some suitably small ϵ_1 the probability that at least one bad event occurs is upper-bounded by ϵ_1. We'll call a transcript (τ, τ^*) *good* if it can be obtained in the ideal world without encountering any of the bad events.

Interpolation Probabilities. Given a transcript (τ, τ^*) and an oracle $\mathcal{O}_b$, we'll examine the *interpolation probability* of (τ, τ^*) in $\mathcal{O}_b$, denoted $\Pr_b[(\tau, \tau^*)]$. This is the probability that the internal random coins of $\mathcal{O}_b$ are *compatible* with

(τ, τ^*), i.e., (τ, τ^*) is obtained as the game transcript as long as $\mathcal{A}$ queries u^1 first and for each $i \in [2..q]$, on observing $v^1, \ldots, v^{i-1}$, next queries u^i. For an adversary who does not make these queries, the probability of obtaining (τ, τ^*) is trivially 0, and we ignore such adversaries when calculating the interpolation probability.

Ratio of Good Probabilities. For the second step of the proof, we'll need to find a suitably small ϵ_2 such that for any arbitrary good transcript (τ, τ^*), the ratio of $\Pr_1[(\tau, \tau^*)]$ and $\Pr_0[(\tau, \tau^*)]$ is lower-bounded by $1 - \epsilon_2$. This ratio is often simply referred to as the *ratio of good probabilities*, whereas the probability of at least one bad event occurring (applicable only for the ideal world) is referred to as the *bad probability*.

Once we have the stated bounds on both the bad probability and the ratio of good probabilities, the main result of the H-Coefficient Technique [Pat09] tells us that the distinguishing advantage of $\mathcal{A}$ between $\mathcal{O}_0$ and $\mathcal{O}_1$ cannot exceed $\epsilon_1 + \epsilon_2$. Below we give the formal statement of the theorem we will use.

Theorem 7 (H-Coefficient Technique). *Suppose for an adversary $\mathcal{A}$ playing a q-query distinguishing game between an ideal object f and a real construction $\mathcal{C}$, we can define bad events and find ϵ_1 and ϵ_2 such that the probability of a bad event in a game against f is at most ϵ_1, and the ratio of good probabilities while interacting with $\mathcal{C}$ and f for any fixed good transcript τ is at least $1 - \epsilon_2$. Then we have*

$$\boldsymbol{Adv}_f^{\mathcal{C}}(\mathcal{A}) \leq \epsilon_1 + \epsilon_2.$$

A.4 Quantum Computing

We present here some quantum algorithms that we will use in this paper. Performing a quantum query to a function f means applying the unitary $O_f : |x\rangle |y\rangle \to |x\rangle |y + f(x)\rangle$. If f is efficiently computable classically then O_f is efficiently computable quantumly.

Simon's Algorithm. A function $g : \{0,1\}^n \to \{0,1\}^m$ is said to have a period s when $g(x) = g(y)$ iff. $x = y$ or $x = y \oplus s$. If g is efficiently computable then Simon's algorithm [Sim97] is able to recover s in time $poly(n)$. A relaxed version of the Simon's Algorithm can be used to detect the presence of a period without recovering it [IHM+18, Sec. 4].

It is also possible to only evaluate g on a subspace as long as the subspace admits s as a period, i.e. if x is the subspace, $x \oplus s$ is also in the subspace.

Grover's search. Given an efficiently computable function $f : \{0,1\}^n \to \{0,1\}$, Grover's search algorithm [Gro94] finds an element x st. $f(x) = 1$ (if such an element exists) in time $O(2^{n/2})$.

BHT algorithm. Given a random function $f : \{0,1\}^n \to \{0,1\}^n$, the BHT algorithm [BHT98] finds a collision, *i.e.* a pair $x, y \neq x$ st. $f(x) = f(y)$ with $O(2^{n/3})$ quantum queries to f. If f is efficiently computable then the quantum running time is also $O(2^{n/3})$, given access to quantum RAM operations.

It is possible to modify the procedure to get the uniform superposition of collisions instead of a random one.

Models for quantum attacks. Different scenarios are possible: The Q1 setting allows the attacker to use a quantum computer but he can make only classical queries to the black-box primitives. The Q2 setting (or superposition attacks) allows the attacker to make superposition queries to the black-box primitives. Many of the published quantum attacks rely on this model [KLLN16b], and as security in this setting represents security in any other intermediate scenarios, we will aim for resistance to these kind of attacks for our construction.

B Reasoning of Theorem 1

We start by recalling the fundamentals of Simon's algorithm [Sim97], starting with the Hadamard gate.

Hadamard gate. the Hadamard gate (H) maps $|0\rangle$ to $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and $|1\rangle$ to $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$.

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad \Big| \; |b\rangle \text{ —— } \boxed{H} \text{ —— } \frac{1}{\sqrt{2}}(|0\rangle + (-1)^b |1\rangle)$$

Simon's Algorithm. A function $g : \{0,1\}^n \to \{0,1\}^m$ is said to have a period s when $g(x) = g(y)$ iff. $x = y$ or $x = y \oplus s$. If g is efficiently computable then Simon's algorithm is able to recover s in time $poly(n)$.

Simon's algorithm consists in applying Simon's routine (see Algorithm 3) $l = O(n)$ times, thus getting $(y_1, ..., y_l)$ and solving the linear system with unknown s

$$\begin{cases} y_1 \cdot s = 0 \\ \quad \vdots \\ y_l \cdot s = 0 \end{cases}$$

For further explanations, we name $RANK$ a circuit that takes $|y_1\rangle ... |y_l\rangle |b\rangle$ and flips b iff the previous system admits a solution other than 0.

This version Simon's algorithm have as a premise that g is a two-to-one function. Luckily, has been studied for random functions that admits a period.

