

Natural Barriers to Quantum Extraction: On the Post-Quantum (In)security of (O)EKE and Masny-Rindal OT

James Bartusek*

Jake Januzelli†

Abstract

Encrypted key exchange (EKE), introduced by Bellare and Merritt (IEEE S&P 1992), and Masny-Rindal OT, introduced by Masny and Rindal (ACM CCS 2019), are highly-efficient methods for compiling essentially any KEM into advanced cryptographic protocols, namely password-authenticated key exchange (PAKE) and oblivious transfer (OT), by relying only on idealized symmetric-key primitives. They have become leading candidates for practically-implementable PAKE and OT due to (1) their simplicity, (2) their plug-and-play nature, allowing for flexibility in the choice of KEM, and (3) existing proofs of UC-security (in the classical adversarial model).

Due to point (2) above, these compilers yield attractive candidates for efficient *post-quantum* PAKE and OT, especially given the recent post-quantum KEM standardization efforts. This motivates the question of whether the (UC-)security of these compilers translates to the quantum adversarial model. In this work, we show that it does not. In particular, we prove that a general family of (O)EKE protocols, as well as Masny-Rindal OT, are *not* UC-secure against quantum polynomial-time adversaries, even when instantiated with a post-quantum KEM. To establish UC-insecurity, we devise an adversarial strategy that provably thwarts any attempt by the simulator to extract its input (the password in the case of PAKE, and the receiver’s choice bit in the case of OT).

To complement these negative results, we establish that both compilers yield certain notions of *game-based* security. In the PAKE setting, we consider OEKE instantiated with the “2-Feistel” cipher, and prove its stand-alone game-based security in the quantum random oracle model. We view these results as a proof of concept that security of (O)EKE and Masny-Rindal OT can yet be redeemed in the quantum setting, though we caution that the situation will be more subtle than in the classical setting due to the breakdown of simulation-based security. Along the way, we establish a novel “advantage-tight” one-way to hiding lemma that may be of independent interest.

*Columbia University. bartusek.james@gmail.com.

†Columbia University. jj3544@columbia.edu

Contents

1	Introduction	1
2	Technical overview	4
2.1	An illustrative commitment scheme	4
2.2	Failure to quantumly extract	4
2.3	Collapse-binding	6
2.4	Applications	6
2.5	Discussion	8
2.6	Future work	8
3	Preliminaries	9
3.1	Quantum information	9
3.2	KEM	11
3.3	Masny-Rindal OT	13
3.3.1	Syntax	13
3.3.2	Game-based security	14
3.3.3	Simulation-based security	14
3.4	OEKE	15
3.4.1	Syntax	15
3.4.2	Game-based security	16
3.4.3	UC security	16
4	Attacks on simulation-based security	18
4.1	Masny-Rindal OT	18
4.2	OEKE with ideal cipher	20
5	Provable game-based security	24
5.1	Masny-Rindal OT	24
5.2	OEKE with 2-Feistel	26
6	Advantage-tight one-way to hiding	30
7	Acknowledgments	36
A	Generalizing the attack	40
B	Modified Measure-and-Reprogram	42

1 Introduction

The threat of quantum computers has precipitated widespread adoption of post-quantum cryptography in recent years. Moving from cryptographic assumptions vulnerable to Shor’s algorithm (like factoring or discrete logarithm) to post-quantum assumptions is the first step in this transition. However, this does not always suffice to establish security against quantum attackers: Classical security proofs often make additional assumptions about the adversary that can fail in the quantum setting (e.g. that one can rewind the adversary [ARU14]).

Practically every efficient post-quantum primitive relies on the Random Oracle Model (ROM) [BR93] to justify security, often through transforms like Fiat-Shamir [PS96] and Fujisaki-Okamoto [FO99]. In 2011, [BDF⁺11] introduced the Quantum Random Oracle Model (QROM), which allows quantum adversaries to query a random oracle in superposition (an ability that any real-life quantum attacker will have by implementing a reversible circuit for the hash function in question). Although many important constructions have now been shown secure in the QROM (e.g. [Zha19, DFMS19, DFM20]), these proofs require expertise in quantum information theory and can often be substantially more complex than in the ROM. In light of this, one may ask whether expending the effort to write a QROM proof is actually necessary: Are there any cryptographic protocols in use that are secure in the ROM but insecure in the QROM? The seminal works of Yamakawa and Zhandry [YZ21, YZ22] construct a variety of theoretical examples including one-way functions and public key encryption, but they are not naturally occurring, and were specifically created with this aim in mind.

In general, there appears to be a belief in the community that for any “naturally-occurring” ROM-based protocol instantiated with post-quantum assumptions, if we currently lack a QROM proof, then this likely indicates a lack of appropriate proof techniques, not an attack. As an illustrative example, a report from the European Telecommunications Standards Institute (one of three official standards organizations for the EU) refers to the [YZ21, YZ22] counterexamples as “completely artificial” and notes that

“it is often stated [DFMS19] that a concrete protocol that is secure in the ROM will remain secure in the Q-ROM (if based on quantum-safe hardness assumptions), although with no formal proof in this sense.”
[Eur24, Section 7.1]

This work. We challenge this claim, and give the first examples of *naturally occurring* cryptographic protocols based on post-quantum assumptions that are secure in the ROM but provably *not* secure in the QROM. Concretely, we show that the oblivious transfer (OT) protocol of Masny and Rindal [MR19] and the (Once) Encrypted Key Exchange, or (O)EKE [BM92, BCP03], family of protocols for Password Authenticated Key Exchange (PAKE) are insecure in the Universally Composable (UC) framework [Can01] when quantum access to the random oracle is allowed, despite being UC-secure in the ROM. As a bonus, we show the same insecurity for variants of OEKE in the Quantum Ideal Cipher Model (QICM).

We complement these negative results with positive ones, showing that both protocols do satisfy some notions of game-based security in the QROM under the same assumptions used for the classical ROM proofs. Prior to this work, there were no fully-established results on either the simulation-based or game-based security of either protocol against quantum adversaries.¹

Masny-Rindal OT. Recall that in 1 out of 2 (random) Oblivious Transfer (OT), a sender outputs two strings s_0, s_1 and a receiver outputs b, s_b for a choice bit b . Security requires the receiver to learn nothing about s_{1-b} and the sender to learn nothing about b .

In 2019, Masny and Rindal [MR19] introduced a highly efficient compiler that upgrades any Key Encapsulation Mechanism (KEM) protocol satisfying certain natural properties to a two-message random OT protocol, in the ROM.² The compiler is described in Figure 1 and can be instantiated with ML-KEM, making it the premier candidate for deployable post-quantum OT. In particular, their results establish that the

¹Although [HHKR25] attempted an analysis of the post-quantum game-based security of OEKE, their handling of the quantum-accessible ideal cipher was only conjectured, not proven.

²In fact, their techniques also yield a *simultaneous-message* OT protocol in some settings, but in this work we focus on the two-message version.

protocol is UC-secure³ in the ROM when instantiated with ML-KEM. In this work, we show that this does not hold true in the *quantum* random oracle model.

