Post-Quantum Security of the Sum of Even-Mansour

YanJin Tan[1], JunTao Gao[1] and XueLian Li[2]

[1] School of Telecommunications Engineering, Xidian University, Xi'an, China
[2] School of Mathematics and Statistics, Xidian University, Xi'an, China

Abstract

The Sum of Even-Mansour (SoEM) construction, introduced at Crypto 2019, implements a pseudorandom function via the XOR sum of independent Even-Mansour structures and serves as the core design for symmetric schemes like CENCPP* and nEHTm. This paper establishes the tight post-quantum security of the single-key, two-permutation variant, SoEM21, under the Q1 adversarial model. We prove that when an adversary makes classical queries to the keyed construction and quantum superposition queries to the underlying permutations, SoEM21 guarantees up to $n/3$ bits of security. This bound exactly matches the $\tilde{O}(2^{n/3})$ complexity of the recent quantum key recovery attack proposed by Li et al. To demonstrate the applicability of this theoretical bound, we further analyze two SoEM-based symmetric primitives. We use our results to prove the post-quantum security—in the same model—of the symmetric-key schemes 1K-CENCPP* (a nonce-based encryption mode) and the core structure of $n\mathsf{EHTm}_p$ (a nonce-based message authentication code).

Keywords: Post-Quantum Cryptography, Sum of Even-Mansour, Q1 Quantum Security, Tight Security Bound

1 Introduction

The advent of quantum computation has prompted a re-examination of the security guarantees provided by classical symmetric-key constructions. A central question is how much security remains when adversaries can perform quantum computation while interacting with deployed keyed primitives only through classical interfaces. This setting is commonly captured by the *Q1 model*: the adversary may make quantum queries to public primitives, such as public permutations, but only classical queries to the keyed construction. This model is particularly relevant for standard symmetric-key deployments, where encryption and authentication algorithms are implemented on classical devices and do not expose quantum superposition access to the secret-key oracle.

In this work, we study the post-quantum security of the Sum of Even-Mansour (SoEM) construction introduced by Chen et al. [CLM19]. SoEM builds pseudorandom functions from public random permutations by summing several Even-Mansour-type components. This design paradigm is attractive in symmetric cryptography because it combines a simple algebraic structure with efficient evaluation from public permutations, and it underlies nonce-based encryption and authentication schemes such as CENCPP* [BDLN22] and

Emails: 24011211360@stu.xidian.edu.cn (YanJin Tan), jtgao@mail.xidian.edu.cn (JunTao Gao), xlli@mail.xidian.edu.cn (XueLian Li)

Date of this document: 2026-05-11.

nEHTm [CLLL20]. We focus on the single-key, two-permutation variant SoEM21, defined by

$$E_k(x) = P_1(x \oplus k) \oplus P_2(x \oplus k) \oplus k,$$

where P_1 and P_2 are independent public permutations and $k \in \{0,1\}^n$ is the secret key.

The security landscape of SoEM21 is only partially understood in the post-quantum setting. Classically, SoEM21 is secure up to the birthday bound, i.e., roughly $2^{n/2}$ queries [CLM19]. In the *Q2 model*, where the adversary is given quantum superposition access to the keyed construction itself, Shinagawa and Iwata [SI22] showed that SoEM21 admits a polynomial-time attack based on Simon's algorithm. More recently, Li et al. [LFG+25] gave an offline Simon-type key-recovery attack against SoEM21 in the Q1 model with complexity $\tilde{O}(2^{n/3})$. This establishes an upper bound on the Q1 security of SoEM21, but no matching lower bound was previously known. Determining whether SoEM21 indeed achieves tight $n/3$-bit security in the Q1 model is the main question addressed in this paper.

Proving such a lower bound is technically delicate. In the Q1 model, the adversary accumulates a classical transcript from the keyed oracle while simultaneously making quantum queries to the public permutations. For SoEM21, each classical query x_i imposes consistency constraints on both P_1 and P_2 at the same hidden coordinate $x_i \oplus k$. Since the adversary's quantum queries cannot be recorded directly without disturbing its state [Zha19], the proof must maintain consistency with the classical transcript while bounding the adversary's ability to detect hidden reprogramming points. We address this difficulty by combining transcript-driven reprogramming with resampling arguments for public permutations.

Our work follows a growing line of research on Q1 security of symmetric constructions. Offline Simon-type attacks show that several classical constructions can lose more security than what is predicted by Grover search alone [BHNPS19]. On the positive side, Jaeger et al. [JST21] proved Q1 lower bounds for the FX construction, while Alagic et al. [ABKM22a] established tight $n/3$-bit Q1 security for Even-Mansour and later extended their techniques to tweakable Even-Mansour [ABKM22b]. Further Q1 analyses have been developed for FX variants [GHY24], key-alternating ciphers [CEM25], block-cipher-based constructions [ABMS26], multi-key settings [SI26], and keyed sponge-based modes [Hos25]. However, these results do not directly cover SoEM21, whose two-permutation summation structure requires simultaneous control of multiple public permutations tied to the same hidden key-dependent coordinate.

1.1 Contributions of This Paper

We close the above gap by proving a matching Q1 lower bound for SoEM21. Our main contributions are as follows.

1. **Tight Q1 security of SoEM21.** We establish that the SoEM21 construction achieves tight $n/3$-bit security in the Q1 adversarial model. Specifically, we prove that for any Q1 adversary making q_E classical queries to the keyed construction and q_P quantum queries to the public permutations, the distinguishing advantage is bounded by

$$16 \cdot \frac{q_E\sqrt{q_P} + q_P\sqrt{q_E}}{2^{n/2}} + \mathcal{O}\left(\frac{q_E^2}{2^n}\right).$$

   This lower bound implies that achieving a constant distinguishing advantage requires $\Omega(2^{n/3})$ queries. This matches the $\tilde{O}(2^{n/3})$ query complexity of the quantum key recovery attack proposed by Li et al. [LFG+25], thereby establishing the tight Q1 security of the SoEM21 construction.

2. **Transcript-driven reprogramming for a two-permutation sum.** The main technical difficulty is that every classical query imposes two coupled constraints, one on each public permutation, at the same hidden coordinate $x_i \oplus k$. Our proof uses a hybrid sequence that alternates between adaptive reprogramming steps, which enforce consistency with the classical transcript, and permutation-resampling steps, which hide the key-dependent coordinates from quantum queries. This yields a construction-specific Q1 proof for the composite SoEM21 architecture. The applicability of this method is demonstrated by the subsequent work of Datta et al. [DDM+26], who applied our hybrid sequence and transcript-driven reprogramming method to establish matching Q1 bounds for the keyed sum of permutations construction.

3. **Applications to SoEM-based primitives.** We apply the SoEM21 bound to two symmetric primitives built from SoEM-type components. In particular, we derive Q1-IND-CPA security for the nonce-based encryption scheme 1K-CENCPP* and Q1-PRF security for the core structure of $n\mathsf{EHTm}_p$. These results show that the applications inherit $n/3$-type Q1 security bounds from the underlying SoEM21 analysis.

To provide a clearer context for our contribution, a comparison of our newly established bound for SoEM21 with existing post-quantum security bounds for other related symmetric-key constructions is summarized in Table 1.

Table 1: Post-quantum security bounds for selected symmetric-key constructions. Our main results are highlighted in bold, successfully establishing the tight lower bound for SoEM21 specifically within the Q1 adversarial model.

Construction	Threat Model	Upper Bound (Known Attack)	Lower Bound (Provable Security)
SoEM21	Classical	$O(2^{n/2})$ [CLM19]	$\Omega(2^{n/2})$ [CLM19]
	Q1	$\tilde{O}(2^{n/3})$ [LFG+25]	$\mathbf{\Omega(2^{n/3})}$ [**Theorem 1**]
	Q2	Polynomial $O(n)$ [SI22]	–
Even-Mansour	Classical	$O(2^{n/2})$ [EM97]	$\Omega(2^{n/2})$ [EM97]
	Q1	$\tilde{O}(2^{n/3})$ [BHNPS19]	$\Omega(2^{n/3})$ [ABKM22a]
	Q2	Polynomial $O(n)$ [KM12]	Polynomial $O(n)$ [KN23]
FX Construction	Classical	$O(2^{(n+k)/2})$ [KR01]	$\Omega(2^{(n+k)/2})$ [KR01]
	Q1	$\tilde{O}(2^{(n+k)/3})$ [BHNPS19]	$\Omega(2^{(k+n)/3})$ [JST21]
	Q2	$O(n2^{k/2})$ [LM17]	–
Tweakable EM	Classical	$O(2^{n/2})$ [CLS15]	$\Omega(2^{n/2})$ [CLS15]
	Q1	$\tilde{O}(2^{n/3})$ [BHNPS19]	$\Omega(2^{n/3})$ [ABKM22b]
	Q2	Polynomial $O(n)$ [KM12]	Polynomial $O(n)$ [KN23]
t-KAC	Classical	$O(2^{\frac{t}{t+1}n})$ [BKL+12]	$\Omega(2^{\frac{t}{t+1}n})$ [CS14]
	Q1	$O(2^{\frac{t(t+1)n}{(t+1)^2+1}})$ [CEM25]	$\Omega(2^{\frac{tn}{2t+1}})$ [CEM25]
	Q2	Polynomial $O(n)$ [KLLNP16]	–

Organization. The remainder of this paper is organized as follows. Section 2 introduces the necessary notation and formal definitions of the SoEM21 construction. We also review the Q1 security game and the fundamental quantum reprogramming and resampling lemmas. Section 3 presents our main result: a tight Q1 security lower bound for SoEM21, along with detailed proofs of the intermediate quantum hybrid lemmas. Section 4 applies our framework to evaluate the post-quantum security of two concrete symmetric primitives: the 1K-CENCPP* encryption scheme and the core structure of the $n\mathsf{EHTm}_p$ message authentication code. Finally, Section 5 concludes the paper and highlights directions for future research.

2 Preliminaries

2.1 Notation and Model

Let n be the security parameter. All operations in this paper are considered over $\{0,1\}^n$. Let $\mathcal{P}_n$ denote the set of all permutations from $\{0,1\}^n$ to $\{0,1\}^n$. We uniformly and independently sample public permutations P_1 and P_2 from $\mathcal{P}_n$. The key k is a bit string uniformly sampled from $\{0,1\}^n$.

SoEM21 Construction: Defined as a function $E_k : \{0,1\}^n \to \{0,1\}^n$, computed as follows:

$$E_k(x) = P_1(x \oplus k) \oplus P_2(x \oplus k) \oplus k.$$

The specific architecture and data flow of the SoEM21 variant are illustrated in detail (See Fig. 1). The visual model highlights the initial bifurcation of the plaintext, the parallel invocation of the two independent public permutations, and the uniform application of the single cryptographic key across all branches.

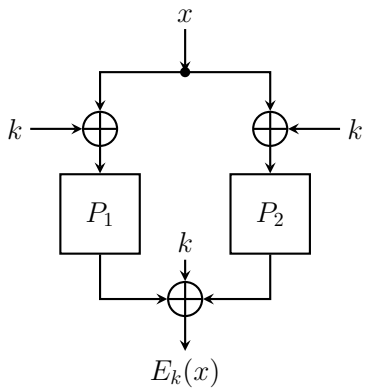

Fig. 1: The SoEM21 cipher.

We consider the adversarial capability of an adversary $\mathcal{A}$ in the **Q1 model**: $\mathcal{A}$ can make **quantum queries** to the public permutations P_1 and P_2 (and their inverses). This means $\mathcal{A}$ can apply the unitary operator $U_P : |x\rangle|y\rangle \to |x\rangle|y \oplus P(x)\rangle$ and its inverse. $\mathcal{A}$ can only make **classical queries** to the keyed construction E_k. That is, $\mathcal{A}$ provides a classical input x and receives the classical output $E_k(x)$. We denote the number of classical queries made by adversary $\mathcal{A}$ as q_E, and the total number of quantum queries (the sum of queries to P_1 and P_2) as q_P.

2.2 Security Definition

Our goal is to prove that SoEM21 is a Pseudorandom Function (PRF). We formalize its security through a distinguishing game. Unlike Pseudorandom Permutations (PRPs), PRFs are not required to be invertible. Consequently, when proving the security of a PRF, the distinguishing game is typically designed to only allow forward queries. This is because the security definition of a PRF focuses on the indistinguishability of the function's output from a random function, without considering the computability of the inverse operation. Allowing inverse queries would introduce capabilities that do not align with the PRF security goal and could invalidate the indistinguishability argument. Therefore, we assume that the adversary is limited to forward queries of the form $E_k(x) = P_1(x \oplus k) \oplus P_2(x \oplus k) \oplus k$.