Theorem 8 (Theorem 2 in [KLLN16a]). *Suppose that $g : \{0,1\}^n \to X$ has a period s, i.e. $g(x \oplus s) = g(x)$ for all $x \in \{0,1\}^n$ and satisfies*

$$\max_{t \notin \{0,s\}} \mathbb{P}(g(x \oplus t) = g(x)) \leq \frac{1}{2}$$

Algorithm 3 Description of Simon's routine

Input: *superposition* oracle access to g
Output: a vector y such that $y \cdot s = 0$

1: Start with the state $|0^n\rangle |0^m\rangle$
2: Apply the Hadamard gate on all qubits of the first register, obtaining the state $\frac{1}{2^{n/2}} \sum_{x\in\{0,1\}^n} |x\rangle \otimes |0^m\rangle$
3: Apply the oracle $O_g : |x\rangle |y\rangle \mapsto |x\rangle |y \oplus g(x)\rangle$ to the state, obtaining $\sum_{x\in\{0,1\}^n} \frac{1}{2^{n/2}} |x\rangle |g(x)\rangle$
4: Measure the second register and get a value $c = g(x_0)$ for a unknown x_0. By the premise, we get the state $\frac{1}{\sqrt{2}} (|x_0\rangle + |x_0 \oplus s\rangle)$.
5: Apply the Hadamard gate on all qubits and we get the state

$$\frac{1}{\sqrt{2^{n+1}}} \sum_{y\in\{0,1\}^n} \left((-1)^{x_0\cdot y} + (-1)^{(x_0\oplus s)\cdot y}\right) |y\rangle .$$

This simplifies to

$$\frac{1}{\sqrt{2^{n+1}}} \sum_{y\in\{0,1\}^n} (-1)^{x_0\cdot y} \underbrace{(1 + (-1)^{s\cdot y})}_{0 \text{ if } y\cdot s=1} |y\rangle .$$

6: Measure the state and get a uniformly random y such that $y \cdot s = 0$.
7: **return** y

When we apply Simon's algorithm to g with cn calls to the routine, it returns s with a probability at least $1 - 2^n \cdot (3/4)^{cn}$. It is running in cn queries to g and time cn^2.

An important remark to Simon's routine (and to Simon's algorithm by consequence) is that we do not need g if we have access to cn superposition states $|\phi_g\rangle = \sum_{x\in\{0,1\}^n} \frac{1}{2^{n/2}} |x\rangle |g(x)\rangle$. Moreover, we do not need the superposition to include all x in $\{0,1\}^n$, it is possible to restrict g to a subset A as long as the subset admits s as a period i.e., $x \in A$ iff. $x \oplus s \in A$, and A does not make an artificial period appear (by restricting on elements such that $g(x \oplus t) = g(x)$ for a certain t). This can be taken to an extreme where $g = 0$ but A has the information of the period.

Corollary 1. *Suppose that $g : A \subseteq \{0,1\}^n \to X$ has a period s, i.e. $x \oplus s \in A$ $g(x \oplus s) = g(x)$ for all $x \in A$ and satisfies*

$$\max_{t\notin\{0,s\}} \mathbb{P}_{x\in A}(\tilde{g}(x \oplus t) = g(x)) \leq \frac{1}{2}$$

$$\text{where } \tilde{g}(x) = \begin{cases} g(x) & \text{if } x \in A \\ \perp & \text{otherwise} \end{cases}$$

When we apply Simon's algorithm to cn copies of $|\phi_g\rangle = \sum_{x\in A} \frac{1}{\sqrt{|A|}} |x\rangle |g(x)\rangle$, it returns s with a probability at least $1 - 2^n \cdot (3/4)^{cn}$. It is running in time cn^2.

Then, because the properties of Simon's algorithm did not change because of the input restriction on g, we can apply the offline Simon's algorithm [BHN+19] ideas.

Theorem 9 (Proposition 2 in [BHN+19]). *Suppose that $m = O(n)$, $f : \{0,1\}^m \times \{0,1\}^n \to \{0,1\}^l$ a public function and $g : \{0,1\}^n \to \{0,1\}^l$ on which we only get some databases $|\phi_g\rangle$ and there is a unique i_0 such that $f_{i_0} \oplus g$ has a period s and*

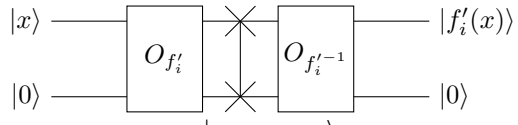
$$\max_{i,t \notin \{0,1\}^m \times \{0\} \cup \{i_0, s\}} \mathbb{P}((f_i \oplus g)(x \oplus t) = (f_i \oplus g)(x)) \leq \frac{1}{2}$$

When we apply the offline Simon's algorithm, with $O(n)$ databases $|\phi_g\rangle$, it returns i_0 with a probability in $\Theta(1)$. It is running in time $O(n^3 2^{m/2})$.

Algorithm 4 Description of the Offline-Simon algorithm

Input: *superposition* oracle access to f and $O(n)$ databases $|\phi_g\rangle$
Output: i_0
1: **Grover search** on i with $O(2^{m/2})$ turns using the following oracle :
2: Compute $O(n)$ copies of $|\phi_{f_i \oplus g}\rangle = O_{f_i} |\phi_g\rangle$
3: Apply the Hadamard gate on all qubits of the first registers of $|\phi_{f_i \oplus g}\rangle$, obtaining $O(n)$ states y
▷ $y \cdot s = 0$ if $i = i_0$ and random otherwise
▷ This is Simon's routine.
4: Apply the $RANK$ circuit on the states y
▷ the flip occurs iff $i = 0$
5: Uncompute the Hadamard gates and the O_{f_i} to recover the databases $|\phi_g\rangle$
6: **EndGrover**
7: Measure and **return** i

This technique relies on the equality $|\phi_{f_i \oplus g}\rangle = O_{f_i} |\phi_g\rangle$ for preparing and recovering the databases $|\phi_g\rangle$. In our case, instead of $f_{i_0} \oplus g$ being periodic, we look for $f_{i_0} \oplus g \circ f'_{i_0}$ being periodic with f'_i as public permutations. We build the operator $IN_{f'_i} : |x\rangle \mapsto |f'_i(x)\rangle$ using ancila qubits and the following circuit:



We can compute $\left|\phi_{f_{i_0} \oplus g \circ f'_{i_0}}\right\rangle = O_{f_i} \circ (IN_{f'^{-1}_i} \otimes I) |\phi_g\rangle$.