Theorem 1.1 (Informal). *Masny-Rindal OT instantiated with any KEM is not post-quantum simulation-secure.*

Our attack exploits an inability for the simulator to *extract* a malicious receiver’s choice bit. However, it does not establish that the malicious receiver can actually obtain both of the sender’s messages. Hence, it does not rule out the possibility that Masny-Rindal OT actually does satisfy some game-based notion of security. We investigate this possibility, and show that it does.

Theorem 1.2 (Informal). *Let KEM be any KEM that satisfies (computational) public-key uniformity and key unpredictability. Then Masny-Rindal OT instantiated with KEM satisfies the following claim in the QROM: The probability that any QPT malicious receiver can guess $s_{b'}$ for random b' is at most $1/2 + \text{negl}$.*

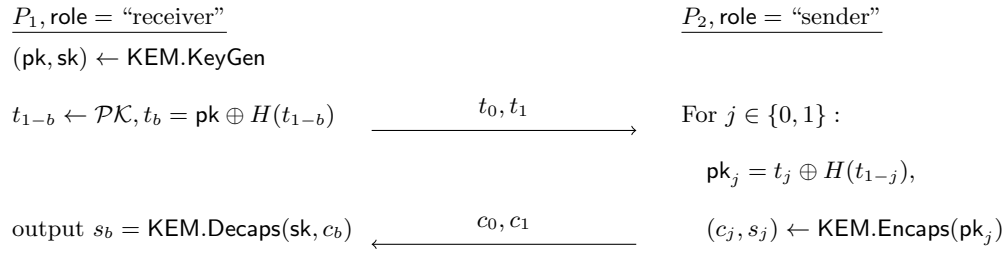


Figure 1: Masny-Rindal OT with random oracle $H : \mathcal{PK} \rightarrow \mathcal{PK}$, a KEM KEM, and receiver’s choice bit $b \in \{0, 1\}$.

PAKE. In the PAKE setting parties hold (possibly low entropy) passwords pw, pw' , and want to run an authenticated key exchange based on them: if $\text{pw} = \text{pw}'$ they agree on a shared secret key, and if not they derive independent keys (or output $\perp$). PAKE protocols offer a form of authentication without public key infrastructure, and have been deployed in password managers, end-to-end encrypted backup recovery, key escrow, and WiFi login.⁴ Although there are both game-based [BPR00] and UC [CHK⁺05] definitions of PAKE security, the UC definition has become the gold standard that deployed protocols target, due to modeling password correlation between parties and security under arbitrary composition: all Round 2 candidates of the 2020 CFRG PAKE competition have proofs of UC-security [Cry19].

We study the dominant paradigm of post-quantum PAKE: Once Encrypted Key Exchange (OEKE) (depicted in Figure 2). Although originally instantiated with an ideal cipher, recent works [MRR20, JRX25, ABJ26] opt to replace the ideal cipher with a two-round Feistel network using a random oracle. This 2-Feistel variant of OEKE is widely favored due to its optimal round complexity, efficiency, and reliance only on the ROM and a post-quantum KEM. In fact, a variant of this protocol is part of an Internet-Draft submitted to the Internet Engineering Task Force for possible standardization [VJW26]. In the corresponding paper [VJW⁺25], the authors prove their protocol is UC-secure in the (classical) ROM, conjecturing that their protocol is also secure in the QROM:

“A crucial next step for theoretical security is to prove security in the quantum random oracle model. We believe that this is possible using compressed quantum random oracles, because the simulated random oracles only rely on historical queries in simple ways.” [VJW⁺25, Section 6]

We refute this conjecture, and show that OEKE is UC-insecure in the QROM.

³The functionality that they prove security with respect to is called *endemic* OT, which is the weakest known notion of random OT that remains meaningful. Here, meaningful refers to the fact that any random OT satisfying endemic security implies a simulation-secure chosen-message OT via the natural compiler in which the sender transmits $m_0 \oplus s_0, m_1 \oplus s_1$ along with its second message, where m_0 and m_1 are its chosen strings.

⁴See <https://github.com/fancy-cryptography/fancy-cryptography> for more information.

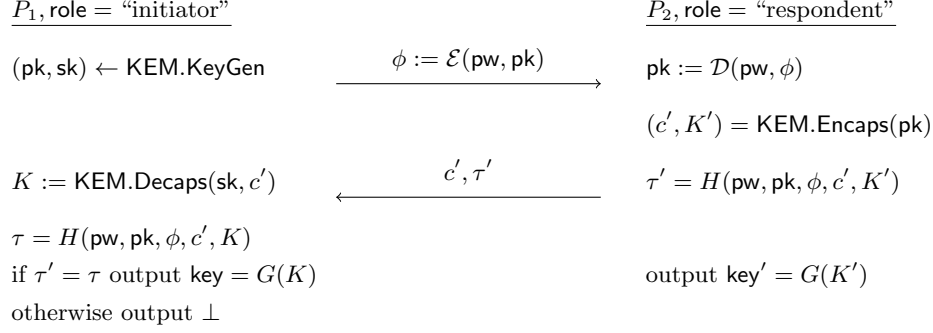


Figure 2: A typical instantiation of OEKE with shared password pw , random oracles H, G , a KEM KEM , ideal cipher encryption $\mathcal{E}$ and decryption $\mathcal{D}$.

Theorem 1.3 (Informal). *OEKE instantiated with any KEM and using either ideal cipher encryption or 2-Feistel encryption is not post-quantum UC-secure.*

In Section 4.2 we prove the attack works for the ideal cipher instantiation, as this suffices to illustrate the main idea. In Appendix A we give some details as to how it generalizes to a wide variety of OEKE variants in the QROM: this includes NoIC [ABJ25], Tempo [ABJ26], OEKE/EKE [JRX25], OQUAKE [VJW⁺25] and (O)CAKE [BCP⁺23, HHKR25].

Similar to the Masny-Rindal OT case, our attack relies on an inability for the simulator to extract (in this case, the password) from a malicious initiator. Therefore, this also leaves open the possibility of game-based security. We prove some positive results when the encryption used is 2-Feistel.⁵

Theorem 1.4 (Informal). *Let KEM be any KEM that satisfies (computational) public-key uniformity and key unpredictability. Then OEKE instantiated with KEM and using 2-Feistel encryption satisfies stand-alone game-based security in the QROM against both malicious initiators and malicious respondents.*⁶

Advantage-tight one-way to hiding. Along the way to our positive results on game-based OEKE security, we establish a new one-way to hiding lemma that to the best of our knowledge has not appeared before, and is of independent interest. One-way to hiding lemmas [Unr14, AHU19] are essentially search-to-decision reductions with respect to quantum-accessible oracles: Given a distribution over two classical functions G and H that differ on a set S , they relate the maximum probability p that an adversary can find an $x \in S$ given quantum access to G, H to the maximum distinguishing advantage f that an adversary has given access to G or H . A typical statement (e.g. [KSS⁺20, Lemma 3.3]) upper bounds f as a function of p and the number of queries q made by the adversary (or its query depth d).

Here, we show that f is upper-bounded by p *itself* (and independent of q or d), up to an additive negligible term. This statement comes with two caveats: (1) The set S must be *efficiently checkable* (though this is also required to obtain $f \leq p$ in the classical setting), and (2) the reduction (search adversary) may run in time much longer than (but still polynomial in) the decision adversary, in particular, the multiplicative blowup can depend on the number of queries q .