Definition 1 (PRF Security Game for SoEM21 in the Q1 Model)**.** The game involves a challenger and a quantum adversary $\mathcal{A}$. The challenger flips a uniform random bit b.

- If $b = 0$ (**Real World**): The challenger randomly selects a key $k \leftarrow \{0,1\}^n$ and permutations $P_1, P_2 \leftarrow \mathcal{P}_n$. The adversary $\mathcal{A}$ is granted **quantum query access** to $P_1, P_2, P_1^{-1}, P_2^{-1}$ and **classical query access** to $E_k(\cdot) = P_1(\cdot \oplus k) \oplus P_2(\cdot \oplus k) \oplus k$.

- If $b = 1$ (**Ideal World**): The challenger samples an independent uniformly random function $R : \{0,1\}^n \to \{0,1\}^n$ with the same domain and range as E_k, and permutations $P_1, P_2 \leftarrow \mathcal{P}_n$.The adversary $\mathcal{A}$ is granted **quantum query access** to $P_1, P_2, P_1^{-1}, P_2^{-1}$ and **classical query access** to $R(\cdot)$.

At the end of the game, the adversary $\mathcal{A}$ outputs a guess bit b'. The advantage of adversary $\mathcal{A}$ in this game is defined as:

$$\mathrm{Adv}^{\text{Q1-PRF}}_{\text{SoEM21}}(\mathcal{A}) = \left|\Pr[b' = 1 \mid b = 0] - \Pr[b' = 1 \mid b = 1]\right|.$$

More generally, for any adversary making at most q_E classical queries and q_P quantum queries, we define:

$$\mathrm{Adv}^{\text{Q1-PRF}}_{\text{SoEM21}}(q_E, q_P) = \max_{\mathcal{A}} \mathrm{Adv}^{\text{Q1-PRF}}_{\text{SoEM21}}(\mathcal{A}).$$

We now introduce Alagic et al.'s Reprogramming Lemma and Resampling Lemma [ABKM22a, ABKM22b].

2.3 Reprogramming Lemma

The Reprogramming Lemma applies to the following experiment: A distinguisher D chooses an arbitrary function F and a randomized procedure B that determines a set B of points where F may be reprogrammed to some known value. Then, D is given quantum access to either F or a reprogrammed version of F; after it finishes its oracle queries, D is given B. The lemma states that as long as no point is reprogrammed with high probability, D cannot determine whether it interacted with F or its reprogrammed version.

For a function $F : \{0,1\}^m \to \{0,1\}^n$ and a set $B \subset \{0,1\}^m \times \{0,1\}^n$ such that each $x \in \{0,1\}^m$ is the first element of at most one tuple in B, define

$$F^{(B)}(x) = \begin{cases} y & \text{if } (x,y) \in B \\ F(x) & \text{otherwise} \end{cases}.$$

Lemma 1 (Reprogramming Lemma)**.** *[ABKM22a] Let D be a quantum distinguisher in the following experiment:*

Phase 1: *D outputs a description of a function $F_0 = F : \{0,1\}^m \to \{0,1\}^n$ and a description of a randomized algorithm B whose output is a set $B \subset \{0,1\}^m \times \{0,1\}^n$, where each $x \in \{0,1\}^m$ is the first element of at most one tuple in B. Let $B_1 = \{x \mid \exists y : (x,y) \in B\}$ and*

$$\epsilon = \max_{x \in \{0,1\}^m} \{\Pr[x \in B_1]\}.$$

Phase 2: *Run B to obtain B. Let $F_1 = F^{(B)}$. Choose a uniform random bit b, and D is given quantum access to F_b.*

Phase 3: *D loses access to F_b and receives the randomness r used to invoke B in Phase 2. Then D outputs a guess b'.*

For any D that makes at most q queries to its oracle in expectation,

$$|\Pr[D \text{ outputs } 1 \mid b = 1] - \Pr[D \text{ outputs } 1 \mid b = 0]| \leq 2q \cdot \sqrt{\epsilon}.$$

2.4 Resampling Lemma

The Resampling Lemma applies to the following experiment: First, a distinguisher D is given quantum access to a random permutation P. Then, in a second phase, P may be reprogrammed so that its value at a single uniform point s is changed to an independent uniform value. Since the distribution of $P(s)$ is identical before and after any such reprogramming, we call it "resampling". D's goal is to determine whether its oracle was resampled. That is, D can tell only if it placed significant amplitude on s in a query during the first phase, even if given s and continued oracle access in the second phase.

Given a permutation $P : \{0,1\}^n \to \{0,1\}^n$ and $s, y \in \{0,1\}^n$, define the reprogrammed function $P_{s\to y} : \{0,1\}^n \to \{0,1\}^n$ as

$$P_{s\to y}(w) = \begin{cases} y & \text{if } w = s \\ P(w) & \text{otherwise} \end{cases}.$$

Lemma 2 (Resampling Lemma)**.** *[ABKM22a] Let D be a quantum distinguisher in the following two-phase experiment:*

Phase 1: *D is given quantum access to a uniform random permutation $P_0 = P$ and its inverse $P_0^{-1} = P^{-1}$.*

Phase 2: *Uniformly choose $s_0, s_1 \in \{0,1\}^n$. Define P_1: if $b = 0$, then $P_1 = P$; if $b = 1$, then $P_1 = P \circ \mathrm{swap}_{s_0,s_1}$, where swap_{s_0,s_1} swaps the values at s_0 and s_1. D receives s_0, s_1 and is given quantum access to P_1, P_1^{-1}. Finally, D outputs a guess bit b'.*

For D making at most q queries to P_0, P_0^{-1} in Phase 1, we have:

$$|\Pr[D \text{ outputs } b' = 0] - \Pr[D \text{ outputs } b' = 1]| \leq 4\sqrt{\frac{q}{2^n}}.$$

3 Post-Quantum Security of the SoEM21 Construction

We now establish the post-quantum security of the SoEM21 construction based on the lemmas from the previous section. Recall that the SoEM21 construction is defined as: $E_k(x) = P_1(x \oplus k) \oplus P_2(x \oplus k) \oplus k$, where $P_1, P_2 : \{0,1\}^n \to \{0,1\}^n$ are independent public random permutations, and $k \in \{0,1\}^n$ is a uniform key. Our proof considers an adversary $\mathcal{A}$ who can make classical queries to E_k (and its inverse, if applicable) and quantum queries to P_1, P_2 (and their inverses). $\mathcal{A}$'s goal is to distinguish the real world (interacting with $E_k[P_1, P_2]$ and P_1, P_2) from the ideal world (interacting with an independent uniformly random function $R : \{0,1\}^n \to \{0,1\}^n$ and independent random permutations P_1, P_2).In the following, we let $\mathcal{P}_n$ denote the set of all permutations over $\{0,1\}^n$. We write $E_k[P_1, P_2]$ to emphasize the dependence on P_1, P_2. Our main result is as follows:

Theorem 1 (Main Theorem)**.** *For any adversary $\mathcal{A}$ in the Q1 model making q_E classical queries and q_P quantum queries, it holds that:*

$$Adv^{Q1\text{-}PRF}_{SoEM21}(q_E, q_P) \leq 16 \cdot \frac{q_E\sqrt{q_P} + q_P\sqrt{q_E}}{2^{n/2}} + \frac{8q_E^2}{2^n}.$$

In particular, when $q_E, q_P \ll 2^{n/3}$, the adversary's advantage is negligible. This proves that SoEM21 provides tight $n/3$-bit security in the Q1 model.

3.1 Formal Proof of the Main Theorem

Proof of Theorem 1. We now apply the established quantum lemmas to prove the Q1 security of SoEM21. Without loss of generality, we assume that the adversary $\mathcal{A}$ never makes a redundant classical query; that is, once it learns the output y for a classical input

x, it never again submits x to its classical oracle. We divide the execution of the adversary $\mathcal{A}$ into $q_E + 1$ phases, denoted $0, \ldots, q_E$.Specifically, phase j is defined as the computation occurring between $\mathcal{A}$'s j-th and $(j+1)$-th classical queries. To maximize its distinguishing advantage, $\mathcal{A}$ may adaptively distribute its total budget of q_P quantum queries across these phases. Let $q_{P,j}$ be the expected number of quantum queries made during phase j, such that:

$$\sum_{j=0}^{q_E} q_{P,j} = q_P. \tag{1}$$

Let the ordered historical transcript of the adversary's first j classical queries be denoted as the set $T_j = \{(x_1, y_1), \ldots, (x_j, y_j)\}$. To maintain strict algebraic consistency between the expanding classical transcript and the underlying quantum oracles, we must dynamically reprogram the public permutations. However, to preserve the bijective property of the permutations, we must avoid internal collisions during the sequential swap operations.

For each recorded query index $i \in \{1, \ldots, j\}$, we define the key-dependent input coordinate as $a_i = x_i \oplus k$. Let $R_{1,i-1}$ and $R_{2,i-1}$ denote the sets of all previously assigned or accessed range values for P_1 and P_2, respectively:

$$R_{1,i-1} = \{u_1, \ldots, u_{i-1}\} \cup \{P_1(a_1), \ldots, P_1(a_{i-1})\},$$

$$R_{2,i-1} = \{t_1, \ldots, t_{i-1}\} \cup \{P_2(a_1), \ldots, P_2(a_{i-1})\}.$$

We uniformly sample an intermediate value $u_i \in \{0,1\}^n$, and define the target for P_2 as $t_i = k \oplus u_i \oplus y_i$. We define the reprogramming collision event $\mathsf{Bad}_{coll}^{(i)}$, which occurs if any of the following conditions hold:

1. $u_i \in R_{1,i-1}$ (Range collision in P_1)
2. $t_i \in R_{2,i-1}$ (Range collision in P_2)
3. $a_i \in \{a_1, \ldots, a_{i-1}, z_1, \ldots, z_{i-1}, w_1, \ldots, w_{i-1}\}$ (Domain collision with prior coordinates)

If $\mathsf{Bad}_{coll}^{(i)}$ occurs, the simulator aborts. Over the entire execution of q_E queries, the cumulative probability of a reprogramming abort is bounded by a union bound. Since u_i is sampled uniformly, we have $|R_{1,i-1}| \leq 2(i-1)$ and $|R_{2,i-1}| \leq 2(i-1)$. Thus:

$$\Pr[\mathsf{Bad}_{coll}^{(i)}] \leq \frac{2(i-1)}{2^n} + \frac{2(i-1)}{2^n} + \frac{3(i-1)}{2^n} = \frac{7(i-1)}{2^n}.$$

Defining $\mathsf{Bad}_{coll} = \bigcup_{i=1}^{q_E} \mathsf{Bad}_{coll}^{(i)}$, we obtain:

$$\Pr[\mathsf{Bad}_{coll}] \leq \sum_{i=1}^{q_E} \frac{7(i-1)}{2^n} \leq \frac{4q_E^2}{2^n}. \tag{2}$$

Conditioned on $\neg\mathsf{Bad}_{coll}$, the unique inverse coordinates $z_i = P_1^{-1}(u_i)$ and $w_i = P_2^{-1}(t_i)$ are guaranteed to be disjoint from previously modified domain points. We then execute the discrete swaps:

$$S_{T_j,P_1,k} = \prod_{i=1}^{j} \mathrm{swap}_{a_i,z_i}, \quad S_{T_j,P_1,k} \circ P_1 = P_{1,T_j,k},$$

$$S_{T_j,P_2,k} = \prod_{i=1}^{j} \mathrm{swap}_{a_i,w_i}, \quad S_{T_j,P_2,k} \circ P_2 = P_{2,T_j,k}.$$

This reprogramming mechanism perfectly simulates the expected classical cipher output:

$$P_{1,T_j,k}(x_i \oplus k) \oplus P_{2,T_j,k}(x_i \oplus k) \oplus k = y_i. \tag{3}$$

For the remainder of the proof, we condition the hybrid experiments on the event $\neg \mathsf{Bad}_{coll}$.