C Additional Results on Classical Proofs

C.1 IND-CPA security proof of n-bit Security using Mirror Theory

In this subsection we prove the following result.

Theorem 10. *For any classical adversary $\mathcal{A}$ playing a q-query IND-CPA game against QuEME^{π}, under the assumption that the Mirror Theorem (Sec. A.2) holds, we have*

$$\boldsymbol{Adv}_{prp}^{\mathsf{QuEME}^{\pi}}(\mathcal{A}) = O\left(\frac{q}{2^n}\right).$$

For any classical adversary $\mathcal{A}'$ playing a q-query IND-CPA game against QuEME^{E} with q' offline queries to E, we have

$$\boldsymbol{Adv}_{prp}^{\mathsf{QuEME}^{E}}(\mathcal{A}') \leq O\left(\frac{q}{2^n}\right) + O\left(\frac{qq'}{2^{2n}}\right) + \alpha^{E}(q, q').$$

Sampler of Internal Transcripts. Here we'll define the sampler $\mathscr{S}$ which takes $\tau = \{(L^i, R^i), (S^i, T^i) \mid i \in [1..q]\}$ as input and samples a $\tau^* = \{(\widehat{L}^i, \widehat{R}^i, \widehat{S}^i, \widehat{T}^i) \mid i \in [1..q]\}$. The sampling proceeds as follows:

1. $\mathscr{S}$ first samples two independent n-bit random permutations π_1^* and π_2^*. Then it sets $\widehat{L}^i \leftarrow \pi_1^*(L^i)$, $\widehat{R}^i \leftarrow \pi_2^*(R^i)$, and $X^i \leftarrow \widehat{L}^i \oplus \widehat{R}^i$ for each $i \in [1..q]$.
2. Let Γ be the set of all $2q$-sequences $(\widehat{S}^1, \widehat{T}^1, \ldots, \widehat{S}^q, \widehat{T}^q)$ satisfying the following conditions:
   - $(\forall i, i' \in [1..q])\widehat{S}^i = \widehat{S}^{i'} \iff S^i = S^{i'}$;
   - $(\forall i, i' \in [1..q])\widehat{T}^i = \widehat{T}^{i'} \iff T^i = T^{i'}$;
   - $(\forall i \in [1..q])\widehat{S}^i \oplus \widehat{T}^i = \widehat{L}^i \oplus \widehat{R}^i$.

   Then $\mathscr{S}$ samples $(\widehat{S}^1, \widehat{T}^1, \ldots, \widehat{S}^q, \widehat{T}^q)$ uniformly at random from the set Γ.

Bad Events. We define the following bad events on the random coins of f and $\mathscr{S}$:

bad_0: For some distinct $i, i' \in [1..q]$, $(S^i, T^i) = (S^{i'}, T^{i'})$;
bad_1: For some distinct $i, i' \in [1..q]$

$$X^i = X^{i'} \wedge \left(S^i = S^{i'} \vee T^i = T^{i'}\right).$$

bad_2: For a path P of even length ≥ 4 in H,

$$\bigoplus_{i \in P} X^i = 0;$$

bad_3: There is a cycle in H.

In $\mathsf{bad}_1, \mathsf{bad}_2$ and bad_3 we assume bad_0 has not happened.

Bad Probabilities. We recall that the bad events (and hence bad probabilities) are only defined in the ideal world. bad_0 involves a random collision over $2n$ bits, with a choice of the two indices i and i'. Thus,

$$\Pr_0[\mathsf{bad}_0] \leq \frac{\binom{q}{2}}{N^2} \leq \frac{q^2}{N^2}. \tag{8}$$

We now bound the probability of bad_1. Let's fix a pair of indices $i, i' \neq i$. Because we are in the ideal case, $S^i, S^{i'}, T^i, T^{i'}, X^i, X^{i'}$ are uniformly random strings in $[N]$. Therefore,

$$\Pr_0[\mathsf{bad}_1] \leq \frac{q(q-1)\cdot 2}{N^2} \leq \frac{2q^2}{N^2}. \tag{9}$$

For a path P of even length $2m \geq 4$, we can argue similarly that the $2m$ values $X^i, i \in P$ are all mutually independent, and sum to 0 with a probability of $1/N$. The event involves $2m-1$ further collisions (for forming the path P), choice of $2m$ indices, and a choice whether the path begins from an S-node or a T-node. Therefore,

$$\begin{aligned}\Pr_0[\mathsf{bad}_2] &\leq \sum_{m\geq 2} \frac{q(q-1)\ldots(q-2m+1)\cdot 2}{N^{2m}} \\ &\leq \sum_{m\geq 2} \frac{2q^{2m}}{N^{2m}} = \frac{2q^2}{N^2}\sum_{m\geq 2}\left(\frac{q^2}{N^2}\right)^{m-1} \leq \frac{2q^2}{N^2}\sum_{m\geq 2}\left(\frac{1}{2}\right)^{m-1} \leq \frac{2q^2}{N^2}. \end{aligned} \tag{10}$$