Nevertheless, we establish a completely “advantage-tight” one-way to hiding theorem as stated below, which turns out to be crucial for establishing the tight post-quantum game-based security of OEKE.

Theorem 1.5 (Informal). *Let D be any sampler that outputs functions H, G that differ on a set S , along with an arbitrary quantum state ρ . Suppose that for all QPT B ,*

$$\Pr_{H, G, S, \rho} [x \in S : x \leftarrow B^{H, G, M_S}(\rho)] \leq p(\lambda) + \text{negl}(\lambda),$$

where M_S is the membership-checking predicate for the set S . Then for all QPT A ,

$$\left| \Pr_{H, G, S, \rho \leftarrow D} [A^H(\rho) = 1] - \Pr_{H, G, S, \rho \leftarrow D} [A^G(\rho) = 1] \right| \leq p(\lambda) + \text{negl}(\lambda).$$

⁵We work with 2-Feistel since (1) it is more efficient and easier to instantiate than an ideal cipher, and (2) the QROM is easier to work with than the QICM.

⁶For those familiar with BPR security [BPR00] see Section 2.6.

2 Technical overview

2.1 An illustrative commitment scheme

Our main techniques can be appreciated by studying a simple bit commitment scheme, divorced from the particulars of OT and PAKE. In particular, we abstract out a commitment scheme underlying the Masny-Rindal OT construction [MR19] and show that it satisfies the following combination of properties, in addition to being computationally hiding.

1. Extractable against classical committers.
2. Computationally collapse-binding (the post-quantum analogue of standard binding).
3. Provably not extractable against quantum committers.

In this overview, we'll present the commitment scheme, argue why it is *not* quantumly-extractable, and then argue why it *is* collapse-binding. We then touch on how these proofs of non-extractability and collapse-binding are at the heart of our results showing that Masny-Rindal OT and OEKE are not simulation-secure against quantum adversaries but do satisfy some notions of game-based security.

The scheme. Let $G : \{0, 1\}^\lambda \rightarrow \{0, 1\}^n$ be an injective PRG, and let $H : \{0, 1\}^n \rightarrow \{0, 1\}^n$ be a random oracle. To commit to a bit b , sample a seed $s_b \leftarrow \{0, 1\}^\lambda$, a string $t_{1-b} \leftarrow \{0, 1\}^n$, and define $t_b := G(s_b) \oplus H(t_{1-b})$. The commitment consists of the two strings (t_0, t_1) . To open, send s_b , allowing the receiver to verify that $t_b = G(s_b) \oplus H(t_{1-b})$.

It is not hard to see that (t_0, t_1) computationally hides the committer's choice of b , since, after switching $G(s_b)$ to uniform, both t_0 and t_1 are distributed uniformly at random. Moreover, [MR19] shows as a component of their OT security proof that one can extract the bit b from any classical committer by observing the *order* in which they make their random oracle queries. In particular, the committer can “program” t_b by querying t_{1-b} first, but then $H(t_b) \oplus t_{1-b}$, which is supposed to equal $G(s_{1-b})$, becomes uncontrollable. Therefore, if t_{1-b} was queried first, the only bit that the committer can hope to open is b .

2.2 Failure to quantumly extract

The first hint that this extraction strategy may fail in the quantum setting is an inability to define “query order” in any meaningful way. Indeed, the quantum random oracle allows for superposition queries, meaning that an adversarial committer could query $H(t_0)$ and $H(t_1)$ at the same time⁷. One natural attempt to thwart the extractor is to simply run the honest committer's strategy coherently over a uniform choice of the bit b , and delay the measurement of b until the opening stage.

Definition of extractability. Before further investigating this strategy, we recall the formal notion of an extractable commitment. First, any adversarial committer $\mathcal{A}$ is split into two parts: (1) the commit stage adversary $\mathcal{A}_{\text{Com}}$ is initialized with some state ρ on register C and outputs (t_0, t_1) and an updated state ρ_C^{Real} , and (2) the opening stage adversary $\mathcal{A}_{\text{Open}}$ takes ρ_C^{Real} and outputs a bit b and a candidate opening s_b .

For any such adversary, there must exist an extractor that takes as input ρ_C and produces a bit b^* and a state ρ_C^{Ideal} satisfying the following two properties:

1. The probability that $\mathcal{A}_{\text{Open}}(\rho^{\text{Ideal}})$ produces a successful opening to $1 - b^*$ is negligible.
2. ρ_C^{Ideal} is computationally indistinguishable from the state ρ_C^{Real} output by the real committer. In fact, this must hold even in the presence of a register D that may be entangled with ρ_C . This indistinguishability in the presence of an auxiliary state (sometimes referred to as the “environment”) is a standard property in simulation-based security and we will use D in our impossibility result.

We allow the extractor to make arbitrary (potentially non-black-box) use of the code of $\mathcal{A}_{\text{Com}}, \mathcal{A}_{\text{Open}}$ and respond to the adversary's random oracle queries using any strategy.

⁷The same observation was made in [Ver22, Section 4 3.1.2].

The coherent honest committer. Now, coming back to our candidate commitment, we see that running the honest commitment strategy in equal superposition over the choice of b gives (omitting normalization)

$$\begin{aligned} & |0\rangle_{\mathbf{B}} \sum_{s_0 \in \{0,1\}^\lambda, t_1 \in \{0,1\}^n} |s_0\rangle_{\mathbf{S}} |G(s_0) \oplus H(t_1)\rangle_{\mathbf{T}_0} |t_1\rangle_{\mathbf{T}_1} \\ & + |1\rangle_{\mathbf{B}} \sum_{s_1 \in \{0,1\}^\lambda, t_0 \in \{0,1\}^n} |s_1\rangle_{\mathbf{S}} |t_0\rangle_{\mathbf{T}_0} |G(s_1) \oplus H(t_0)\rangle_{\mathbf{T}_1}. \end{aligned}$$

Measuring registers $\mathbf{T}_0, \mathbf{T}_1$ yields a commitment string (t_0, t_1) and a left-over committer state

$$\sum_{b: t_b \oplus H(t_{1-b}) \in \text{Im}(G)} |b\rangle_{\mathbf{B}} |s_b = G^{-1}(t_b \oplus H(t_{1-b}))\rangle_{\mathbf{S}},$$

which is then passed to $\mathcal{A}_{\text{Open}}$. It appears we may already have a simple counterexample to extractability: $\mathcal{A}_{\text{Open}}$ can simply measure in the standard basis to obtain a valid (b, s_b) , each with probability $1/2$ (where the probability is taken over its entire execution). So if the extractor makes a guess b^* before this point, it seems that it will be wrong with probability $1/2$, violating the property (1) above.

However, recall that the extractor has full access to the committer's state on registers $\mathbf{B}, \mathbf{S}$ and can apply an arbitrary operation before sending it to $\mathcal{A}_{\text{Open}}$, as long as this operation preserves computational indistinguishability. A natural extraction attempt would therefore have the extractor *pre-measure* this state in the standard basis, producing $|b^*, s_{b^*}\rangle$, and fixing the single choice of b^* that $\mathcal{A}_{\text{Open}}$ will be able to open. In fact, this extractor *does* satisfy property (2): In the following sub-section we will argue that no QPT adversary will be able to recover both strings s_0 and s_1 and thus, by the quantum equivalence of mapping and distinguishing,

$$\frac{1}{\sqrt{2}} (|0, s_0\rangle + |1, s_1\rangle) \approx_c \frac{1}{2} |0, s_0\rangle\langle 0, s_0| + \frac{1}{2} |1, s_1\rangle\langle 1, s_1|.$$

Hence, we will have to alter the adversarial strategy if we hope to rule out the existence of *any* extractor.