With this transcript consistency mechanism rigorously established, we formally define the baseline hybrid experiments H_j and H'_j to progressively orchestrate the mathematical transition from the real-world cryptographic deployment to the ideal statistical simulation.

We now define hybrid experiments H_j for $j = 0, 1, \cdots, q_E$.

Experiment $\mathbf{H}_j$:

1. The distinguisher $\mathcal{D}$ samples two independent random permutations $P_1, P_2 \leftarrow \mathcal{P}_n$ and an independent uniformly random function $R : \{0,1\}^n \to \{0,1\}^n$.
2. It then runs $\mathcal{A}$, answering its quantum queries using quantum oracle access to P_1 and P_2, and answering its classical queries using oracle access to R, **stopping immediately before its $(j+1)$-st classical query**. Let $\mathbf{T_j} = \{(\mathbf{x_1}, \mathbf{y_1}), \ldots, (\mathbf{x_j}, \mathbf{y_j})\}$ be the list of classical query-response pairs.
3. For the remaining execution of $\mathcal{A}$, use $E_k[P_{1,T_j,k}, P_{2,T_j,k}]$ to answer its classical queries, and use $P_{1,T_j,k}, P_{2,T_j,k}$ to answer its quantum queries.

We now define hybrid experiments H'_j for $j = 0, 1, \cdots, q_E - 1$.

Experiment $\mathbf{H}'_j$:

1. $\mathcal{D}$ samples two independent random permutations $P_1, P_2 \leftarrow \mathcal{P}_n$ and an independent uniformly random function $R : \{0,1\}^n \to \{0,1\}^n$.
2. It then runs $\mathcal{A}$, answering its quantum queries using quantum oracle access to P_1 and P_2, and answering its classical queries using oracle access to R, **stopping immediately after its $(j+1)$-st classical query**. Let $\mathbf{T_{j+1}} = \{(\mathbf{x_1}, \mathbf{y_1}), \ldots, (\mathbf{x_{j+1}}, \mathbf{y_{j+1}})\}$ be the list of classical query-response pairs.
3. For the remaining execution of $\mathcal{A}$, use $E_k[P_{1,T_{j+1},k}, P_{2,T_{j+1},k}]$ to answer its classical queries, and use $P_{1,T_{j+1},k}, P_{2,T_{j+1},k}$ to answer its quantum queries.

To intuitively illustrate the distinction between the experiments, their respective oracle sequences are depicted as follows:

Oracle sequence for H_j :

$$\underbrace{(P_1, P_2, R)^j}_{j \text{ classical queries}} \ , \quad \underbrace{\left(P_{1,T_j,k}, P_{2,T_j,k}, E_k[P_{1,T_j,k}, P_{2,T_j,k}]\right)^{q_E - j}}_{q_E - j \text{ remaining classical queries}}$$

Oracle sequence for H'_j :

$$\underbrace{(P_1, P_2, R)^{j+1}}_{j+1 \text{ classical queries}} \ , \quad \underbrace{\left(P_{1,T_{j+1},k}, P_{2,T_{j+1},k}, E_k[P_{1,T_{j+1},k}, P_{2,T_{j+1},k}]\right)^{q_E - j - 1}}_{q_E - j - 1 \text{ remaining classical queries}}$$

In subsequent Lemma 3 and Lemma 4, we establish bounds on the distinguishability between H_j and H'_j and between H'_j and H_{j+1}. For $0 \le j < q_E$ we have:

$$\left|\Pr[\mathcal{A}(\mathrm{H}_j) = 1] - \Pr[\mathcal{A}(\mathrm{H}'_j) = 1]\right| \le 16\sqrt{\frac{q_P}{2^n}} + \frac{3j}{2^n} + \frac{2q_E}{2^n}, \tag{4}$$

$$\left|\Pr[\mathcal{A}(\mathrm{H}'_j) = 1] - \Pr[\mathcal{A}(\mathrm{H}_{j+1}) = 1]\right| \le 2 \cdot q_{P,j+1}\sqrt{2(j+1)/2^n}. \tag{5}$$

Then we have:

$$\begin{aligned}
& |\Pr[\mathcal{A}(\mathrm{H}_0)=1]-\Pr[\mathcal{A}(\mathrm{H}_{q_E})=1]| \\
\leq\ & \Pr[\mathsf{Bad}_{coll}]+\sum_{j=0}^{q_E-1}\Big|\Pr[\mathcal{A}(\mathrm{H}_j)=1]-\Pr[\mathcal{A}(\mathrm{H}'_j)=1]\Big|_{\neg\mathsf{Bad}_{coll}} \\
& \qquad +\sum_{j=0}^{q_E-1}\Big|\Pr[\mathcal{A}(\mathrm{H}'_j)=1]-\Pr[\mathcal{A}(\mathrm{H}_{j+1})=1]\Big|_{\neg\mathsf{Bad}_{coll}} \\
\leq\ & \frac{4q_E^2}{2^n}+\sum_{j=0}^{q_E-1}\left[16\sqrt{\frac{q_P}{2^n}}+\frac{3j}{2^n}+\frac{2q_E}{2^n}+2\cdot q_{P,j+1}\sqrt{\frac{2(j+1)}{2^n}}\right] \\
\leq\ & \frac{4q_E^2}{2^n}+\frac{2q_E^2}{2^n}+\frac{3q_E(q_E-1)}{2\cdot 2^n}+\sum_{j=0}^{q_E-1}\left(16\sqrt{\frac{q_P}{2^n}}+2\cdot q_{P,j+1}\sqrt{\frac{2q_E}{2^n}}\right) \\
\leq\ & \frac{4q_E^2}{2^n}+\frac{2q_E^2}{2^n}+\frac{3q_E^2}{2\cdot 2^n}+16q_E\sqrt{\frac{q_P}{2^n}}+2q_P\sqrt{\frac{2q_E}{2^n}} \\
\leq\ & \frac{8q_E^2}{2^n}+16q_E\sqrt{\frac{q_P}{2^n}}+3q_P\sqrt{\frac{q_E}{2^n}} \\
\leq\ & 16\cdot\frac{q_E\sqrt{q_P}+q_P\sqrt{q_E}}{2^{n/2}}+\frac{8q_E^2}{2^n}.
\end{aligned}$$

Since we operate in the regime where $q_E, q_P \ll 2^{n/2}$, the classical collision term $O(q_E^2/2^n)$ is strictly dominated by the principal quantum term and does not affect the asymptotic security lower bound. A constant distinguishing advantage therefore requires queries on the $2^{n/3}$ scale in the balanced regime $q_E \approx q_P$. This matches, up to logarithmic factors, the offline Simon-type key-recovery attack of Li et al. [LFG+25], which runs in $\widetilde{O}(2^{n/3})$ queries. Hence the bound establishes the tight $n/3$-bit Q1 security of SoEM21 in the query-exponent sense. □

3.2 Proof of Intermediate Quantum Lemmas

To formally validate the overarching bounds utilized in the summation above, we now present the rigorous proofs governing the statistical perturbations of the intermediate state transitions.

Lemma 3. *Let $\mathcal{A}$ be an adversary making at most q_E classical queries and q_P quantum queries. Then for $j = 0, \cdots, q_E - 1$, we have:*

$$\left|\Pr[\mathcal{A}(H'_j)=1]-\Pr[\mathcal{A}(H_{j+1})=1]\right| \leq 2\cdot q_{P,j+1}\sqrt{2(j+1)/2^n}.$$

Proof. We can write the oracle sequences defined by H'_j and H_{j+1} as follows:

$$\mathrm{H}'_j : \underbrace{(P_1,P_2,R)^{j+1}}_{j+1 \text{ queries}},\underbrace{\mathbf{P_1},_{\mathbf{T_{j+1}},\mathbf{k}},\mathbf{P_2},_{\mathbf{T_{j+1}},\mathbf{k}}}_{\text{phase } j+1 \text{ quantum}},\underbrace{\left(P_{1,T_{j+1},k},P_{2,T_{j+1},k},E_k[P_{1,T_{j+1},k},P_{2,T_{j+1},k}]\right)^{q_E-j-1}}_{q_E-j-1 \text{ remaining queries}}$$

$$\mathrm{H}_{j+1} : \underbrace{(P_1,P_2,R)^{j+1}}_{j+1 \text{ queries}},\quad \underbrace{\mathbf{P_1},\mathbf{P_2}}_{\text{phase } j+1 \text{ quantum}}\quad ,\underbrace{\left(P_{1,T_{j+1},k},P_{2,T_{j+1},k},E_k[P_{1,T_{j+1},k},P_{2,T_{j+1},k}]\right)^{q_E-j-1}}_{q_E-j-1 \text{ remaining queries}}$$

We now define an augmented combined function to facilitate the use of the Reprogramming Lemma that explicitly captures both forward and inverse queries.[3] Let $F : \{+,-\}\times\{1,2\}\times\{0,1\}^n \to \{0,1\}^n$, defined as:

[3]Specifically, reprogramming a permutation via a swap operation intrinsically alters its inverse mapping at two distinct coordinates. The augmented function explicitly tracks these dual modifications to prevent undetected inverse queries.

- $F(+,1,x) = P_1(x)$ and $F(-,1,y) = P_1^{-1}(y)$
- $F(+,2,x) = P_2(x)$ and $F(-,2,y) = P_2^{-1}(y)$

The adversary's quantum queries to $P_1, P_2, P_1^{-1}, P_2^{-1}$ are now seamlessly viewed as queries to F. Therefore, the adversary's total quantum queries to F remains q_P.

Bounding Reprogramming Probability ϵ: To strictly preserve the bijectivity of the permutations without creating output collisions, the reprogramming of P_1 and P_2 at the key-dependent operational point $a_i = x_i \oplus k$ is implemented via swap operations. For any query $i \in \{1, \ldots, j+1\}$, choosing a uniformly random intermediate value $u_i \in \{0,1\}^n$ means:

1. Reprogramming $P_1(a_i)$ to u_i implicitly swaps the outputs at a_i and the inverse coordinate $z_i = P_1^{-1}(u_i)$. This modifies four specific points in our augmented function F: the forward evaluations at a_i and z_i, and the inverse evaluations at $P_1(a_i)$ and u_i.
2. Reprogramming $P_2(a_i)$ to $t_i = k \oplus u_i \oplus y_i$ implicitly swaps the outputs at a_i and the inverse coordinate $w_i = P_2^{-1}(t_i)$. This similarly modifies four points in F: the forward evaluations at a_i and w_i, and the inverse evaluations at $P_2(a_i)$ and t_i.

Therefore, the set of reprogrammed inputs for F is a subset of:

$$B_1 = \bigcup_{i=1}^{j+1} \left\{ \begin{array}{l} (+,1,a_i), (+,1,z_i), (-,1,P_1(a_i)), (-,1,u_i), \\ (+,2,a_i), (+,2,w_i), (-,2,P_2(a_i)), (-,2,t_i) \end{array} \right\}.$$

For any fixed coordinate $(d, \mathrm{idx}, v) \in \{+,-\} \times \{1,2\} \times \{0,1\}^n$, we calculate the probability it is reprogrammed by the i-th query:

- **If $d = +$ and $\mathbf{idx} = 1$:** The target v is reprogrammed if $v = a_i$ or $v = z_i$. Because the hidden key k is uniformly random and independent of x_i, a_i is perfectly uniform. Because the intermediate mask u_i is uniformly sampled, z_i is also uniformly distributed. Thus, $\Pr[v = a_i \vee v = z_i] \leq \frac{1}{2^n} + \frac{1}{2^n} = \frac{2}{2^n}$.
- **If $d = -$ and $\mathbf{idx} = 1$:** The target v is reprogrammed if $v = P_1(a_i)$ or $v = u_i$. Both $P_1(a_i)$ and u_i are uniformly distributed, yielding a probability $\leq \frac{2}{2^n}$.
- **For $\mathbf{idx} = 2$:** The exact same uniformity guarantees apply to P_2 for both forward and inverse targets, as t_i and w_i are uniformly distributed due to the randomness of the intermediate mask u_i.