Finally, a cycle of length $2m$ (with $m \geq 2$, since a cycle of length 2 would imply bad_0) will need $2m$ collisions for the cycle and give a choice of $2m$ indices and a choice of whether the first node is an S or a T; since the choice of this 'first node' is arbitrary, we divide the total count by m. This gives

$$\begin{aligned}\Pr_0[\mathsf{bad}_3] &\leq \sum_{m\geq 2} \frac{q(q-1)\ldots(q-2m+1)\cdot 2}{N^{2m}\cdot m} \\ &\leq \sum_{m\geq 2} \frac{2q^{2m}}{mN^{2m}} = \frac{q^2}{N^2}\sum_{m\geq 2}\left(\frac{q^2}{N^2}\right)^{m-1} \leq \frac{q^2}{N^2}\sum_{m\geq 2}\left(\frac{1}{2}\right)^{m-1} \leq \frac{q^2}{N^2}. \end{aligned} \tag{11}$$

From Eqs. 8-11 it follows that the probability that at least one of the bad events occurs is upper-bounded by

$$\epsilon_1 = \frac{6q^2}{N^2}. \tag{12}$$

Ratio of Good Probabilities. Next we turn to the second step of using the Coefficient H Technique: lower-bounding the ratio of good probabilities. Suppose (τ, τ^*) is a good transcript. Recall that q_1, q_2, q_3, q_4 are the number of distinct values respectively of L^i, R^i, S^i, T^i in τ. Further suppose that in τ^*, there are r distinct values of X^i, with the number of queries they appear in being $t_1, \ldots, t_r$, where $t_1 + \ldots + t_r = q$.

Real World. In the real world, for each $j \in [1..4]$, the probability that π_j is compatible with (τ, τ^*) is $1/(N)_{q_j}$, and the probability that $\widetilde{\pi}$ is compatible with (τ, τ^*) is $1/[(N)_{t_1}\ldots(N)_{t_r}]$. Thus,

$$\Pr_1[(\tau, \tau^*)] = \frac{1}{(N)_{q_1}\ldots(N)_{q_4}(N)_{t_1}\ldots(N)_{t_r}}. \tag{13}$$

Ideal World. In the ideal world, the probability that f is compatible with τ is $1/(N^2)^q$. The probabilities that π_1^* and π_2^* are compatible with τ^* are respectively $1/(N)_{q_1}$ and $1/(N)_{q_2}$. For the second step, the probability that the $(\widehat{S}^1, \widehat{T}^1, \ldots, \widehat{S}^q, \widehat{T}^q)$ comes from Γ is $1/|\Gamma|$. Thus,

$$\Pr_0[(\tau, \tau^*)] = \frac{1}{(N^2)^q (N)_{q_1} (N)_{q_2} |\Gamma|}. \tag{14}$$

Eqs. 13 and 14 give the ratio

$$\rho := \frac{\Pr_1[(\tau, \tau^*)]}{\Pr_0[(\tau, \tau^*)]} = \frac{(N^2)^q |\Gamma|}{(N)_{q_3} (N)_{q_4} (N)_{t_1} \ldots (N)_{t_r}}. \tag{15}$$

Since $(N)_{t_j} < N^{t_j}$, and $t_1 + \ldots + t_r = q$, we have $(N)_{t_1} \ldots (N)_{t_r} < N^q$. Plugging this in Eq. 15 gives

$$\rho \geq \frac{N^q |\Gamma|}{(N)_{q_3} (N)_{q_4}}. \tag{16}$$

Using Mirror Theory as a Black-box. We recall that in the ideal world, having chosen all $\widehat{L}^i$ and $\widehat{R}^i$, we need to choose $\widehat{S}^i$ and $\widehat{T}^i$ such that, for each $i, j \in [1..q]$:

- $\widehat{S}^i = \widehat{S}^j \iff S^i = S^j$;
- $\widehat{T}^i = \widehat{T}^j \iff T^i = T^j$;

and for each $i \in [1..q]$, $\widehat{S}^i \oplus \widehat{T}^i = X^i$. Then we can formulate our problem as one of Mirror Theory as follows: we need to find $\widehat{S}_1, \ldots, \widehat{S}_{q_3}$, all distinct, and $\widehat{T}_1$, $\ldots, \widehat{T}_{q_4}$, all distinct, satisfying q bi-variate equations of the form $\widehat{S}_i + \widehat{T}_j = \delta_{ij}$. Since this is a good transcript, we know that none of the bad events happened, making this system of equations cycle-free and consistent. Then from the Mirror Theorem (Theorem 5) we see that

$$|\Gamma| \geq \frac{(N)_{q_3} (N)_{q_4}}{N^q} \cdot (1 - \epsilon_2), \tag{17}$$

where $\epsilon_2 = O(q/N)$.

Putting the bound of Eq. 17 in Eq. 16 gives the desired bound

$$\rho \geq 1 - \epsilon_2, \tag{18}$$

thus completing the proof.