Adding a dummy branch. One way to see the issue with the above attempt is that the distinguisher does not have a computationally accessible description of the committer's state, and thus cannot detect if the extractor is messing with it. To remedy this, we consider an adversarial strategy that runs the following two strategies with $1/2$ probability: (0) a “dummy” strategy that replaces $G(s_0), G(s_1)$ with uniform strings u_0, u_1 , and (1) the coherent honest committer's strategy given above. The key is that branch (1) will enable successful opening of either $b = 0$ or $b = 1$, while branch (0) will enable the distinguisher to efficiently check that the extractor did not collapse the state. Even though successful opening and efficient checking cannot be performed simultaneously, we will be able to argue that the extractor cannot distinguish between the two branches and thus cannot tailor their strategy to one or the other. Here, we make use of the inaccessible distinguisher register $\mathbf{D}$, as follows.

Consider the following mixed state over the uniformly random choice of branch $c \in \{0, 1\}$, again omitting normalization:

$$\begin{aligned} \text{if } c = 0 : & \sum_{b \in \{0,1\}, u_b \in \{0,1\}^n, t_{1-b} \in \{0,1\}^n} |u_b\rangle_{\mathbf{D}} |b\rangle_{\mathbf{B}} |u_b, t_{1-b}\rangle_{\mathbf{T}}, \\ \text{if } c = 1 : & \sum_{b \in \{0,1\}, s_b \in \{0,1\}^\lambda, t_{1-b} \in \{0,1\}^n} |s_b\rangle_{\mathbf{D}} |b\rangle_{\mathbf{B}} |G(s_b), t_{1-b}\rangle_{\mathbf{T}}. \end{aligned}$$

Here, $\mathbf{D}$ is the register held by the distinguisher, and $\mathbf{C} = (\mathbf{B}, \mathbf{T})$ is given to the committer. Completing the honest committer's strategy coherently and measuring the resulting commitment (t_0, t_1) yields the left-over state

$$\begin{aligned} \text{if } c = 0 : & \frac{1}{\sqrt{2}} (|0, t_0 \oplus H(t_1)\rangle + |1, t_1 \oplus H(t_0)\rangle), \\ \text{if } c = 1 : & \sum_b \beta_b |b, s_b\rangle, \end{aligned}$$

for some weights β_b whose values will not be important for the argument. If $c = 0$, the distinguisher uses its knowledge of t_0, t_1 to project its remaining state onto $\frac{1}{\sqrt{2}}(|0, t_0 \oplus H(t_1)\rangle + |1, t_1 \oplus H(t_0)\rangle)$. Since this measurement accepts with probability 1 in the real execution, any valid extractor must output a state in which this measurement accepts with probability $1 - \text{negl}(\lambda)$. Then we can appeal to the pseudorandomness of G to argue that the extractor, who just sees register C , cannot distinguish between the two branches, and hence, the state on branch $c = 1$ can be undetectably replaced with the state on branch $c = 0$. At this point, measuring the B register of the state produces a uniformly random bit b , *independent* of whatever choice b^* the extractor fixed earlier in the experiment, yielding our final contradiction.

2.3 Collapse-binding

Despite the provable inability to extract an adversarial committer's bit, we show that this commitment scheme *does* retain meaningful security against quantum adversaries. In particular, the commitment satisfies collapse-binding, which is the standard notion of computational post-quantum binding.

Below, we'll just argue that it satisfies post-quantum "classical-style" binding, which stipulates that it is hard for any QPT adversary to output a commitment (t_0, t_1) along with valid openings s_0, s_1 for each bit, meaning that $G(s_0) = t_0 \oplus H(t_1)$ and $G(s_1) = t_1 \oplus H(t_0)$. Standard techniques (e.g. [DS23]) can be used to show that, since there is only one valid opening per bit,⁸ classical-style binding of this commitment suffices to establish collapse-binding, which is equivalent to the following notion of sum-binding: After the commit stage, the challenger samples a $b^* \leftarrow \{0, 1\}$, and we require that no QPT adversary can output a valid opening s_{b^*} except with probability $1/2 + \text{negl}(\lambda)$.

Measure-and-reprogram. We show classical-style binding using an appropriate application of the measure-and-reprogram technique [DFMS19, DFM20], which will allow us to *program* in an external challenge $G(s)$ for an unknown $s \leftarrow \{0, 1\}^\lambda$. If the adversary then manages to output s as one of their pair (s_0, s_1) , they immediately contradict the pseudorandomness of the PRG G .

In more detail, measure-and-reprogram considers the following scenario. An adversary queries a random oracle H and then outputs a set of k oracle inputs $\{x_i\}_{i \in [k]}$ and some auxiliary information z . Consider any predicate V over $\{x_i\}_i, \{H(x_i)\}_i$, and z , and suppose that $\Pr[V(\{x_i\}_i, \{H(x_i)\}_i, z) = 1] = \epsilon$. Then, consider a *simulated* experiment where the adversary is run as in the real experiment except that the input register to k randomly-chosen queries are fully measured in the standard basis to yield $\{x_i^*\}_i$, and the $H(x_i^*)$ are re-programmed to fresh uniformly random outputs $\{y_i^*\}_i$. Essentially (eliding certain details for simplicity of presentation), it holds that in this simulated experiment, $\Pr[V(\{x_i^*\}_i, \{y_i^*\}_i, z) = 1] \approx \frac{\epsilon}{q^{2k}}$, where q is the number of queries made by the adversary.

For arguing the classical-style binding of our commitment, we consider the following predicate: V takes as input the commitment (t_0, t_1) and the openings (s_0, s_1) and outputs 1 iff $G(s_0) = t_0 \oplus H(t_1)$ and $G(s_1) = t_1 \oplus H(t_0)$. Thus, $k = 2$ in this example. After moving to the simulated experiment, we can consider the following re-programming strategy: Upon measuring the first input t_1^* , program $H(t_1^*) = y^*$ for a uniformly random y^* , and upon measuring the second input t_2^* , program $H(t_2^*) = t_1^* \oplus G(s)$, where $G(s)$ is sampled externally and given to the simulator. The measure-and-reprogram lemma establishes that V still passes with $1/\text{poly}(\lambda)$ probability, which means that $s \in \{s_0, s_1\}$, thus contradicting the security of G .

2.4 Applications

Next, we argue how the techniques described above can be applied to both Masny-Rindal OT and OEKE to establish their simulation-based insecurity and game-based security.