Applying the union bound over these mutually exclusive targets for a single query yields a maximum reprogramming probability for any fixed point of $\frac{2}{2^n}$. To capture the cumulative risk over the entire historical transcript of $j+1$ classical queries, we apply a final union bound over the summation index i:

$$\epsilon = \max_{(d,\mathrm{idx},v)} \Pr[(d,\mathrm{idx},v) \in B_1] \leq \sum_{i=1}^{j+1} \frac{2}{2^n} = \frac{2(j+1)}{2^n}.$$

Applying Lemma 1 with these bounds, we obtain:

$$\left|\Pr[\mathcal{A}(\mathrm{H}'_j) = 1] - \Pr[\mathcal{A}(\mathrm{H}_{j+1}) = 1]\right| \leq 2 \cdot q_{P,j+1} \sqrt{\frac{2(j+1)}{2^n}}.$$

□

Lemma 4. *Let $\mathcal{A}$ be an adversary making at most q_E classical queries and q_P quantum queries. Then for $j = 0, \cdots, q_E$, we have:*

$$\left|\Pr[\mathcal{A}(H_j) = 1] - \Pr[\mathcal{A}(H'_j) = 1]\right| \leq 16\sqrt{\frac{q_P}{2^n}} + \frac{3j}{2^n} + \frac{2q_E}{2^n}.$$

Proof. To prove Lemma 4, we introduce auxiliary experiments: H_j^*, H_j^{**}, H_j^{***}.

(Note: These intermediate hybrids progressively decouple the key k from the adversary's prior quantum observations via permutation resampling, bridging the algebraic transition to the ideal random function R.)

Experiment $\mathbf{H}_j^*$:

1. $\mathcal{D}$ samples two independent random permutations $P_1, P_2 \leftarrow \mathcal{P}_n$ and an independent uniformly random function $R : \{0,1\}^n \to \{0,1\}^n$.
2. Run $\mathcal{A}$, using R to answer its classical queries and P_1, P_2 to answer its quantum queries, until $\mathcal{A}$ issues the $(j+1)$-th classical query x_{j+1}.
3. Uniformly sample $s_0, s_1 \in \{0,1\}^n$, define $k = s_0 \oplus x_{j+1}$. Define $P_1^{(1)}$ as

$$P_1^{(1)}(x) = P_1 \circ \mathrm{swap}_{s_0,s_1}(x)$$

(i.e., apply swap of s_0 and s_1 to P_1). Then continue running $\mathcal{A}$, using E_k to answer its remaining classical queries (including the $(j+1)$-th), and using $P_{1,T_j,k}^{(1)}, P_{2,T_j,k}$ to answer its quantum queries, where $P_{1,T_j,k}^{(1)} = S_{T_j,P_1,k} \circ P_1^{(1)}$.

Experiment $\mathbf{H}_j^{**}$:

1. $\mathcal{D}$ samples two independent random permutations $P_1, P_2 \leftarrow \mathcal{P}_n$ and an independent uniformly random function $R : \{0,1\}^n \to \{0,1\}^n$.
2. Run $\mathcal{A}$, using R to answer its classical queries and P_1, P_2 to answer its quantum queries, until $\mathcal{A}$ issues the $(j+1)$-th classical query x_{j+1}.
3. Uniformly sample $s_0, s_1 \in \{0,1\}^n$, define $k = s_0 \oplus x_{j+1}$. Define $P_1^{(1)}$ as

$$P_1^{(1)}(x) = P_1 \circ \mathrm{swap}_{s_0,s_1}(x)$$

and define $P_2^{(1)}$ as

$$P_2^{(1)}(x) = P_2 \circ \mathrm{swap}_{s_0,s_1}(x).$$

Then continue running $\mathcal{A}$, using E_k to answer its remaining classical queries (including the $(j+1)$-th), and using $P_{1,T_j,k}^{(1)}, P_{2,T_j,k}^{(1)}$ to answer its quantum queries, where $P_{1,T_j,k}^{(1)} = S_{T_j,P_1,k} \circ P_1^{(1)}$ and similarly for $P_{2,T_j,k}^{(1)}$.

Experiment $\mathbf{H}_j^{***}$:

1. $\mathcal{D}$ samples two independent random permutations $P_1, P_2 \leftarrow \mathcal{P}_n$ and an independent uniformly random function $R : \{0,1\}^n \to \{0,1\}^n$.
2. Run $\mathcal{A}$, using R to answer its classical queries and P_1, P_2 to answer its quantum queries, until the conclusion of $\mathcal{A}$'s $(j+1)$-th classical query.
3. Uniformly sample $s_0, s_1 \in \{0,1\}^n$, define $k = s_0 \oplus x_{j+1}$. Use the ideal random function $R(x_{j+1})$ to directly answer the $(j+1)$-th classical query. Use E_k to answer its remaining classical queries (definitions of $P_{1,T_j,k}^{(1)}$ and $P_{2,T_j,k}^{(1)}$ as above). And use $P_{1,T_j,k}^{(1)}, P_{2,T_j,k}^{(1)}$ to answer its quantum queries, where $P_{1,T_j,k}^{(1)} = S_{T_j,P_1,k} \circ P_1^{(1)}$ and similarly for $P_{2,T_j,k}^{(1)}$.

To intuitively illustrate the progressive decoupling of the key k in these intermediate experiments, their respective oracle sequences are depicted as follows:

Oracle sequence for H_j^* :

$$\underbrace{(P_1,P_2,R)^j}_{j \text{ queries}}, \underbrace{\left(P^{(1)}_{1,T_j,k},P_{2,T_j,k},E_k[P^{(1)}_{1,T_j,k},P_{2,T_j,k}]\right)^{q_E-j}}_{q_E-j \text{ remaining queries}}$$

Oracle sequence for H_j^{**} :

$$\underbrace{(P_1,P_2,R)^j}_{j \text{ queries}}, \underbrace{\left(P^{(1)}_{1,T_j,k},P^{(1)}_{2,T_j,k},E_k[P^{(1)}_{1,T_j,k},P^{(1)}_{2,T_j,k}]\right)^{q_E-j}}_{q_E-j \text{ remaining queries}}$$

Oracle sequence for H_j^{***} :

$$\underbrace{(P_1,P_2,R)^j}_{j \text{ queries}}, \underbrace{\mathbf{R}}_{(j+1)\text{-th query}}, \underbrace{\left(P^{(1)}_{1,T_j,k},P^{(1)}_{2,T_j,k},E_k[P^{(1)}_{1,T_j,k},P^{(1)}_{2,T_j,k}]\right)^{q_E-j-1}}_{q_E-j-1 \text{ remaining queries}}$$

We have the following inequalities which will be proven in subsequent lemmas:

$$\left|\Pr[\mathcal{A}(\mathrm{H}_j)=1]-\Pr[\mathcal{A}(\mathrm{H}_j^*)=1]\right| \le 4\sqrt{\frac{q_P}{2^n}}, \tag{6}$$

$$\left|\Pr[\mathcal{A}(\mathrm{H}_j^*)=1]-\Pr[\mathcal{A}(\mathrm{H}_j^{**})=1]\right| \le 4\sqrt{\frac{q_P}{2^n}}, \tag{7}$$

$$\left|\Pr[\mathcal{A}(\mathrm{H}_j^{**})=1]-\Pr[\mathcal{A}(\mathrm{H}_j^{***})=1]\right| \le \frac{j}{2^n}, \tag{8}$$

$$\left|\Pr[\mathcal{A}(\mathrm{H}_j^{***})=1]-\Pr[\mathcal{A}(\mathrm{H}'_j)=1]\right| \le 8\sqrt{\frac{q_P}{2^n}}+\frac{2(j+q_E)}{2^n}. \tag{9}$$

Brief justifications for the intermediate bounds:

- **Inequalities** (6) **and** (7) follow from sequential applications of the Resampling Lemma to P_1 and P_2, respectively. The adversary's quantum query budget q_P limits its ability to detect the hidden coordinate swap introduced during the resampling phase, leading to a bounded distinguishing advantage of at most $4\sqrt{\frac{q_P}{2^n}}$ for each permutation.
- **Inequality** (8) specifically bounds the distinguishability of the response to the $(j+1)$-th classical query between the two hybrid experiments. In H_j^{**}, due to the coordinate swap, the response evaluates to $y_{j+1} = P_{1,T_j,k}(s_1) \oplus P_{2,T_j,k}(s_1) \oplus k$. In H_j^{***}, the response is provided directly by the ideal random function $R(x_{j+1})$. As long as the uniformly random resampling target s_1 does not fall into the set S of j previously queried or reprogrammed points, $P_1(s_1)$ and $P_2(s_1)$ remain unreprogrammed random values, making the algebraic sum uniformly random and indistinguishable from $R(x_{j+1})$. The divergence only occurs if $s_1 \in S$, which would cause the output to collapse to a previous query result y_i, an event bounded by the probability $\frac{j}{2^n}$.
- **Inequality** (9) follows from bounding the distinguishing advantage between the quantum oracles and applying a union bound over the classical bad events. Transitioning the outputs of P_1 and P_2 at the coordinate s_0 to the targets defined in H'_j requires two independent applications of the Resampling Lemma, which bounds the difference in the quantum oracles by $2 \cdot 4\sqrt{q_P/2^n} = 8\sqrt{q_P/2^n}$. For the classical queries, the oracle systems in H_j^{***} and H'_j are identical unless one of the following

events occurs: $\mathsf{Bad}_0 = \{s_1 \in S\}$, $\mathsf{Bad}_1 = \{s_0 \in S\}$, or Bad_2, the event that a subsequent classical query x satisfies $x \oplus k \in \{s_0, s_1\}$. Because s_0 and s_1 are uniformly distributed over $\{0,1\}^n$, the probabilities of these events are bounded by $j/2^n$, $j/2^n$, and $2q_E/2^n$, respectively. Conditioned on the complement of their union, the views of the adversary are identical. Summing these bounds yields the stated inequality. Hence

$$\left|\Pr[\mathcal{A}(H_j^{***}) = 1] - \Pr[\mathcal{A}(H'_j) = 1]\right| \leq \Pr[\mathsf{Bad}_0 \cup \mathsf{Bad}_1 \cup \mathsf{Bad}_2] \leq 8\sqrt{\frac{q_P}{2^n}} + \frac{2(j+q_E)}{2^n}.$$

Applying the triangle inequality, we get:

$$\left|\Pr[\mathcal{A}(\mathrm{H}_j) = 1] - \Pr[\mathcal{A}(\mathrm{H}'_j) = 1]\right| \leq 16\sqrt{\frac{q_P}{2^n}} + \frac{3j}{2^n} + \frac{2q_E}{2^n}.$$

□

Lemma 5. *Let $\mathcal{A}$ be an adversary making at most q_E classical queries and q_P quantum queries. Then for $j = 0, \cdots, q_E$, we have:*

$$\left|\Pr[\mathcal{A}(H_j) = 1] - \Pr[\mathcal{A}(H_j^*) = 1]\right| \leq 4\sqrt{\frac{q_P}{2^n}}.$$

Proof. We can write the oracle sequences defined by H_j and H_j^* as follows:

$$\mathrm{H}_j : \quad \underbrace{(P_1, P_2, R)^j}_{j \text{ classical queries}} , \quad \underbrace{\left(\mathbf{P_{1,T_j,k}}, P_{2,T_j,k}, E_k[\mathbf{P_{1,T_j,k}}, P_{2,T_j,k}]\right)^{q_E - j}}_{q_E - j \text{ remaining queries}}$$

$$\mathrm{H}_j^* : \quad \underbrace{(P_1, P_2, R)^j}_{j \text{ classical queries}} , \quad \underbrace{\left(\mathbf{P^{(1)}_{1,T_j,k}}, P_{2,T_j,k}, E_k[\mathbf{P^{(1)}_{1,T_j,k}}, P_{2,T_j,k}]\right)^{q_E - j}}_{q_E - j \text{ remaining queries}}$$

Construct a distinguisher $\mathcal{D}$ that participates in the resampling experiment for P_1. Its behavior is divided into two stages:

Phase 1: $\mathcal{D}$ is given quantum access to random permutations P_1, P_2 and samples an independent uniformly random function $R : \{0,1\}^n \to \{0,1\}^n$ (used to answer classical queries). $\mathcal{D}$ runs the adversary $\mathcal{A}$ and answers its queries: quantum queries are answered with P_1, P_2; classical queries are answered with R. $\mathcal{D}$ continues running $\mathcal{A}$ until $\mathcal{A}$ submits the $(j+1)$-th classical query x_{j+1} (let the list of the first j classical queries be $T_j = ((x_1, y_1), ..., (x_j, y_j))$).