C.2 IND-CCA security proof of n-bit Security using Mirror Theory

In this subsection we prove Theorem 2. For simplicity we assume there are no cycle queries; the case when there are cycle queries can be similarly handled. Let $\widehat{L}_1, \ldots, \widehat{L}_{q_1}$, $\widehat{R}_1, \ldots, \widehat{R}_{q_2}$, $\widehat{S}_1, \ldots, \widehat{S}_{q_3}$ and $\widehat{T}_1, \ldots, \widehat{T}_{q_4}$ be the distinct values we need to choose for each permutation. Let H_τ be the graph analogous to G_τ, but for x and y, i.e., two vertices i and i' are adjacent in H_τ if $L^i = L^{i'}$ or $R^i = R^{i'}$; the edge (i, i') is blue for the first condition and red for the second. We change the sampling mechanism in the ideal world as follows:

1. $\mathscr{S}$ first samples a $X^1, \ldots, X^q$ such that on any (non-empty) path P of even length in G or H,
$$\bigoplus_{i \in P} X^i \neq 0.$$
Let Λ denote the set of all $(X^1, \ldots, X^q)$ satisfying this condition.
2. Next, $\mathscr{S}$ samples $\widehat{L}_1, \ldots, \widehat{L}_{q_1}$, all distinct, $\widehat{R}_1, \ldots, \widehat{R}_{q_2}$, all distinct, subject to q bi-variate equations of the form $\widehat{L}_i \oplus \widehat{R}_j = \delta_{ij}^G$. Let Γ^G denote the set of all solutions to this system.
3. Finally $\mathscr{S}$ samples $\widehat{S}_1, \ldots, \widehat{S}_{q_3}$, all distinct, and $\widehat{T}_1, \ldots, \widehat{T}_{q_4}$, all distinct, subject to q bi-variate equations of the form $\widehat{S}_i \oplus \widehat{T}_j = \delta_{ij}^H$. Let Γ^H denote the set of all solutions to this system.

Since we have assumed there are no cycles in G and H, we have $q_1 + q_2 - \alpha = q_3 + q_4 - \beta = q$.

Then the Tight Mirror Theorem (Theorem 6) tells us that

$$|\Gamma^G| \geq \prod_{j=1}^{\alpha} \left[\left(N - Q_1^{(j)}\right)^{q_1^{(j)}} \left(N - Q_2^{(j)}\right)^{q_2^{(j)}} \right] \cdot \frac{1}{N^q} \cdot (1 - \epsilon_2), \tag{19}$$

$$|\Gamma^H| \geq \prod_{j=1}^{\beta} \left[\left(N - Q_3^{(j)}\right)^{q_3^{(j)}} \left(N - Q_4^{(j)}\right)^{q_4^{(j)}} \right] \cdot \frac{1}{N^q} \cdot (1 - \epsilon_3), \tag{20}$$

where ϵ_1 and ϵ_2 are both $O(q/N)$.

Similarly, we can show that

$$|\Lambda| \geq N^q \left[\prod_{j=1}^{\alpha} \frac{(N)_{q_1^{(j)}} (N)_{q_2^{(j)}}}{N^{q_1^{(j)} + q_2^{(j)}}} \right] \left[\prod_{j=1}^{\beta} \frac{(N)_{q_3^{(j)}} (N)_{q_4^{(j)}}}{N^{q_3^{(j)} + q_4^{(j)}}} \right] (1 - \epsilon_3). \tag{21}$$

We observe that

$$\begin{aligned} \left(N - Q_1^{(j)}\right)^{q_1^{(j)}} (N)_{q_1^{(j)}} &= \prod_{k=0}^{q_1^{(j)}-1} \left(N - Q_1^{(j)}\right) (N - k) \\ &\geq \prod_{k=0}^{q_1^{(j)}-1} \left(N - Q_1^{(j)} - k\right) N \\ &= \left(N - Q_1^{(j)}\right)_{q_1^{(j)}} N^{q_1^{(j)}}, \end{aligned} \tag{22}$$

so that

$$\prod_{j=1}^{\alpha} \left(N - Q_1^{(j)}\right)^{q_1^{(j)}} (N)_{q_1^{(j)}} \geq (N)_{q_1} N^{q_1}. \tag{23}$$

We can show a similar inequality for q_2, q_3, and q_4.

Thus,

$$\frac{|\Gamma^G||\Gamma^H||\Lambda|}{(N)_{q_1}(N)_{q_2}(N)_{q_3}(N)_{q_4}} \geq \frac{1}{N^q}(1-\epsilon_2-\epsilon_3-\epsilon_4). \tag{24}$$

Since

$$\Pr_0[(\tau,\tau^*)] = \frac{1}{(N^2)^q|\Gamma^G||\Gamma^H||\Lambda|}, \tag{25}$$

we have

$$\rho \geq 1-\epsilon_2-\epsilon_3-\epsilon_4, \tag{26}$$

which completes the proof.

C.3 IND-CPA security proof of (2n/3)-bit Security

In this subsection we prove the following result.

Theorem 11. *For any classical adversary $\mathcal{A}$ playing a q-query IND-CPA game against QuEME^{π}, we have*

$$\boldsymbol{Adv}_{prp}^{\mathsf{QuEME}^{\pi}}(\mathcal{A}) = O\left(\frac{q^3}{2^{2n}}\right).$$

For any classical adversary $\mathcal{A}'$ playing a q-query IND-CPA game against QuEME^{E} with q' offline queries to E, we have

$$\boldsymbol{Adv}_{prp}^{\mathsf{QuEME}^{E}}(\mathcal{A}') \leq O\left(\frac{q^3}{2^{2n}}\right) + O\left(\frac{qq'}{2^{2n}}\right) + \alpha^E(q,q').$$

For this proof we change the second step of the sampling procedure of Sec. C.1. We observe that for each $j \in [1..\ell]$, once a value is assigned to $\widehat{S}^{i_j}$, it induces a value on $\widehat{S}^i, \widehat{T}^i$ for each $i \in C_j$, according to the following rules:

- if $\widehat{S}^i$ is already set, $\widehat{T}^i \longleftarrow \widehat{L}^i \oplus \widehat{R}^i \oplus \widehat{S}^i$;
- if $\widehat{T}^i$ is already set, $\widehat{S}^i \longleftarrow \widehat{L}^i \oplus \widehat{R}^i \oplus \widehat{T}^i$;
- if $S^i = S^{i'}$ and $\widehat{S}^{i'}$ is already set, $\widehat{S}^i \longleftarrow \widehat{S}^{i'}$;
- if $T^i = T^{i'}$ and $\widehat{T}^{i'}$ is already set, $\widehat{T}^i \longleftarrow \widehat{T}^{i'}$.