Masny-Rindal OT: Simulation-based insecurity. The first message of the [MR19] protocol can be seen as taking the above commitment scheme and replacing the PRG G with the key generation function $\text{KeyGen}(1^\lambda; r) \rightarrow (\text{pk}, \text{sk})$ of a KEM. That is, the receiver samples random coins r , uses them to (deterministically) generate a public key pk , and applies the rest of the commitment algorithm on choice

⁸One could also generalize the construction to use any G that is one-way and *collapsing*, but we stick with the case of an injective PRG, which trivially satisfies both properties, for the purpose of this overview.

bit b and string $\mathbf{pk}$ to obtain (t_0, t_1) . Given (t_0, t_1) , the OT sender derives the two possible public keys $\mathbf{pk}_0 = t_0 \oplus H(t_1)$, $\mathbf{pk}_1 = t_1 \oplus H(t_0)$ and then generates and returns ciphertexts (c_0, c_1) encrypting output keys K_0, K_1 .

The impossibility of extracting a malicious receiver’s choice bit b^* follows essentially the same argument as outlined above for the commitment scheme, with the function $\text{KeyGen} : r \rightarrow (\mathbf{pk}, *)$ in place of the PRG $G : s \rightarrow G(s)$. Arguing that no simulator can distinguish between the receiver’s real and dummy branches still requires assuming that $\mathbf{pk}$ is computationally indistinguishable from random. However, this assumption comes for free: If $\mathbf{pk}$ was distinguishable from uniform, then the OT protocol would be insecure against a malicious *sender*. Indeed, the sender can compute the two candidate public keys $\mathbf{pk}_0, \mathbf{pk}_1$, where $\mathbf{pk}_b$ is generated by KeyGen , and $\mathbf{pk}_{1-b}$ is sampled uniformly at random. Distinguishing $\mathbf{pk}_b$ from random with any non-negligible advantage thus gives the same advantage in guessing the honest receiver’s choice bit b .

Masny-Rindal OT: Game-based security. Our argument for the game-based security of the protocol broadly follows the measure-and-reprogram argument introduced above. Against a malicious receiver, we ask that the following game cannot be won with probability better than $1/2 + \text{negl}(\lambda)$: After the honest sender returns c_0, c_1 , the challenger samples a bit $b^* \leftarrow \{0, 1\}$ and the receiver succeeds if they manage to output a correct guess for K_{b^*} . Note that this can be seen as sum-binding for a commitment where K_{b^*} is regarded as the valid opening. In a similar manner as above, we can use measure-and-reprogram to reduce to the security of the KEM, though the game is not simply inverting the function KeyGen . In particular, the reduction will ultimately need to take $\mathbf{pk}_{b^*}$ and a ciphertext c_{b^*} from its external challenger, where c_{b^*} must be transmitted to the adversary as a part of the sender’s OT message. This actually introduces a slight subtlety in the argument: The reprogramming of the random oracle H must occur *before* this message is sent. Since the adversary may continue to make random oracle queries after the message, we appeal to a version of measure-and-reprogram in which the predicate V may also make oracle queries to H , and we refer the reader to Section 5.1 for further details.

Encrypted key exchange: Simulation-based insecurity. To generalize further to the setting of (O)EKE, we first abstract out an invertible “cipher” underlying our commitment and the [MR19] OT protocol. Given a bit b and the public key $\mathbf{pk}$, the receiver applies the (randomized) function

$$\mathcal{E} : (b, \mathbf{pk}) \rightarrow (t_0, t_1) \quad \text{where } t_{1-b} \leftarrow \{0, 1\}^n, t_b = \mathbf{pk} \oplus H(t_{1-b}).$$

Although $\mathcal{E}$ is randomized, there exists a deterministic inversion operation $\mathcal{D}$ that given any choice of $b \in \{0, 1\}$, returns the corresponding input $\mathbf{pk}$:

$$\mathcal{D} : (b, (t_0, t_1)) \rightarrow \mathbf{pk} \quad \text{where } \mathbf{pk} = t_b \oplus H(t_{1-b}).$$

In OEKE, a KEM public key $\mathbf{pk}$ is enciphered not under a single bit b , but under a (multi-bit) *password* $\mathbf{pw}$. While the above choice of $(\mathcal{E}, \mathcal{D})$ does not efficiently generalize to multi-bit passwords, researchers have considered other methods that do. Perhaps most simply, one can model $(\mathcal{E}, \mathcal{D})$ as an ideal cipher, and have the PAKE initiator send $\phi = \mathcal{E}(\mathbf{pw}, \mathbf{pk})$ and the respondent recover $\mathbf{pk} = \mathcal{D}(\mathbf{pw}, \phi)$. A simple and highly-efficient alternative that has been a focus of recent study instantiates $(\mathcal{E}, \mathcal{D})$ as a 2-Feistel, which directly uses a random oracle H as follows.

$$\mathcal{E}(\mathbf{pw}, \mathbf{pk}) \rightarrow (T, r \oplus H(\mathbf{pw}, T)) \quad \text{where } r \leftarrow \{0, 1\}^n, T = H(\mathbf{pw}, r) \oplus \mathbf{pk}.$$

Notice that, given $\mathbf{pw}$, $\mathcal{D}$ can recover the randomness r used by $\mathcal{E}$, which then allows it to recover $\mathbf{pk}$. And as mentioned in the introduction, there are other tweaks to $(\mathcal{E}, \mathcal{D})$ that have been proposed in the literature, for instance the Tempo protocol [ABJ26].

(Un)fortunately, our attack on extractability is essentially agnostic to the choice of $(\mathcal{E}, \mathcal{D})$ utilized by the protocol (as long as $\mathcal{D}$ can recover any randomness sampled by $\mathcal{E}$), and we show that for all known variants, there does not exist a simulator that can extract the choice of $\mathbf{pw}$ from a malicious initiator, invalidating claims of post-quantum UC-security. Similar to our negative results for Masny-Rindal OT, our impossibility here holds for *any* choice of KEM: While the impossibility of extraction requires the KEM to have pseudorandom public keys, this assumption is already required to obtain security against offline password guessing attacks.

Encrypted key exchange: Game-based security. We next study the post-quantum *game-based* security of OAKE, which has thus far remained an open question, even when instantiated with a post-quantum KEM. In this work, we focus on the 2-Feistel variant, and consider security in the quantum random oracle model. We separately model security against malicious initiators and malicious respondents and in this overview, we’ll focus on the more challenging case of malicious initiators. The security definition we target is as follows. Suppose the honest respondent samples their password $\text{pw} \leftarrow D$ from some distribution D with min-entropy δ . Then the best that we can hope for is that any malicious initiator, after interacting with the respondent, can distinguish the respondent’s key from a uniformly random string with advantage at most $2^{-\delta} + \text{negl}(\lambda)$. This captures the intuition that the best an adversary can do is make a single password guess pw' and, if $\text{pw}' \neq \text{pw}$, they will end up with $\text{negl}(\lambda)$ information about the honest party’s key.

We begin by applying a similar measure-and-reprogram technique as above (though this requires extra care due to the structure of the 2-Feistel) to establish the following claim: Suppose a malicious initiator after sending its first message ϕ is challenged with two different respondent messages c_0, c_1 encrypted using two different passwords pw_0, pw_1 and yielding respondent keys K_0, K_1 . Then, the probability that the initiator can output *both* K_0, K_1 is $\text{negl}(\lambda)$.