Phase 2: $\mathcal{D}$ receives $s_0, s_1 \in \{0,1\}^n$ from the resampling experiment (s_0, s_1 are uniformly random) and quantum access to $P_1^{(b)}$, where b is the random bit of the resampling experiment, and

$$P_1^{(b)}(x) = P_1 \circ \mathrm{swap}_{s_0,s_1}(x).$$

Set the key component $k = s_0 \oplus x_{j+1}$. Answer $\mathcal{A}$'s $(j+1)$-th classical query:

$$y_{j+1} = E_k[P^{(b)}_{1,T_j,k}, P_2](x_{j+1}) = P^{(b)}_{1,T_j,k}(x_{j+1} \oplus k) \oplus P_2(x_{j+1} \oplus k) \oplus k.$$

Substituting $k = s_0 \oplus x_{j+1}$, this simplifies to $y_{j+1} = P^{(b)}_{1,T_j,k}(s_0) \oplus P_{2,T_j,k}(s_0) \oplus k$. Continue running $\mathcal{A}$, answering remaining queries: classical queries are answered with $E_k[P^{(b)}_{1,T_j,k}, P_{2,T_j,k}]$; quantum queries are answered with the modified permutations $P_1^{(b)}$ and P_2, where $P^{(b)}_{1,T_j,k} = S_{T_j,P_1,k} \circ P_1^{(b)}$ (the sequence $S_{T_j,P_1,k}$ ensures consistency of $E_k[P^{(b)}_{1,T_j,k}, P_{2,T_j,k}]$ with classical queries); $\mathcal{D}$ outputs $\mathcal{A}$'s output bit.

When $b = 0$, $P_1^{(0)} = P_1$, and thus $\mathcal{D}$ simulates experiment H_j: $\Pr[\mathcal{D} \text{ outputs } 1 \mid b = 0] = \Pr[\mathcal{A}(\mathrm{H}_j) = 1]$. When $b = 1$, $P_1^{(1)} = P_1 \circ \mathrm{swap}_{s_0,s_1}$, and thus $\mathcal{D}$ simulates experiment H_j^*: $\Pr[\mathcal{D} \text{ outputs } 1 \mid b = 1] = \Pr[\mathcal{A}(\mathrm{H}_j^*) = 1]$. Applying the Resampling Lemma (Lemma 2), the number of queries $\mathcal{D}$ makes to P_1 in Phase 1 is q_{P1}, i.e., the number of quantum queries $\mathcal{A}$ makes to P_1 in Phase 1. Since $\mathcal{A}$'s total quantum queries are q_P, distributed between P_1 and P_2, we have $q_{P1} \leq q_P$. Combined with the conclusion of the Resampling Lemma, we get:

$$\begin{aligned}\left|\Pr[\mathcal{A}(\mathrm{H}_j) = 1] - \Pr[\mathcal{A}(\mathrm{H}_j^*) = 1]\right| &= |\Pr[\mathcal{D} \text{ outputs } 1 \mid b = 0] - \Pr[\mathcal{D} \text{ outputs } 1 \mid b = 1]| \\ &\leq 4\sqrt{\frac{q_{P1}}{2^n}} \leq 4\sqrt{\frac{q_P}{2^n}}.\end{aligned}$$

□

Lemma 6. *Let $\mathcal{A}$ be an adversary making at most q_E classical queries and q_P quantum queries. Then for $j = 0, \cdots, q_E$, we have:*

$$\left|\Pr[\mathcal{A}(H_j^*) = 1] - \Pr[\mathcal{A}(H_j^{**}) = 1]\right| \leq 4\sqrt{\frac{q_P}{2^n}}.$$

Proof. We can write the oracle sequences defined by H_j^* and H_j^{**} as follows:

$$\mathrm{H}_j^*: \quad \underbrace{(P_1, P_2, R)^j}_{j \text{ classical queries}}, \quad \underbrace{\left(P_{1,T_j,k}^{(1)}, \mathbf{P_{2,T_j,k}}, E_k[P_{1,T_j,k}^{(1)}, \mathbf{P_{2,T_j,k}}]\right)^{q_E-j}}_{q_E-j \text{ remaining queries}}$$

$$\mathrm{H}_j^{**}: \quad \underbrace{(P_1, P_2, R)^j}_{j \text{ classical queries}}, \quad \underbrace{\left(P_{1,T_j,k}^{(1)}, \mathbf{P_{2,T_j,k}^{(1)}}, E_k[P_{1,T_j,k}^{(1)}, \mathbf{P_{2,T_j,k}^{(1)}}]\right)^{q_E-j}}_{q_E-j \text{ remaining queries}}$$

Construct a distinguisher $\mathcal{D}$ that participates in the resampling experiment for P_2. Its behavior is divided into two stages:

Phase 1:$\mathcal{D}$ is given quantum access to random permutations P_1, P_2 and samples an independent uniformly random function $R : \{0,1\}^n \to \{0,1\}^n$ (used to answer classical queries). $\mathcal{D}$ runs the adversary $\mathcal{A}$ and answers its queries: quantum queries are answered with P_1, P_2; classical queries are answered with R. $\mathcal{D}$ continues running $\mathcal{A}$ until $\mathcal{A}$ submits the $(j+1)$-th classical query x_{j+1}. Let $T_j = \{(x_1, y_1), ..., (x_j, y_j)\}$ be the list of the first j classical queries.

Phase 2: $\mathcal{D}$ receives $s_0, s_1 \in \{0,1\}^n$ from the resampling experiment (s_0, s_1 are uniformly random) and quantum access to $P_2^{(b)}$ (where b is the random bit of the resampling experiment). Set the key component $k = s_0 \oplus x_{j+1}$. $\mathcal{D}$ answers $\mathcal{A}$'s $(j+1)$-th classical query:

$$y_{j+1} = E_k[P_{1,T_j,k}^{(1)}, P_{2,T_j,k}^{(b)}] = P_{1,T_j,k}^{(1)}(x_{j+1} \oplus k) \oplus P_{2,T_j,k}^{(b)}(x_{j+1} \oplus k) \oplus k.$$

Substituting $k = s_0 \oplus x_{j+1}$, this simplifies to $y_{j+1} = P_{1,T_j,k}^{(1)}(s_0) \oplus P_{2,T_j,k}^{(b)}(s_0)$. $\mathcal{D}$ continues running $\mathcal{A}$: it answers remaining classical queries with $E_k[P_{1,T_j,k}^{(1)}, P_{2,T_j,k}^{(b)}]$ and quantum queries with the modified permutations $P_1^{(1)}$ and $P_2^{(b)}$. $\mathcal{D}$ outputs $\mathcal{A}$'s output bit.

When $b = 0$: $P_2^{(0)} = P_2$. Then $\mathcal{D}$ simulates experiment H_j^*: $\Pr[\mathcal{D} \text{ outputs } 1 \mid b = 0] = \Pr[\mathcal{A}(\mathrm{H}_j^*) = 1]$. When $b = 1$: $P_2^{(1)} = P_2 \circ \mathrm{swap}_{s_0,s_1}$. Then $\mathcal{D}$ simulates experiment H_j^{**}: $\Pr[\mathcal{D} \text{ outputs } 1 \mid b = 1] = \Pr[\mathcal{A}(\mathrm{H}_j^{**}) = 1]$. By the Resampling Lemma, the number of queries $\mathcal{D}$ makes to P_2 in Phase 1 is q_{P2}, i.e., the number of quantum queries $\mathcal{A}$ makes to

P_2 in Phase 1. Since $\mathcal{A}$'s total quantum queries are q_P, distributed between P_1 and P_2, we have $q_{P2} \leq q_P$. Therefore:

$$\begin{aligned}\left|\Pr[\mathcal{A}(\mathrm{H}_j^*)=1]-\Pr[\mathcal{A}(\mathrm{H}_j^{**})=1]\right| &= |\Pr[\mathcal{D}\text{ outputs }1 \mid b=0]-\Pr[\mathcal{D}\text{ outputs }1 \mid b=1]| \\ &\leq 4\sqrt{\frac{q_{P2}}{2^n}} \leq 4\sqrt{\frac{q_P}{2^n}}.\end{aligned}$$

□

Lemma 7. *Let $\mathcal{A}$ be an adversary making at most q_E classical queries and q_P quantum queries. Then for $j = 0, \cdots, q_E$, we have:*

$$\left|\Pr[\mathcal{A}(H_j^{**})=1]-\Pr[\mathcal{A}(H_j^{***})=1]\right| \leq \frac{j}{2^n}.$$

Proof.

$$\mathrm{H}_j^{**}: \underbrace{(P_1,P_2,R)^j}_{j\text{ queries}}, \underbrace{\mathbf{E_k}[\mathbf{P}^{(\mathbf{1})}_{\mathbf{1},\mathbf{T_j},\mathbf{k}},\mathbf{P}^{(\mathbf{1})}_{\mathbf{2},\mathbf{T_j},\mathbf{k}}]}_{(j+1)\text{-th query}}, \underbrace{\left(P^{(1)}_{1,T_j,k},P^{(1)}_{2,T_j,k},E_k[P^{(1)}_{1,T_j,k},P^{(1)}_{2,T_j,k}]\right)^{q_E-j-1}}_{q_E-j-1\text{ remaining queries}}$$

$$\mathrm{H}_j^{***}: \underbrace{(P_1,P_2,R)^j}_{j\text{ queries}}, \underbrace{\mathbf{R}}_{(j+1)\text{-th query}}, \underbrace{\left(P^{(1)}_{1,T_j,k},P^{(1)}_{2,T_j,k},E_k[P^{(1)}_{1,T_j,k},P^{(1)}_{2,T_j,k}]\right)^{q_E-j-1}}_{q_E-j-1\text{ remaining queries}}$$

Construct a distinguisher $\mathcal{D}$ whose behavior is divided into two stages:

Phase 1: $\mathcal{D}$ is given quantum access to random permutations P_1, P_2 and samples an independent uniformly random function $R : \{0,1\}^n \to \{0,1\}^n$ (used to answer classical queries). $\mathcal{D}$ runs the adversary $\mathcal{A}$ and answers its queries: quantum queries are answered with P_1, P_2; classical queries are answered with R. $\mathcal{D}$ continues running $\mathcal{A}$ until $\mathcal{A}$ submits the $(j+1)$-th classical query x_{j+1}. Let $T_j = \{(x_1, y_1), \ldots, (x_j, y_j)\}$ be the list of the first j classical queries.

Phase 2: $\mathcal{D}$ receives s_0, s_1 from the resampling experiment (uniformly random). And quantum access to $P^{(1)}_{1,T_j,k}$ and $P^{(1)}_{2,T_j,k}$ (b is the random bit of the resampling experiment). Set the key component $k = s_0 \oplus x_{j+1}$. $\mathcal{D}$ continues running $\mathcal{A}$: it answers the remaining queries: for the $(j+1)$-th classical query x_{j+1}, use $E_k[P^{(1)}_{1,T_j,k}, P^{(1)}_{2,T_j,k}]$ to answer; H_j^{***} uses the random function R to answer. The other queries are identical in H_j^{**} and H_j^{***}. Classical queries are answered with $E_k[P^{(1)}_{1,T_j,k}, P^{(1)}_{2,T_j,k}]$, and quantum queries are answered with the modified permutations $P^{(1)}_{1,T_j,k}$ and $P^{(1)}_{2,T_j,k}$.

We need to prove that in H_j^{**}, the response y_{j+1} for the $(j+1)$-th query is sufficiently uniformly random from the adversary's perspective, and is indistinguishable from the response using the random function in H_j^{***}.

In H_j^{**},

$$y_{j+1} = E_k[P^{(1)}_{1,T_j,k}, P^{(1)}_{2,T_j,k}](x_{j+1}) = P^{(1)}_{1,T_j,k}(x_{j+1}\oplus k)\oplus P^{(1)}_{2,T_j,k}(x_{j+1}\oplus k)\oplus k,$$

$P^{(1)}_{1,T_j,k}(x_{j+1}\oplus k) = P^{(1)}_{1,T_j,k}(s_0) = P_{1,T_j,k}(s_1), P^{(1)}_{2,T_j,k}(x_{j+1}\oplus k) = P^{(1)}_{2,T_j,k}(s_0) = P_{2,T_j,k}(s_1)$.
Let $S = \{a_i = x_i \oplus k, 1 \leq i \leq j\}$,

$$P_{1,T_j,k}(s_1) = \begin{cases} u_i, & \text{if } s_1 \in S, s_1 = a_i \\ P_1(s_1), & \text{if } s_1 \notin S \end{cases},$$

$$P_{2,T_j,k}(s_1) = \begin{cases} y_i \oplus u_i \oplus k, & \text{if } s_1 \in S, s_1 = a_i \\ P_2(s_1), & \text{if } s_1 \notin S \end{cases},$$

where u_i is the random value set when reprogramming P_1.