Since C_j is connected, all nodes in C_j are guaranteed to be covered in this manner. We refer to this as *extending* $\widehat{S}^{i_j}$ to all of C_j.

$\mathscr{S}$ uses two (initialised as empty) sets $D_{\widehat{S}}$ and $D_{\widehat{T}}$, to tabulate the sampled values of $\widehat{S}^i$ and $\widehat{T}^i$ respectively; every time a value is assigned to some $\widehat{S}^i$ or $\widehat{T}^i$, it is added to the corresponding table. For a subgraph C of G_τ (which can be seen as a subset of the queries), $D_{\widehat{S}|C}$ and $D_{\widehat{T}|C}$ will respectively denote $D_{\widehat{S}}$ and $D_{\widehat{T}}$ restricted to C. The connected components will be considered listed by decreasing size.

Step 1 is the same as in Sec. C.1. In Step 2 $\mathscr{S}$ samples $\widehat{S}^1$ uniformly at random from $\{0,1\}^n$ and extends $\widehat{S}^1$ to all of C^1, and for each $j \in [2..\ell]$ $\mathscr{S}$ samples over C_j as follows:

1. it computes the *banned set* B_j defined with the following condition: if and only if $\widehat{S}^{i_j}$ is assigned a value in B_j and extended to all of C_j, at least one of the sets $D_{\widehat{S}|C_j} \cap D_{\widehat{S}}$ and $D_{\widehat{T}|C_j} \cap D_{\widehat{T}}$ will be non-empty; [5]
2. it samples $\widehat{S}^{i_j}$ from $\{0,1\}^n \setminus B_j$ and extends it to all of C_j. [6]

Once $\mathscr{S}$ has completed Step 3 for C_ℓ, the sampling of τ^* is complete. We also include the additional bad event defined as follows:

$\mathsf{bad_4}$: There are is a path of length ≥ 3 in G_τ.

This happens with $O(q^3/N^2)$ probability. When $\mathsf{bad_5}$ has not happened, G_τ has no component of size ≥ 2. Thus, for each $j \in [2..\ell]$, $|B_j| \leq 2(j-1)$.

Good probability in ideal world. For each $j \in [1..\ell]$, the probability that the random sampling in the modified second step of $\mathscr{S}$ correctly outputs the $\widehat{S}^{i_j}$ is $1/(N - |B_j|)$. (We take B_1 to be the empty set.) Thus,

$$\Pr_0[(\tau, \tau^*)] = \frac{1}{(N^2)^q (N)_{q_1} (N)_{q_2} N(N-|B_2|)\dots(N-|B_\ell|)}. \tag{27}$$

Eqs. 13 and 27 give the ratio

$$\rho = \frac{(N^2)^q N(N-|B_2|)\dots(N-|B_\ell|)}{(N)_{q_3}(N)_{q_4}(N)_{t_1}\dots(N)_{t_r}}. \tag{28}$$

Like in the derivation of Eq. 16, we plug in $(N)_{t_1}\dots(N)_{t_r} < N^q$ in Eq. 28 to get

$$\rho \geq \frac{N^q N(N-|B_2|)\dots(N-|B_\ell|)}{(N)_{q_3}(N)_{q_4}}. \tag{29}$$

Since $q \geq \max(q_3, q_4)$, we have $\ell \leq \min(q_3, q_4)$. Thus, we can rewrite Eq. 29 as

$$\begin{aligned}
\rho &\geq \frac{N^q N(N-2)\dots(N-(2\ell-2))}{(N)_{q_3}(N)_{q_4}} \\
&\geq \frac{N^{q-\ell}}{(N-\ell)_{q_3-\ell}(N-\ell)_{q_4-\ell}} \cdot \frac{N^\ell N(N-2)\dots(N-(2\ell-2))}{(N)_\ell (N)_\ell} \\
&\geq \prod_{j=1}^{\ell} \frac{N(N-2(j-1))}{(N-(j-1))^2} \\
&= \prod_{j=1}^{\ell} \frac{N^2 - 2(N)(j-1)}{N^2 - 2(N)(j-1) + (j-1)^2} \\
&= \prod_{j=1}^{\ell} \left(1 - \frac{(j-1)^2}{N^2 - 2(N)(j-1) + (j-1)^2}\right)
\end{aligned}$$

[5] To avoid more complicated notation we assume here that the newly assigned values are kept in $D_{\widehat{S}|C_j}$ and $D_{\widehat{T}|C_j}$ but not yet added to $D_{\widehat{S}}$ and $D_{\widehat{T}}$.

[6] It is easy to see that $D_{\widehat{S}} \subseteq B_j$.

$$\geq \prod_{j=1}^{\ell} \left(1 - \frac{2(j-1)^2}{N^2}\right)$$
$$\geq 1 - \sum_{j=1}^{\ell} \frac{2(j-1)^2}{N^2} \geq 1 - \frac{\ell^3}{N^2}, \tag{30}$$

which completes the proof with $\epsilon_2 = \ell^3/N^2$.

D AES specification and attacks

D.1 AES: specifications and discussion

AES is the most popular and widely used symmetric blockcipher. Its internal state size is of 128 bits. Three variants are standardized, with a 128, a 192 or a 256-bit keys, each one with 10, 12 or 14 rounds respectively. AES-256 would then provide, regarding Grover a key-recovery security of 128 bits, but when used in most common modes, collisions on internal states would provide other kind of attacks, potentially better than classical attacks under some assumptions for the attackers. We provide here a basic description of AES-128 and we point to [DR00] for more details.