In order to translate this $\text{negl}(\lambda)$ search security into a $2^{-\delta} + \text{negl}(\lambda)$ distinguishing bound, we apply two quantum information techniques:

- We derive a generalization of the mapping-distinguishing bound from [DS23] that allows us to reason about the following scenario. Suppose that after ϕ is sent by the malicious initiator, a password $\text{pw} \leftarrow D$ is sampled and used to produce respondent message c and output K . By sampling *two* independent passwords pw_0, pw_1 from D and then *rewinding* the initiator, we can turn any initiator that predicts K with better than $2^{-\delta} + \text{negl}(\lambda)$ probability into one that can find both K_0 and K_1 with better than $\text{negl}(\lambda)$ probability.
- We next have to translate the search bound of $2^{-\delta} + \text{negl}(\lambda)$ into (ideally) the *same* distinguishing bound $2^{-\delta} + \text{negl}(\lambda)$. In order to do so, we apply our advantage-tight one-way to hiding theorem (Theorem 6.1), which builds on the “measure-rewind-measure” technique of [KSS⁺20] and quantum rewinding [GSLW19]. We defer a more in-depth overview of this theorem to Section 6.

2.5 Discussion

Broadly speaking, our work adds to the growing list of methods that are known to separate the quantum and classical adversarial models, even under post-quantum assumptions, following e.g. [YZ21, YZ22, LMQW22]. A major contribution of this work is that we did not design a (contrived) protocol, but showed that the security of practically-relevant PAKE and OT proposals fails to translate to the post-quantum setting.

For example, one could likely construct a contrived commitment scheme that is extractable against classical adversaries but not against quantum adversaries by incorporating a proof of quantumness protocol (e.g. [BCM⁺18, YZ22]) into the design of the commitment. However, we show that there is *already* a natural and widely used commitment scheme admitting such a separation.

We believe that this adds even more weight to the argument that proofs of security in the *classical* random oracle model (even from post-quantum assumptions) have limited meaning against quantum adversaries, and that the quantum adversarial model must be taken seriously when designing protocols based on post-quantum assumptions. In particular, it is crucial to gain a better understanding of the post-quantum security of (O)EKE as the community continues with standardization efforts.

2.6 Future work

Our work raises several directions for further research, a few of which we describe here.

- **Consequence of our separation.** While we exhibit a formal breakdown of simulation-based security for (O)EKE and Masny-Rindal OT, our attacks don’t have immediately “obvious” consequences. Indeed, we show that the protocols do retain some notions of game-based security. It would be interesting to find examples of broader contexts in which using (O)EKE or Masny-Rindal OT as a building block results in clear security issues stemming from the inability to extract the PAKE or OT input from the adversary.

- **Alternative constructions.** Is there a practically-efficient and truly post-quantum alternative to the “query-order” extraction technique underlying Masny-Rindal OT and (O)EKE? In fact, can we come up with *any* practically-efficient compiler from KEM to (two-message) OT or PAKE in the quantum random oracle model that achieves simulation-based security?
- **Post-quantum BPR security.** While we establish stand-alone game-based security for OEKE, the gold standard for game-based PAKE security considers a man-in-the-middle attacker and the fully *concurrent* setting, following the definitions of [BPR00]. We leave the full BPR security of (2-Feistel-based) OEKE as an important direction for future work.
- **Security in the quantum ideal cipher model.** One can also ask about the post-quantum game-based security of (O)EKE when instantiated with an *ideal cipher*, especially given recent techniques for analyzing quantum-accessible random permutations (e.g. [CHL⁺25, Car26, CM26]).

3 Preliminaries

3.1 Quantum information

We denote quantum registers A with sans-serif font. A quantum state ρ_A on register A is a positive semi-definite operator with trace 1. The reduced state $\rho_A = \text{Tr}_B(\rho_{AB})$, where $\text{Tr}(\cdot)$ is the partial trace operator. We denote pure states as $|\psi\rangle$ with corresponding density operator $|\psi\rangle\langle\psi|$. We let I denote the identity matrix. The trace distance between two state ρ, σ is defined as

$$\text{TD}(\rho, \sigma) = \frac{1}{2} \|\rho - \sigma\|_1.$$

Quantum information lemmas. We first state the Gentle Measurement lemma.

Lemma 3.1. *Let ρ be a quantum state and let $(\Pi, I - \Pi)$ be a projective measurement such that $\text{Tr}(\Pi\rho) \geq 1 - \delta$. Let*

$$\rho' = \frac{\Pi\rho\Pi}{\text{Tr}(\Pi\rho)}$$

be the state after applying $(\Pi, I - \Pi)$ to ρ and post-selecting on obtaining the first outcome. Then, $\text{TD}(\rho, \rho') \leq 2\sqrt{\delta}$.

We will need the Cauchy-Schwarz inequality with respect to the Hilbert-Schmidt inner product $\langle A, B \rangle = \text{Tr}(A^\dagger B)$.

Lemma 3.2. *If A, B are complex matrices s.t $A^\dagger B$ is defined,*

$$|\text{Tr}(A^\dagger B)|^2 \leq \text{Tr}(A^\dagger A) \text{Tr}(B^\dagger B).$$

We use the following quantum rewinding statement, which says if two different computations (on the same state) give specific outputs with sufficiently high probability, performing the computations sequentially obtains both outputs with noticeable probability.

Lemma 3.3 ([DS23, Lemma 36]). *Let P, Q be projectors and ρ any quantum state, with $\epsilon = \text{Tr}(P\rho) + \text{Tr}(Q\rho) - 1 \geq 0$. Then*

$$\frac{\epsilon^2}{4} \leq \text{Tr}(PQP\rho).$$

We also need our own generalization of the above to n different computations chosen according to a distribution D .

Lemma 3.4. *Let $n \geq 2$, let D be a (non-point-mass) distribution on $[n]$ with min-entropy γ , define D^2 to be the distribution on $[n] \times [n]$ defined by sampling $i \leftarrow D$ and then $j \leftarrow D$ conditioned on $j \neq i$, let ρ be any state, and let $P_1, \dots, P_n$ be projectors with*

$$\mathbb{E}_{i \leftarrow D} [\text{Tr}(P_i \rho)] = 2^{-\gamma} + \epsilon$$

for $\epsilon \geq 0$. Then

$$\mathbb{E}_{(i,j) \leftarrow D^2} [\text{Tr}(P_i P_j P_i \rho)] \geq \epsilon^3.$$

Proof. We follow the outline of the proof of Lemma 3.3 in [DS23] and adapt it to our setting. Define $p_i := \Pr_{j \leftarrow D}[j = i]$ and

$$s := \mathbb{E}_{i \leftarrow D} [\text{Tr}(P_i \rho)] = \sum_i p_i \text{Tr}(P_i \rho) = 2^{-\gamma} + \epsilon.$$

By applying Lemma 3.2 with $A^\dagger = \sum_i p_i P_i \sqrt{\rho}$, $B = \sqrt{\rho}$ and using $\text{Tr}(\sqrt{\rho}^\dagger \sqrt{\rho}) = \text{Tr}(\rho) = 1$, we have that

$$\begin{aligned} s^2 &\leq \text{Tr} \left(\left(\sum_i p_i P_i \right) \rho \left(\sum_i p_i P_i \right) \right) \\ &= \sum_{i,j} p_i p_j \text{Tr}(P_i \rho P_j) \\ &= \sum_i p_i^2 \text{Tr}(P_i \rho P_i) + \sum_{i \neq j} p_i p_j \text{Tr}(P_i \rho P_j). \end{aligned}$$