In H_j^{***}, $y_{j+1} = R(x_{j+1})$ is a random value. Due to the randomness of s_1, we have $\Pr[s_1 \in S] \leq \frac{j}{2^n}$.

Now analyze the randomness of y_{j+1}.

When $s_1 \notin S$: $P_{1,T_j,k}(s_1) = P_1(s_1)$ (not reprogrammed); $P_{2,T_j,k}(s_1) = P_2(s_1)$ (not reprogrammed). Since P_1, P_2 are random permutations, $P_1(s_1)$ and $P_2(s_1)$ are uniformly random, and combined with the uniformly random k, y_{j+1} is uniformly random.

When $s_1 \in S$: Reprogramming sets $P_{1,T_j,k}$ to some random value u_i and $P_{2,T_j,k}$ to $y_i \oplus u_i \oplus k$. In this case, y_{j+1} in H_j^{**} is $P_{1,T_j,k}(s_1) \oplus P_{2,T_j,k}(s_1) \oplus k = u_i \oplus (y_i \oplus u_i \oplus k) \oplus k = y_i$, which is identical to a result obtained from a previous query, allowing the adversary to significantly distinguish H_j^{**} from H_j^{***}.

Therefore, we can give a specific upper bound:

$$\left|\Pr[\mathcal{A}(\mathrm{H}_j^{**}) = 1] - \Pr[\mathcal{A}(\mathrm{H}_j^{***}) = 1]\right| \leq \Pr[s_1 \in S] \leq \frac{j}{2^n}.$$

□

Lemma 8. *Let $\mathcal{A}$ be a quantum adversary in the Q1 model making at most q_E classical queries and q_P quantum queries. Then for any $j \in \{0, \ldots, q_E - 1\}$, we have:*

$$\left|\Pr[\mathcal{A}(H_j^{***}) = 1] - \Pr[\mathcal{A}(H'_j) = 1]\right| \leq 8\sqrt{\frac{q_P}{2^n}} + \frac{2(j + q_E)}{2^n}.$$

Proof. The fundamental divergence between experiments H_j^{***} and H'_j lies in the programming of the quantum oracles at the hidden input coordinate $s_0 = x_{j+1} \oplus k$. In H_j^{***}, the underlying permutations are $P^{(1)}_{1,T_j,k}$ and $P^{(1)}_{2,T_j,k}$, which incorporate a random swap at coordinates s_0 and s_1. Consequently, the evaluations at s_0 are given by $P_1(s_1)$ and $P_2(s_1)$. Conversely, H'_j enforces strict algebraic consistency with the classical output $y_{j+1} = R(x_{j+1})$ by explicitly reprogramming the permutations at s_0 to uniformly sampled targets u_{j+1} and $R(x_{j+1}) \oplus u_{j+1} \oplus k$, respectively.

To rigorously bound the adversary's distinguishing advantage without compromising the optimal query complexity, we decompose the transition from H_j^{***} to H'_j into two sequential applications of the Resampling Lemma (Lemma 2), followed by a union bound over classical consistency failures.

Since s_1 is a uniformly random coordinate that remains entirely hidden from the classical transcript, the values $P_1(s_1)$ and $P_2(s_1)$ in H_j^{***} are independently and uniformly distributed over the remaining permutation codomain. Similarly, in H'_j, the intermediate mask u_{j+1} is sampled uniformly, and $R(x_{j+1})$ is an independent ideal random string; thus, the explicit targets u_{j+1} and $R(x_{j+1}) \oplus u_{j+1} \oplus k$ are also independent and uniformly distributed.

Therefore, transitioning the quantum oracles from the distribution in H_j^{***} to that in H'_j is mathematically equivalent to resampling the permutations at the hidden coordinate s_0 with fresh, independent uniform values. We define two intermediate quantum resampling steps:

1. **Resampling P_1:** To transition the first permutation, we implicitly apply a swap at s_0 and a uniformly random coordinate $z = P_1^{-1}(u_{j+1})$. Because u_{j+1} is sampled uniformly, z acts as a uniformly distributed coordinate. Applying the Resampling Lemma over the adversary's quantum queries bounds the distinguishing advantage of this step by $4\sqrt{\frac{q_P}{2^n}}$.

2. **Resampling P_2:** Similarly, for the second permutation, we apply a swap at s_0 and $w = P_2^{-1}(R(x_{j+1}) \oplus u_{j+1} \oplus k)$. Given that $R(x_{j+1})$ is drawn from an ideal random

function, the target value is perfectly uniform, making w a uniformly distributed coordinate. A second application of the Resampling Lemma yields an additional advantage bounded by $4\sqrt{\frac{q_P}{2^n}}$.

Following these independent resampling operations, the quantum oracles exactly mirror the distributions of $P_{1,T_{j+1},k}$ and $P_{2,T_{j+1},k}$ as defined in H'_j. It remains to guarantee that these re-randomized coordinates do not induce classical collisions that would trivially distinguish the hybrids. Let $S = \{x_i \oplus k \mid 1 \le i \le j\}$ denote the set of hidden input coordinates associated with the prior classical transcript. We define the following failure events:

- Bad_0: The resampling target collides with the established transcript ($s_1 \in S$). Since s_1 is uniform, $\Pr[\mathsf{Bad}_0] \le \frac{j}{2^n}$.
- Bad_1: The hidden coordinate collides with the established transcript ($s_0 \in S$). Since s_0 depends on the uniform key k, $\Pr[\mathsf{Bad}_1] \le \frac{j}{2^n}$.
- Bad_2: Any of the adversary's subsequent $q_E - j - 1$ classical queries explicitly hits the hidden reprogramming coordinates ($x \oplus k \in \{s_0, s_1\}$). Over all remaining queries, $\Pr[\mathsf{Bad}_2] \le \frac{2q_E}{2^n}$.

Conditioned on the event $\neg(\mathsf{Bad}_0 \cup \mathsf{Bad}_1 \cup \mathsf{Bad}_2)$, the classical transcript is perfectly consistent across the transitions. Applying the union bound, the cumulative distinguishability is strictly bounded by the sum of the quantum resampling penalties and the classical collision probabilities:

$$\left|\Pr[\mathcal{A}(\mathrm{H}_j^{***}) = 1] - \Pr[\mathcal{A}(\mathrm{H}'_j) = 1]\right| \le 8\sqrt{\frac{q_P}{2^n}} + \frac{2j}{2^n} + \frac{2q_E}{2^n} = 8\sqrt{\frac{q_P}{2^n}} + \frac{2(j+q_E)}{2^n}.$$

□

4 Applications

In this section, we apply the theoretical results from Section 3 to evaluate the post-quantum security of practical symmetric cryptographic primitives. Specifically, we investigate two distinct applications: the nonce-based encryption scheme 1K-CENCPP* and the core structure of the message authentication code $n\mathsf{EHTm}_p$. By specializing our tight Q1-PRF bounds for both constructions, we establish a rigorous theoretical foundation for their deployment and demonstrate the versatility of our proof techniques in analyzing diverse permutation-based ciphers in the post-quantum era.

4.1 Post-Quantum Security of the 1K-CENCPP* Construction

The CENCPP* scheme is a beyond-birthday-secure nonce-based encryption mode based on public permutations [BDLN22]. In its original generalized form with a width parameter $w \ge 1$, it employs $w+1$ independent permutations $P_1, \ldots, P_{w+1}$ and two independent keys K_1, K_2. By restricting this framework to $w = 1$ and a single uniform key (i.e., setting $K_1 = K_2 = k$), we define 1K-CENCPP*, a single-key, single-block variant of CENCPP*.

Let $P_1, P_2 \in \mathcal{P}_n$ be independent public permutations. For a uniform secret key $k \in \{0,1\}^n$, a nonce $N \in \{0,1\}^n$, and a plaintext block $M \in \{0,1\}^n$, the encryption algorithm of 1K-CENCPP* is defined as:

$$C = M \oplus Y.$$

where the keystream block Y is generated via the SoEM21 construction:

$$Y = E_k(N) = P_1(N \oplus k) \oplus P_2(N \oplus k) \oplus k.$$

Decryption is trivially identical, recovering the plaintext via $M = C \oplus E_k(N)$.

We evaluate the security of 1K-CENCPP* under the Q1 model, where the adversary has quantum access to all public primitives but only classical access to the secretly keyed scheme. We formulate IND-CPA security in terms of a distinguishing experiment, providing a direct proof of quantum indistinguishability rather than relying on a standard reduction to a computationally hard problem. In this setting, the adversary must distinguish the outputs of the encryption oracle from truly random strings.

Definition 2 (Q1-IND-CPA Security)**.** The Q1-IND-CPA security game involves a challenger and a quantum adversary $\mathcal{A}$. The challenger samples a uniform key $k \leftarrow \{0,1\}^n$, random permutations $P_1, P_2 \leftarrow \mathcal{P}_n$, and a uniform bit $b \leftarrow \{0,1\}$. $\mathcal{A}$ is granted quantum superposition access to P_1, P_2 (and their inverses). Concurrently, $\mathcal{A}$ makes classical queries (N_i, M_i) to an encryption oracle $\mathcal{O}_{enc}$, under the strict Nonce-Respecting restriction (i.e., $N_i \neq N_j$ for $i \neq j$).

- If $b = 0$ (Real World), $\mathcal{O}_{enc}(N_i, M_i)$ returns $C_i = M_i \oplus E_k(N_i)$.
- If $b = 1$ (Ideal World), $\mathcal{O}_{enc}(N_i, M_i)$ returns $C_i = M_i \oplus \$(N_i)$, where $\$$ is an ideal uniformly random function.

The advantage of $\mathcal{A}$ is defined as:

$$\mathrm{Adv}^{\text{Q1-IND-CPA}}_{\text{1K-CENCPP}^*}(\mathcal{A}) = \left|\Pr[b' = 1 \mid b = 0] - \Pr[b' = 1 \mid b = 1]\right|.$$

It is crucial to clarify how the nonce mechanism defends against quantum period-finding attacks in this context. In the Q2 model, even with a nonce, superposition queries to the encryption oracle would expose the internal algebraic structure. However, in the Q1 model, the strict non-repeating property of the nonce guarantees that each classical encryption query corresponds to a completely independent input coordinate system $(N \oplus k)$ for the underlying permutations. This coordinate dispersion ensures that the adversary cannot collect consistent collision pairs or construct the periodic superposition states required for Simon's algorithm.

As we now show, Theorem 1 readily implies the IND-CPA security of 1K-CENCPP* in the post-quantum setting.

Theorem 2. *Let $\mathcal{A}$ be a quantum adversary in the Q1 model making at most q_E classical queries under the strictly enforced Nonce-Respecting rule (i.e., each encryption query uses a globally unique nonce) to its first oracle, and q_P quantum queries to its second and third oracles. Then*

$$\left|\Pr_{k\leftarrow\{0,1\}^n, P_1, P_2 \leftarrow \mathcal{P}_n}\left[\mathcal{A}^{1K\text{-}CENCPP^*_k[P_1,P_2],P_1,P_2} = 1\right] - \Pr_{P_1,P_2\leftarrow\mathcal{P}_n,\$}\left[\mathcal{A}^{\$,P_1,P_2} = 1\right]\right|$$
$$\leq 16 \cdot \frac{q_E\sqrt{q_P} + q_P\sqrt{q_E}}{2^{n/2}} + \frac{8q_E^2}{2^n}.$$

Proof. Given a Q1 quantum adversary $\mathcal{A}$ attempting to break the IND-CPA security of 1K-CENCPP*, we formalize the proof through a sequence of computationally indistinguishable hybrid games (Game-Hopping), progressively reducing the real-world encryption scenario to perfect random noise.

Game 0 (Real World): This game perfectly reflects the real 1K-CENCPP* execution environment ($b = 0$). The challenger samples a uniform key $k \leftarrow \{0,1\}^n$ and independent permutations $P_1, P_2 \leftarrow \mathcal{P}_n$. The adversary $\mathcal{A}$ is given quantum access to P_1 and P_2.