AES State The state of is composed of elements of $\mathbb{F}_{256}$ organized in a 4×4 matrix :

$$\begin{bmatrix} \alpha_0 & \alpha_4 & \alpha_8 & \alpha_{12} \\ \alpha_1 & \alpha_5 & \alpha_9 & \alpha_{13} \\ \alpha_2 & \alpha_6 & \alpha_{10} & \alpha_{14} \\ \alpha_3 & \alpha_7 & \alpha_{11} & \alpha_{15} \end{bmatrix}$$

Composition of a round AES-128 is composed of 10 rounds which are composed of :

- AddKey xors the state with the round key (see Key Schedule);
- SubBytes which applies the AES Sbox on all individual elements α_i;
- ShiftRows which shifts the i-th row by i position;
- MixColumns which multiplies each column by a fixed matrix.

The last round omits the Mixcolumns operation and applies one extra AddKey.

Key Schedule The round key expansion from the 128-bit master key $K = (k_0|k_2|k_3|k_4)$ for the subkey of round i, K_i for E is as follows :

$$K_0 = (k_0|k_1|k_2|k_3) \text{ and } K_{i+1} = (k_{4i+4}|k_{4i+5}|k_{4i+6}|k_{4i+7}) \text{ for } i \text{ from 0 to 9}$$

$$\begin{aligned} k_{4i+4} &= \text{SubWord}\left(\text{RotWord}(k_{4i+3})\right) \oplus k_{4i} \oplus rc_i \\ k_{4i+5} &= k_{4i+4} \oplus k_{4i+1} \\ k_{4i+6} &= k_{4i+5} \oplus k_{4i+2} \\ k_{4i+7} &= k_{4i+6} \oplus k_{4i+3} \end{aligned}$$

$$\text{with } rc_i = \begin{pmatrix} X^i \bmod X^8 + X^4 + X^3 + X + 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

D.2 Best known attacks on AES

List of reduced-round attacks We expose a quick list of the best known attack against round-reduced AES-128 in the different settings.

Attack	Rounds	Time	Data	Reference
Mixture Differential	5	$2^{21.5}$	$2^{21.5}$	[BDK+18]
Yoyo	5	2^{33}	$2^{13.3}$	[RBH17]
Partial Sum	5	2^{40}	2^{8}	[Tun12]
Rectangle	5	2^{23}	2^{9}	[DKRS20]
Rectangle	5	$2^{16.5}$	2^{15}	[DKRS20]
Improved Square	5	2^{35}	2^{33}	[FKL+00]
Boomeyong	5	2^{49}	2^{49}	[RSP21]
Rectangle	6	2^{80}	2^{26}	[DKRS20]
Partial Sum	6	2^{44}	$2^{34.5}$	[FKL+00]
Truncated Differential	6	$2^{78.7}$	$2^{71.3}$	[BGL20]
Boomeyong	6	$2^{79.72}$	$2^{79.72}$	[RSP21]
Impossible Differential	7	$2^{117.2}$	$2^{112.2}$	[LDKK08]
Meet-in-the-Middle	7	2^{116}	2^{116}	[DKS10]
Impossible Differential	7	2^{113}	$2^{105.1}$	[BLSNP18]
Impossible Differential	7	$2^{110.9}$	$2^{104.9}$	[LP21]
Meet-in-the-Middle	7	2^{99}	2^{97}	[DFJ13]

Fig. 6: Current cryptanalysis for round-reduced AES-128 in the secret-key model.

E Best attacks found of Double-AES

The two best attacks we have found work on 3 rounds in E, E_3 and E_4 blocks and any number of rounds in the two first ones (X-3-3); and on 2 rounds in E while having any number of rounds in the upper and lower blocks (X-2-X) and are, logically, quantum attacks. They both use a step consisting on guessing at least a whole 128-bit key.

They also exploit the following property:

First middle round canceled. Given the key-addition operation of the AES, the input of the middle block after the first key addition will be equal to the output of E_2. This property is very interesting. As an example, if we consider differences and if the input of the right part of the state is fixed, the first SB transfer of the middle round will have no active sboxes,

Attack	Rounds	Time	Data	Reference
Related-key				
RK Boomerang	7	2^{97}	2^{97}	[BN10]
Chosen-key				
Multi-collision	9	2^{55}	2^{55}	[FJP13]
Multiple-of-n	9	2^{64}	2^{64}	[GLR+20]
Known-key				
Uniform Distribution	10	2^{64}	2^{64}	[Gil14]
Uniform Distribution	12	2^{82}	2^{82}	[GR20]

Fig. 7: Current cryptanalysis for AES-128 in the related-key/chosen-key/known-key model.

Attack on X-3-3 without last MC in blocks E_1, E_2 and E. This attack is based on the square attack [FKL+00]. From it we know that, if the input is an active diagonal only in the left part of the plaintext, we will recover 2^{24} sets that verify that the state 4 rounds later is an state where all of its bytes take all the possible values independently. The aim is to generate a square state thanks to the left input at the beginning of the middle E, but we have to be careful, as this input state will also influence its subkeys. In order to make it work, we will consider a byte (in the before last column, not to affect too much further subkeys due to key-schedule), that takes all the 2^8 possible values in the input instead of structures of 2^{32}, and we will then cover with the distinguisher one less round than the original square attack.

We guess k_1 of E_1, in order to generate an input to E with two fully active bytes, as shown in figure 8. Therefor, any number of rounds in E_1 would allow the attack to work. In addition, we guess the subkeys from k_3 associated to: 32 bits of antidiagonal for the last subkey, and 8 bits of subkey for the before-last equivalent subkey, as shown in figure 9. This allows us to compute, for a given left output, a byte after the first MC transformation in E_3 and check whether the sum of the resulting bytes is 0 or not. This filters one guess out of 256.