Since P_i is a projector, Tr is cyclic, and $p_i \leq 2^{-\gamma}$ for every i ,

$$\sum_i p_i^2 \text{Tr}(P_i \rho P_i) = \sum_i p_i^2 \text{Tr}(P_i \rho) \leq 2^{-\gamma} s,$$

so rearranging gives

$$s\epsilon = s^2 - 2^{-\gamma} s \leq \sum_{i \neq j} p_i p_j \text{Tr}(P_i \rho P_j) \leq \sum_{i \neq j} p_i p_j \sqrt{\text{Tr}(P_i P_j P_i \rho) \text{Tr}(P_j \rho)},$$

where the second inequality follows by taking $A^\dagger = P_j \sqrt{\rho}$ and $B = \sqrt{\rho} P_i P_j$ in Lemma 3.2. Applying Cauchy-Schwarz again, we obtain

$$\begin{aligned} s^2 \epsilon^2 &\leq \left(\sum_{i \neq j} p_i p_j \text{Tr}(P_i P_j P_i \rho) \right) \left(\sum_{i \neq j} p_i p_j \text{Tr}(P_j \rho) \right) \\ &\leq s \left(\sum_{i \neq j} p_i p_j \text{Tr}(P_i P_j P_i \rho) \right), \end{aligned}$$

so we can conclude that

$$\sum_{i \neq j} p_i p_j \text{Tr}(P_i P_j P_i \rho) \geq s \epsilon^2 \geq \epsilon^3.$$

Finally, by definition of D^2 ,

$$\begin{aligned} \mathbb{E}_{(i,j) \leftarrow D^2} [\text{Tr}(P_i P_j P_i \rho)] &= \sum_{i \neq j} \frac{p_i p_j}{1 - p_i} \text{Tr}(P_i P_j P_i \rho) \\ &\geq \sum_{i \neq j} p_i p_j \text{Tr}(P_i P_j P_i \rho) \\ &\geq \epsilon^3. \end{aligned}$$

□

Measure-and-reprogram. Recall the well-known measure-and-reprogram technique introduced in [DFMS19] and expanded in [DFM20]. The idea is as follows: suppose $\mathcal{A}^H$ makes quantum queries to a random oracle H and outputs x and (possibly quantum) z such that $V(x, H(x), z) = 1$ where V is some predicate. Then there is a (black box) simulator $\mathcal{S}$ such that, given access to $\mathcal{A}$, and a random Θ , measures a random H query of $\mathcal{A}$ to get x and reprograms $H(x) = \Theta$. The guarantee is that with noticeable probability, $\mathcal{S}^{\mathcal{A}}$ will output x, z such that $V(x, \Theta, z) = 1$. For reprogramming t queries, the simulator $\mathcal{S}$ works in multiple stages: it first outputs a permutation π of $\{0, \dots, t-1\}$ and $x_{\pi(0)}$, and takes an input $\Theta_{\pi(0)}$. It then runs, and outputs $x_{\pi(1)}$ and takes as input $\Theta_{\pi(1)}$, etc., until it outputs $(\pi, \pi(\mathbf{x}), z)$. Formally:

Theorem 3.5. *There exists a black-box polynomial time $(t+1)$ -stage quantum algorithm $\mathcal{S}$ with syntax as already outlined, with the following property. Let $\mathcal{A}$ be a quantum oracle algorithm that makes q queries to a random oracle $H : \mathcal{X} \rightarrow \mathcal{Y}$ and that outputs $(\mathbf{x} = (x_0, \dots, x_{t-1}), z)$. Then for any fixed $\mathbf{x}^*$ with distinct entries and (possibly quantum/randomized) predicate V ,*

$$\begin{aligned} & \Pr_{\Theta}[\mathbf{x} = \mathbf{x}^* \wedge V^{H(\mathbf{x} * \Theta)}(\mathbf{x}, \Theta, z) = 1 : (\pi, \pi(\mathbf{x}), z) \leftarrow \langle \mathcal{S}^{\mathcal{A}}, \pi(\Theta) \rangle] \\ & \geq \frac{1}{(2q+1)^{2t}} \Pr_H[\mathbf{x} = \mathbf{x}^* \wedge V^H(\mathbf{x}, H(\mathbf{x}), z) = 1 : (\mathbf{x}, z) \leftarrow \mathcal{A}^H] \end{aligned}$$

Here $\Theta = (\Theta_0, \dots, \Theta_{t-1})$ are chosen uniformly at random from $\mathcal{Y}$.

Theorem 3.5 is identical to [DFM20, Theorem 6] except that in our version V has access to H in the original game and the reprogrammed oracle $H(\mathbf{x} * \Theta)$ in the simulated game. The proof is very similar to [DFM20, Theorem 6] and is sketched in Appendix B.⁹

We also use a result of [DFMS22]: Given an algorithm $\mathcal{A}$ with access to a random oracle $H : \mathcal{X} \rightarrow \mathcal{Y}$ that eventually outputs a value $t = H(x)$ for some unknown x , there is a simulator $\mathcal{S}$ that simulates access to H using an interface $\mathcal{S}.RO$, and allows extraction of x given t via an interface $x = \mathcal{S}.E(t)$.

Theorem 3.6 (Quantum random oracle simulation with extraction [DFMS22]). *There is an efficient quantum algorithm $\mathcal{S}$ with two interfaces $\mathcal{S}.RO, \mathcal{S}.E$ such that for any algorithm $\mathcal{A}^H$ with quantum access to a random oracle $H : \mathcal{X} \rightarrow \mathcal{Y}$, we have the following guarantees;*

1. [DFMS22, Theorem 4.3 1] *If no $\mathcal{S}.E$ query is made, $\mathcal{S}.RO$ is perfectly indistinguishable from H .*
2. [DFMS22, Theorem 4.3 2(c)] *Any two independent queries to $\mathcal{S}.RO, \mathcal{S}.E$ $8\sqrt{2/|\mathcal{Y}|}$ -almost commute (independent means they could be made in either order).*
3. [DFMS22, Proposition 4.5] *Suppose $\mathcal{A}$ makes q queries to H . Then*

$$\Pr_{\substack{x, t \leftarrow \mathcal{A}^{\mathcal{S}.RO} \\ h \leftarrow \mathcal{S}.RO(x) \\ x' \leftarrow \mathcal{S}.E(t)}}[x' \neq x \wedge h = t] \leq O\left(\frac{q^3}{|\mathcal{Y}|}\right).$$

3.2 KEM

A key encapsulation mechanism

$$\text{KEM} = (\text{KEM.KeyGen}, \text{KEM.Encaps}, \text{KEM.Decaps})$$

consists of three polynomial time algorithms:

- $\text{KEM.KeyGen}(1^\lambda; r) \rightarrow \text{pk}, \text{sk}$ is a randomized algorithm that generates a public key $\text{pk} \in \mathcal{PK}_\lambda$ and a secret key $\text{sk} \in \mathcal{SK}_\lambda$. It takes an input the security parameter 1^λ and we will sometimes be explicit about the random coins $r \in \mathcal{R}_\lambda$ that it uses. We will sometimes denote its output by $(\text{pk}(r), \text{sk}(r)) = \text{KEM.KeyGen}(r)$, leaving the security parameter implicit.