During the classical query phase, $\mathcal{A}$ adaptively submits queries (N_i, M_i). The game strictly enforces the Nonce-Respecting rule, meaning for any $i \neq j$, $N_i \neq N_j$. The oracle configurations are defined as:

- Quantum Access: $\mathcal{O}_{P_1} = U_{P_1}$, $\mathcal{O}_{P_2} = U_{P_2}$.
- Classical Access: $\mathcal{O}_{enc}(N_i, M_i) = M_i \oplus E_k(N_i) = M_i \oplus P_1(N_i \oplus k) \oplus P_2(N_i \oplus k) \oplus k$.

We have:

$$\Pr[\mathcal{A}(\text{Game } 0) = 1] = \Pr\left[\mathcal{A}^{\mathcal{O}_{enc}, \mathcal{O}_{P_1}, \mathcal{O}_{P_2}} = 1\right].$$

Game 1 (Ideal PRF Replacement): In this game, the underlying SoEM21 keystream generator $E_k(\cdot)$ is replaced by an ideal, uniformly random function $R : \{0,1\}^n \to \{0,1\}^n$ that is completely independent of P_1 and P_2. Crucially, $\mathcal{A}$ maintains its independent quantum access to the native permutations P_1 and P_2. The oracles are updated to:

- Quantum Access: $\mathcal{O}_{P_1} = U_{P_1}$, $\mathcal{O}_{P_2} = U_{P_2}$.
- Classical Access: $\mathcal{O}_{enc}(N_i, M_i) = M_i \oplus R(N_i)$.

By Theorem 1, the distinguishing advantage between Game 0 and Game 1 is tightly bounded by the Q1-PRF security of the SoEM21 construction:

$$|\Pr[\mathcal{A}(\text{Game } 0) = 1] - \Pr[\mathcal{A}(\text{Game } 1) = 1]| \leq \text{Adv}^{\text{Q1-PRF}}_{\text{SoEM21}}(q_E, q_P).$$

Game 2 (Ideal One-Time Pad): In this game, we evaluate the algebraic properties of the ciphertexts $C_i = M_i \oplus R(N_i)$. Crucially, the Nonce-Respecting restriction guarantees that every classical query index i corresponds to a unique nonce N_i. Because R is a perfect random function, the generated keystream outputs $Y_i = R(N_i)$ are perfectly uniform and mutually independent sequences in $\{0,1\}^n$. Consequently, for the adversary, each encryption $C_i = M_i \oplus Y_i$ acts identically to a perfect one-time pad (OTP). The ciphertexts C_i are uniformly distributed and completely uncoupled from the plaintext statistics. Thus, $\mathcal{O}_{enc}$ acts identically to a stateless oracle \$ that uniformly outputs random strings ($b = 1$):

- Quantum Access: $\mathcal{O}_{P_1} = U_{P_1}$, $\mathcal{O}_{P_2} = U_{P_2}$.
- Classical Access: $\mathcal{O}_{enc}(N_i, M_i) = \$(N_i)$.

Therefore:

$$\Pr[\mathcal{A}(\text{Game } 1) = 1] = \Pr\left[\mathcal{A}^{\$, \mathcal{O}_{P_1}, \mathcal{O}_{P_2}} = 1\right].$$

Combining the sequential transitions, the overall IND-CPA advantage of $\mathcal{A}$ is formally bounded by:

$$\begin{aligned}\text{Adv}^{\text{Q1-IND-CPA}}_{\text{1K-CENCPP*}}(\mathcal{A}) &= \left|\Pr[\mathcal{A}(\text{Game } 0) = 1] - \Pr\left[\mathcal{A}^{\$, \mathcal{O}_{P_1}, \mathcal{O}_{P_2}} = 1\right]\right| \\ &\leq \text{Adv}^{\text{Q1-PRF}}_{\text{SoEM21}}(q_E, q_P) \leq 16 \cdot \frac{q_E\sqrt{q_P} + q_P\sqrt{q_E}}{2^{n/2}} + \frac{8q_E^2}{2^n}.\end{aligned}$$

In particular, in the balanced-query regime this gives an inherited $n/3$-bit Q1 security bound for 1K-CENCPP* under the nonce-respecting restriction. □

4.2 Post-Quantum Security Analysis of the nEHTm$_p$ Core Structure

The nEHTm$_p$ scheme is a nonce-based message authentication code (MAC) based on the Enhanced Hash-then-Mask paradigm, which utilizes public permutations to achieve beyond-birthday-bound security guarantees [CLLL20]. Before evaluating the unforgeability of the MAC, we first establish the Q1-PRF security of its underlying core structure. To

achieve this, we adapt the hybrid sequence utilized in Section 3, optimizing the proof strategy specifically for the nonce-respecting setting. Let the core function be defined as

$$F_{K,K_h}(N, M) = P_1(N \oplus K) \oplus P_2(N \oplus K \oplus H_{K_h}(M)),$$

where $P_1, P_2 \leftarrow \mathcal{P}_n$, $K \leftarrow \{0,1\}^n$, and H_{K_h} is an ϵ-almost-XOR-universal (ϵ-AXU) hash function.

Theorem 3. *For any quantum adversary $\mathcal{A}$ in the Q1 model making at most q_E classical queries (under the nonce-respecting restriction) and q_P quantum queries to the public permutations, the distinguishing advantage against a truly random function $R : \mathcal{N} \times \mathcal{M} \to \{0,1\}^n$ is bounded by:*

$$Adv^{Q1\text{-}PRF}_{n\mathsf{EHTm}_p}(q_E, q_P) \leq 6 \cdot \frac{q_E\sqrt{q_P} + q_P\sqrt{q_E}}{2^{n/2}} + \frac{q_E^2}{2} \cdot \epsilon + \frac{3q_E^2}{2^{n+1}}.$$

Proof. We partition the execution timeline of the adversary $\mathcal{A}$ into $q_E + 1$ sequential phases, labeled $0, \ldots, q_E$. Let $T_j = \{(N_1, M_1, y_1), \ldots, (N_j, M_j, y_j)\}$ denote the classical transcript up to the j-th query.

Unlike the symmetric reprogramming required for SoEM21, the unique nonces in $n\mathsf{EHTm}_p$ guarantee that the input coordinates $a_i = N_i \oplus K$ to P_1 are strictly distinct across all classical queries. Thus, we employ an **asymmetric reprogramming strategy**: we force P_1 to absorb the necessary randomness to match the ideal response, leaving P_2 entirely native and unperturbed.

Independence of the hash key. The hash key K_h, together with the block-cipher key K, is sampled uniformly and kept internal to the simulator. During the initial hybrid phases, classical queries are answered by the ideal random function R, which is independent of K_h. Therefore the adversary's view reveals no information about K_h in these phases. Consequently, even if the messages M_i are chosen adaptively from the previous transcript, they remain independent of K_h conditioned on the adversary's view. This allows the simulator to compute the hash values $H_{K_h}(M_i)$ required for the reprogramming procedure.

For each query $i \in \{1, \ldots, j\}$, the simulator uses its knowledge of internal keys to define the intermediate reprogramming target for P_1 as:

$$u_i = y_i \oplus P_2(N_i \oplus K \oplus H_{K_h}(M_i)).$$

We define the cumulative swap operator on P_1 as $S_{T_j,P_1,K} = \prod_{i=1}^{j} \mathrm{swap}_{a_i,z_i}$, where $z_i = P_1^{-1}(u_i)$. The updated permutation is $P_{1,T_j,K} = S_{T_j,P_1,K} \circ P_1$.

We define the primary hybrid experiments H_j and H'_j to systematically transition from the real construction to the ideal random function:

Experiment $\mathbf{H}_j$:

1. For the first j classical queries, the oracle returns $R(N_i, M_i)$.
2. From the $(j+1)$-th query onwards, classical queries are answered by F_{K,K_h}, and quantum queries are answered by the reprogrammed $P_{1,T_j,K}$ and the native P_2.

Experiment $\mathbf{H}'_j$:

1. Identical to H_j, except the ideal random function R is extended to answer the $(j+1)$-th classical query.
2. The quantum permutation P_1 is updated to $P_{1,T_{j+1},K}$ immediately after the $(j+1)$-th query.

Step 1: Bounding H'_j to H_{j+1} (Adaptive Reprogramming). The statistical distance between H'_j and H_{j+1} is governed entirely by the adversary's probability of querying the hidden coordinate $a_{j+1} = N_{j+1} \oplus K$ during phase $j+1$. Since K is uniformly random and independent, applying Lemma 1 (analogous to Lemma 3) yields:

$$\left|\Pr[\mathcal{A}(\mathrm{H}'_j) = 1] - \Pr[\mathcal{A}(\mathrm{H}_{j+1}) = 1]\right| \leq 2 \cdot q_{P,j+1}\sqrt{\frac{2(j+1)}{2^n}}.$$

Step 2: Bounding H_j to H'_j via key resampling and bad events. To bridge H_j and H'_j, we introduce intermediate hybrids H^*_j and H^{**}_j. We sample $s_0, s_1 \leftarrow \{0,1\}^n$ uniformly and define

$$K = s_0 \oplus N_{j+1}.$$

For each hybrid transition, the resampling points are chosen freshly and independently. The indistinguishability between the corresponding quantum oracle systems is then bounded using the resampling lemma, and the remaining difference between the hybrids is bounded by a union bound over the bad events.

In H^*_j, we apply Lemma 2 *only* to P_1, defining $P_1^{(1)} = P_1 \circ \mathrm{swap}_{s_0,s_1}$. Because P_2 is not resampled, the quantum indistinguishability penalty is halved compared to the SoEM21 proof:

$$\left|\Pr[\mathcal{A}(\mathrm{H}_j) = 1] - \Pr[\mathcal{A}(\mathrm{H}^*_j) = 1]\right| \leq 4\sqrt{\frac{q_P}{2^n}}.$$

Next, in H^{**}_j, we replace the $(j+1)$-st classical output with $R(N_{j+1}, M_{j+1})$. The two hybrids may differ only if one of the following bad events occurs.

Let

$$a_i = N_i \oplus K \qquad \text{and} \qquad b_i = N_i \oplus K \oplus H_{K_h}(M_i)$$

denote the input coordinates to the P_1 and P_2 branches, respectively, for the i-th classical query.

- Bad_0 (Resampling collision on the P_1 branch): the random target s_1 collides with a previous P_1-input coordinate, namely

  $$s_1 = a_i \quad \text{for some } i \leq j.$$

  Since s_1 is uniform over $\{0,1\}^n$, we have

  $$\Pr[\mathsf{Bad}_0] \leq \frac{j}{2^n}.$$

- Bad_1 (Collision on the native P_2 branch): the new P_2-input coordinate collides with a previous one, namely

  $$b_{j+1} = b_i \quad \text{for some } i \leq j.$$

  Equivalently,

  $$N_{j+1} \oplus K \oplus H_{K_h}(M_{j+1}) = N_i \oplus K \oplus H_{K_h}(M_i),$$

  or, after cancelling K,

  $$H_{K_h}(M_{j+1}) \oplus H_{K_h}(M_i) = N_{j+1} \oplus N_i.$$

  In the relevant hybrid, K_h is independent of the adversary's view. Therefore, even though the messages are chosen adaptively as a function of the previous transcript, they are independent of the hidden hash key conditioned on that view. By the ϵ-AXU property of H_{K_h},

  $$\Pr[\mathsf{Bad}_1] \leq j\epsilon.$$

- Bad_2 (Future collision): some subsequent classical query has a nonce N such that its P_1-input coordinate hits one of the hidden resampling points:

$$N \oplus K \in \{s_0, s_1\}.$$

Since there are at most $q_E - j - 1$ remaining classical queries, and s_0 and s_1 are uniform over $\{0,1\}^n$, we have

$$\Pr[\mathsf{Bad}_2] \leq \frac{2(q_E - j - 1)}{2^n}.$$

Conditioned on none of these events occurring, the oracle systems in the two hybrids coincide for the adversary's view. Hence, by the union bound,

$$\left|\Pr[\mathcal{A}(H_j^*) = 1] - \Pr[\mathcal{A}(H_j') = 1]\right| \leq \Pr[\mathsf{Bad}_0 \cup \mathsf{Bad}_1 \cup \mathsf{Bad}_2].$$

Therefore,

$$\left|\Pr[\mathcal{A}(H_j^*) = 1] - \Pr[\mathcal{A}(H_j') = 1]\right| \leq \frac{j}{2^n} + j\epsilon + \frac{2(q_E - j - 1)}{2^n}.$$

Step 3: Aggregating the Bound. Summing the statistical distances over all $j = 0, \ldots, q_E - 1$, we synthesize the global bound.