In order to increase this sieving, we choose, instead of one fixed state for the right input, 21, which would provide a sieving of $2^{-8\times 21}$ to have balanced bytes each time, leaving approximately $2^{128+5\times 8} \times 2^{-8\times 21} = 1$ key guess.

Complexity will be $21 \times 2^8 \times 2^{(128+32+8)/2} = 2^{96.5}$ data and time.

We expect that these attacks might be extended to 4-4-4, or to 4-3-4 with MC by having a closer look at the properties generated by the key-schedule of the middle block.

Attack on X-2-X without last MC in blocks E_1, E_2 and E.

Guessing k_3. We start with a fixed pair (P_1, P_2) of left blocks of plaintexts and perform the encryption through Double-AES-X-2-X of (P_1, R), (P_2, R), for a fixed value R. We consider exclusively the left part of the output, we obtain

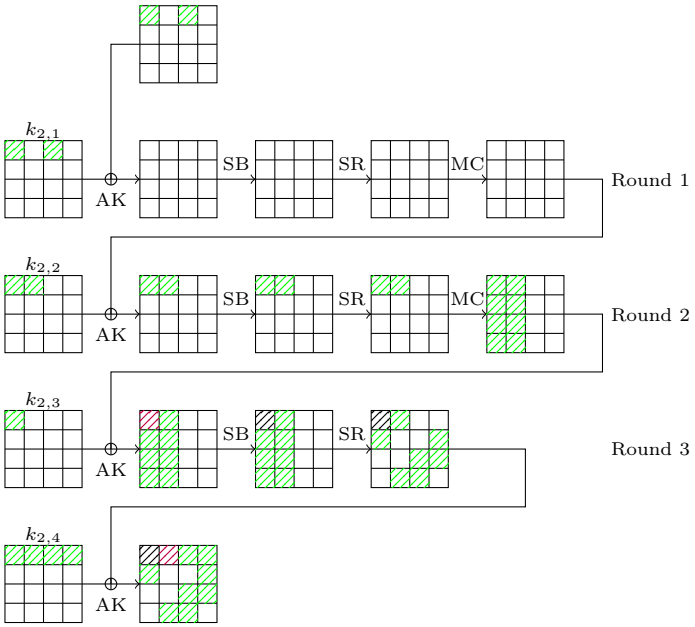


Fig. 8: Square-like property on the middle part. We use green for bytes that take every values, purple for balanced bytes and black for ignored bytes.

C_1 and C_2. For each guess of key k_3 from E_3, we will try a decryption through E_3 of C_1 and C_2 and note the difference δ.

Guessing k_1 and some middle subkeys. For each guess of key k_1 from E_1, we will try a encryption through E_1 of P_1 and P_2. This will produce values x_1 and x_2 that correspond to the values that should enter the middle part E.

Then, we will experience the cancellation of the first round as described earlier. The second round starts by a key addition, and we can get to know the differences on the bytes 0,4,8 and 12 (the first line) before the second SB for one additional guess of the byte 13 of $E_2(R)$. Each one of this differences can be associated to $2^{32-4} = 2^{28}$ output differences through the DDT of these four sboxes. The output differences of these second SB will be determined by δ xored to the last subkey of the middle round. In order to compute the difference of this subkey for the first line, and therefore the posible values for finding a match of this first line with the δs, we can perform an additional guess guess of the byte 14 of $E_2(R)$ and the xor of the bytes 1,5,9 and 13 of $E_2(R)$. We therefore get to compute the possible output differences of E and compare them to the differences δ we described earlier.

As we just said, the probability of this sieving is of 2^{-4} because of the DDT.

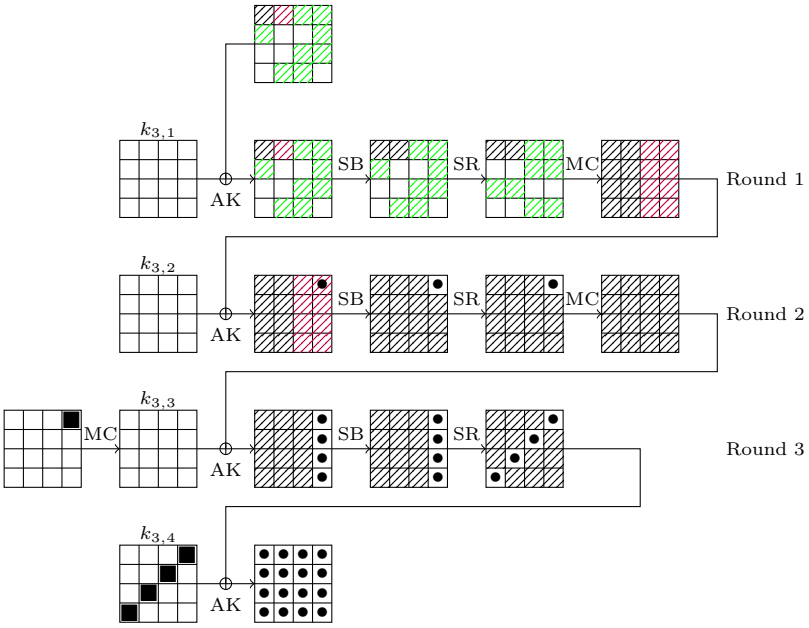


Fig. 9: Recovery on the bottom part. We use green for bytes that take every value, purple for balanced bytes, black for ignored bytes, ■ for guessed bytes and • for deduced and known bytes.

More pairs. In order to sieve more guesses, we use 70 pairs instead of one, which leaves approximately $2^{128+128+3\times 8} \times 2^{-4\times 70} = 1$ combination.

Complexity. We can then use an element distinctiveness algorithm to find the correct combination of (k_1, k_3). Thanks to Ambainis algorithm [Amb07], the cost of this attack will be about $70 \times (2^{128} + 2^{128+24})^{2/3} = 2^{107.5}$ time and memory.