First, we perform the exact arithmetic progression summation for the classical domain collision errors triggered by the bad events:

$$\sum_{j=0}^{q_E-1} \Pr[\mathrm{Bad}_0] \leq \sum_{j=0}^{q_E-1} \frac{j}{2^n} = \frac{q_E(q_E-1)}{2 \cdot 2^n} \leq \frac{q_E^2}{2^{n+1}}.$$

$$\sum_{j=0}^{q_E-1} \Pr[\mathrm{Bad}_1] \leq \sum_{j=0}^{q_E-1} j \cdot \epsilon = \frac{q_E(q_E-1)}{2} \cdot \epsilon \leq \frac{q_E^2 \cdot \epsilon}{2}.$$

$$\sum_{j=0}^{q_E-1} \Pr[\mathrm{Bad}_2] \leq \sum_{j=0}^{q_E-1} \frac{2(q_E - j - 1)}{2^n} = \frac{2}{2^n}\frac{q_E(q_E-1)}{2} \leq \frac{q_E^2}{2^n} = \frac{2q_E^2}{2^{n+1}}.$$

Summing the contributions of Bad_0, Bad_1, and Bad_2 over all hybrid transitions, the total classical error is bounded by

$$\frac{q_E^2}{2^{n+1}} + \frac{2q_E^2}{2^{n+1}} + \frac{q_E^2\epsilon}{2} = \frac{3q_E^2}{2^{n+1}} + \frac{q_E^2\epsilon}{2}.$$

Next, for the quantum interference error aggregation, we apply the maximum weight bounding (Supremum Bound) technique to the quantum query distribution. Since the non-negative expected query weights must satisfy $\sum_{j=0}^{q_E-1} q_{P,j+1} = q_P$, and the amplitude sequence $\sqrt{j+1}$ is strictly monotonically increasing with a maximum upper bound fixed at $\sqrt{q_E}$, we derive:

$$\sum_{j=0}^{q_E-1} 2q_{P,j+1}\sqrt{\frac{2(j+1)}{2^n}} \leq 2\sqrt{\frac{2q_E}{2^n}} \sum_{j=0}^{q_E-1} q_{P,j+1} = \frac{2q_P\sqrt{2q_E}}{\sqrt{2^n}}.$$

Finally, accumulating the baseline independent resampling penalty $\sum 4\sqrt{q_P/2^n} \leq 4q_E\sqrt{q_P}/\sqrt{2^n}$ and linearly merging all mathematically verified bounds, we formally establish the tight Q1-PRF indistinguishability advantage against the ideal random function R:

$$\mathrm{Adv}^{\text{Q1-PRF}}_{n\mathsf{EHTm}_p}(q_E, q_P) \leq 6 \cdot \frac{q_E\sqrt{q_P} + q_P\sqrt{q_E}}{2^{n/2}} + \frac{q_E^2}{2} \cdot \epsilon + \frac{3q_E^2}{2^{n+1}}.$$

This proves the claimed Q1-PRF bound for the core structure of $n\mathsf{EHTm}_p$ under the nonce-respecting restriction. In the balanced-query regime, the SoEM-type term yields an $n/3$-type bound, with additional losses coming from the ϵ-AXU collision term and the classical bad-event probabilities.

□

5 Conclusion and Future Work

The advent of quantum computing necessitates a rigorous re-evaluation of symmetric cryptographic architectures. In this paper, we established the tight post-quantum security of the SoEM21 pseudorandom function within the Q1 model, capturing realistic threat scenarios where an adversary's access to the keyed primitive is classically restricted. By jointly leveraging the Reprogramming and Resampling Lemmas, we proved a tight Q1 security lower bound of $n/3$ bits, which exactly matches the known optimal quantum attack complexity. Furthermore, we demonstrated the practical impact of this theoretical foundation by extending our analysis to concrete downstream protocols. Specifically, we formally established the Q1-IND-CPA security of the 1K-CENCPP* encryption scheme and the Q1-PRF security of the $n\mathsf{EHTm}_p$ MAC core structure. These results show that permutation-based summation architectures can retain provable security guarantees under the Q1 access model considered in this work.

Several open problems remain for future work. A natural extension is to generalize our proof framework to t-branch SoEM constructions ($t > 2$) to evaluate the security scaling behavior. Additionally, investigating whether a provable security bound exists for SoEM under the strictly stronger Q2 model remains a challenging open question. Finally, applying our combined analytical methodology to other permutation-based, summation-type symmetric components could yield broader insights into the design of quantum-resilient cryptographic primitives.

References

[ABKM22a] Gorjan Alagic, Chen Bai, Jonathan Katz, and Christian Majenz. Post-quantum security of the Even-Mansour cipher. In *Advances in Cryptology – EUROCRYPT 2022*, volume 13277 of *Lecture Notes in Computer Science*, pages 458–487. Springer, 2022. doi:10.1007/978-3-031-07082-2_17.

[ABKM22b] Gorjan Alagic, Chen Bai, Jonathan Katz, and Christian Majenz. Post-quantum security of tweakable Even-Mansour, and applications. In *Advances in Cryptology – CRYPTO 2022*, volume 13509 of *Lecture Notes in Computer Science*, pages 403–432. Springer, 2022. doi:10.1007/978-3-031-15979-4_14.

[ABMS26] Gorjan Alagic, Chen Bai, Christian Majenz, and Kaiyan Shi. Post-quantum security of block cipher constructions. Cryptology ePrint Archive, Paper 2026/209, 2026. URL: https://eprint.iacr.org/2026/209.

[BDLN22] Arghya Bhattacharjee, Avijit Dutta, Eik List, and Mridul Nandi. CENCPP*: Beyond-birthday-secure encryption from public permutations. *Designs, Codes and Cryptography*, 90:1381–1425, 2022. doi:10.1007/s10623-022-01045-z.

[BHNPS19] Xavier Bonnetain, Akinori Hosoyamada, Maria Naya-Plasencia, and André Schrottenloher. Quantum attacks without superposition queries: The offline Simon's algorithm. In *Advances in Cryptology – ASIACRYPT 2019*, volume

11921 of *Lecture Notes in Computer Science*, pages 552–583. Springer, 2019. doi:10.1007/978-3-030-34578-5_20.

[BKL+12] Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, François-Xavier Standaert, John Steinberger, and Elmar Tischhauser. Key-alternating ciphers in a provable setting: Encryption using a small number of public permutations. In *Advances in Cryptology – EUROCRYPT 2012*, volume 7237 of *Lecture Notes in Computer Science*, pages 45–62. Springer, 2012. doi:10.1007/978-3-642-29011-4_5.

[CEM25] Bai Chen, Mehdi Esmaili, and Atul Mantri. Quantum security analysis of the key-alternating ciphers. Cryptology ePrint Archive, Report 2025/945, 2025. URL: https://eprint.iacr.org/2025/945.

[CLLL20] Wonseok Choi, Byeonghak Lee, Yeongmin Lee, and Jooyoung Lee. Improved security analysis for nonce-based enhanced hash-then-mask MACs. In *Advances in Cryptology – ASIACRYPT 2020*, volume 12491 of *Lecture Notes in Computer Science*, pages 697–723. Springer, 2020. doi:10.1007/978-3-030-64837-4_23.

[CLM19] Yu Long Chen, Eran Lambooij, and Bart Mennink. How to build pseudorandom functions from public random permutations. In *Advances in Cryptology – CRYPTO 2019*, volume 11692 of *Lecture Notes in Computer Science*, pages 266–293. Springer, 2019. doi:10.1007/978-3-030-26948-7_9.

[CLS15] Benoît Cogliati, Rodolphe Lampe, and Yannick Seurin. Tweaking Even-Mansour ciphers. In *Advances in Cryptology – CRYPTO 2015*, volume 9215 of *Lecture Notes in Computer Science*, pages 189–208. Springer, 2015. doi:10.1007/978-3-662-47989-6_10.

[CS14] Shan Chen and John Steinberger. Tight security bounds for key-alternating ciphers. In *Advances in Cryptology – EUROCRYPT 2014*, volume 8441 of *Lecture Notes in Computer Science*, pages 327–350. Springer, 2014. doi:10.1007/978-3-642-55220-5_18.

[DDM+26] Nilanjan Datta, Avijit Dutta, Sougata Mandal, Hrithik Nandi, and Amlan Sinha. Post-quantum security of keyed sum of permutations and its siblings. Cryptology ePrint Archive, Paper 2026/426, 2026. URL: https://eprint.iacr.org/2026/426.

[EM97] Shimon Even and Yishay Mansour. A construction of a cipher from a single pseudorandom permutation. *Journal of Cryptology*, 10(3):151–161, 1997. doi:10.1007/s001459900025.

[GHY24] Chun Guo, Anjia Huang, and Yu Yu. Quantum Q1 security proof for FX key-length extension construction. *Journal of Cryptologic Research*, 11(5):1139–1151, 2024. doi:10.13868/j.cnki.jcr.000682.

[Hos25] Akinori Hosoyamada. Post-quantum security of keyed sponge-based constructions through a modular approach. In *Advances in Cryptology – ASIACRYPT 2025*, volume 16245 of *Lecture Notes in Computer Science*, pages 411–445. Springer, 2025. doi:10.1007/978-981-95-5018-0_14.

[JST21] Jonathan Jaeger, Fang Song, and Stefano Tessaro. Quantum key-length extension. In *Theory of Cryptography – TCC 2021*, volume 13043 of *Lecture Notes in Computer Science*, pages 209–239. Springer, 2021. doi:10.1007/978-3-030-90459-3_8.

[KLLNP16] Marc Kaplan, Gaëtan Leurent, Anthony Leverrier, and María Naya-Plasencia. Breaking symmetric cryptosystems using quantum period finding. In *Advances in Cryptology – CRYPTO 2016*, volume 9815 of *Lecture Notes in Computer Science*, pages 207–237. Springer, 2016. doi:10.1007/978-3-662-53008-5_8.

[KM12] Hiroyuki Kuwakado and Masakatu Morii. Security of the quantum-type Even-Mansour cipher. In *Proceedings of the International Symposium on Information Theory and Its Applications (ISITA 2012)*, pages 312–316. IEEE, 2012. doi:10.1109/ISITA.2012.6400943.

[KN23] Akinori Kawachi and Yuki Naito. Quantum query lower bounds for key recovery attacks on the Even-Mansour cipher. In *Computing and Combinatorics – COCOON 2023*, volume 14423 of *Lecture Notes in Computer Science*, pages 3–16. Springer, 2023. doi:10.1007/978-3-031-49193-1_1.

[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. doi:10.1007/s001450010002.

[LFG+25] Zhenqiang Li, Shuqin Fan, Fei Gao, Yonglin Hao, Hongwei Sun, Xichao Hu, and Dandan Li. Quantum attacks on sum of Even-Mansour construction utilizing online classical queries. *EPJ Quantum Technology*, 12, 2025. doi:10.1140/epjqt/s40507-025-00374-x.

[LM17] Gregor Leander and Alexander May. Grover meets Simon – quantumly attacking the FX-construction. In *Advances in Cryptology – ASIACRYPT 2017*, volume 10625 of *Lecture Notes in Computer Science*, pages 161–178. Springer, 2017. doi:10.1007/978-3-319-70697-9_6.

[SI22] Kenta Shinagawa and Tetsu Iwata. Quantum attacks on sum of Even-Mansour pseudorandom functions. *Information Processing Letters*, 173:106172, 2022. doi:10.1016/j.ipl.2021.106172.

[SI26] Rentaro Shiba and Tetsu Iwata. Multi-key security in the quantum world: Revisiting tweakable Even-Mansour and FX. Cryptology ePrint Archive, Paper 2026/382, 2026. URL: https://eprint.iacr.org/2026/382.

[Zha19] Mark Zhandry. How to record quantum queries, and applications to quantum indifferentiability. In *Advances in Cryptology – CRYPTO 2019*, volume 11693 of *Lecture Notes in Computer Science*, pages 239–268. Springer, 2019. doi:10.1007/978-3-030-26954-8_9.