⁹The coherent measure-and-reprogram theorem proved in [CGLS24] already allows V to access H , but we opt not to use it, as the structure of Theorem 3.5 is more suited to our needs.

- $\text{KEM.Encaps}(\text{pk}) \rightarrow c, K$ is a randomized algorithm that, given a public key pk , generates a ciphertext c and output key K .
- $\text{KEM.Decaps}(\text{sk}, c) = K$ is a deterministic algorithm that, given a ciphertext c and secret key sk , generates an output key K .

In what follows, we will often leave the security parameter λ implicit, for example when writing the sets $\mathcal{PK}$, $\mathcal{SK}$, and $\mathcal{R}$. $P[i]$ denotes the (zero-indexed) i -th element of the tuple P .

Definition 3.7 (KEM: Correctness). *A KEM KEM has correctness error $\delta(\lambda)$ if*

$$\Pr \left[\text{KEM.Decaps}(\text{sk}, c) \neq K : \begin{array}{l} (\text{pk}, \text{sk}) \leftarrow \text{KEM.KeyGen}(1^\lambda) \\ (c, K) \leftarrow \text{KEM.Encaps}(\text{pk}) \end{array} \right] \leq \delta(\lambda).$$

We say that the KEM is correct if $\delta(\lambda) = \text{negl}(\lambda)$.

We next define several notions of security for KEM that will be used in this work.

Public-key uniformity. The post-quantum computational public key uniformity experiment for KEM consists of the following security game played against a quantum polynomial time adversary $\mathcal{A}$.

$\text{PKU}_{\text{KEM}, \lambda}^{\mathcal{A}}$

- $\text{pk}_0 \leftarrow \mathcal{PK}$
- $(\text{pk}_1, *) = \text{KEM.KeyGen}(1^\lambda, r); r \leftarrow \mathcal{R}$
- $c \leftarrow \{0, 1\}$
- $c' \leftarrow \mathcal{A}(1^\lambda, \text{pk}_c)$

Define the advantage of $\mathcal{A}$ in the above game as

$$\text{Adv}(\text{PKU}_{\text{KEM}, \lambda}^{\mathcal{A}}) = |\Pr[c' = 1 | c = 0] - \Pr[c' = 1 | c = 1]|.$$

Definition 3.8 (KEM: Public key uniformity). *A KEM KEM has post-quantum computationally uniform public keys if for any QPT adversary $\mathcal{A}$,*

$$\text{Adv}(\text{PKU}_{\text{KEM}, \lambda}^{\mathcal{A}}) \leq \text{negl}(\lambda).$$

By purifying the randomness of the challenger, we arrive at an equivalent definition of the security game that will be more convenient for us to work with.

$\text{PKU}_{\text{KEM}, \lambda}^{\mathcal{A}}$

- $|\mu_0\rangle = \frac{1}{\sqrt{|\mathcal{PK}|}} \sum_{\text{pk} \in \mathcal{PK}} |\text{pk}\rangle_{X_0} |\text{pk}\rangle_{Y_0}$
- $|\mu_1\rangle = \frac{1}{\sqrt{|\mathcal{R}|}} \sum_{r \in \mathcal{R}} |r\rangle_{X_1} |\text{pk}(r)\rangle_{Y_1}$
- $c \leftarrow \{0, 1\}$
- $c' \leftarrow \mathcal{A}(1^\lambda, Y_c)$

Key unpredictability. Consider the following experiment.

$\text{UNP}_{\text{KEM},\lambda}^{\mathcal{A}}$

- $(\text{pk}, \text{sk}) \leftarrow \text{KEM.KeyGen}(1^\lambda)$
- $(c, K) \leftarrow \text{KEM.Encaps}(\text{pk})$
- $K' \leftarrow \mathcal{A}(1^\lambda, \text{pk}, c)$

Define the advantage of $\mathcal{A}$ in the above game as

$$\text{Adv}(\text{UNP}_{\text{KEM},\lambda}^{\mathcal{A}}) = \Pr[K' = K].$$

Definition 3.9 (KEM: Unpredictability Security). *A KEM has post-quantum key unpredictability if for any QPT adversary $\mathcal{A}$,*

$$\text{Adv}(\text{UNP}_{\text{KEM},\lambda}^{\mathcal{A}}) \leq \text{negl}(\lambda).$$

Key indistinguishability. Consider the following experiment.

$\text{IND}_{\text{KEM},\lambda}^{\mathcal{A}}$

- $(\text{sk}, \text{pk}) \leftarrow \text{KEM.KeyGen}(1^\lambda)$
- $(c, K_0) \leftarrow \text{KEM.Encaps}(\text{pk})$
- $K_1 \leftarrow \{0, 1\}^\lambda$
- $b \leftarrow \{0, 1\}$
- $b' \leftarrow \mathcal{A}(1^\lambda, \text{pk}, c, K_b)$

Define the advantage of $\mathcal{A}$ in the above game as

$$\text{Adv}(\text{IND}_{\text{KEM},\lambda}^{\mathcal{A}}) = |\Pr[b' = 1 | b = 0] - \Pr[b' = 1 | b = 1]|.$$

Definition 3.10 (KEM: Indistinguishability Security). *A KEM has post-quantum key indistinguishability if for any QPT adversary $\mathcal{A}$,*

$$\text{Adv}(\text{IND}_{\text{KEM},\lambda}^{\mathcal{A}}) \leq \text{negl}(\lambda).$$

3.3 Masny-Rindal OT

3.3.1 Syntax

Masny-Rindal (MR) OT [MR19] is a compiler from KEM to oblivious transfer in the random oracle model. Let $\text{KEM} = (\text{KEM.KeyGen}, \text{KEM.Encaps}, \text{KEM.Decaps})$, and let $H : \mathcal{PK} \rightarrow \mathcal{PK}$, where $\mathcal{PK}$ is the public key space of KEM. Assume for simplicity that elements of $\mathcal{PK}$ are naturally specified as bitstrings so that $\oplus$ is an additive operation over the group $\mathcal{PK}$. The OT protocol is specified as follows.

- $\text{Rec}_1(1^\lambda, b)$:
 - Sample $t_{1-b} \leftarrow \mathcal{PK}$, and $(\text{pk}_b, \text{sk}_b) \leftarrow \text{KEM.KeyGen}(1^\lambda)$.
 - Set $t_b := \text{pk}_b \oplus H(t_{1-b})$.
 - Send $\text{msg}_R := (t_0, t_1)$ to the sender and keep $\text{sk} := (b, \text{sk}_b)$.
- $\text{Sen}(\text{msg}_R)$:
 - Set $\text{pk}_0 := t_0 \oplus H(t_1)$ and $\text{pk}_1 := t_1 \oplus H(t_0)$.
 - Sample $c_0, K_0 \leftarrow \text{KEM.Encaps}(\text{pk}_0)$ and $c_1, K_1 \leftarrow \text{KEM.Encaps}(\text{pk}_1)$.
 - Send $\text{msg}_S := (c_0, c_1)$ to the receiver and output $m_0 := K_0, m_1 := K_1$.
- $\text{Rec}_2(\text{sk}, \text{msg}_S)$:
 - Output $m = \text{KEM.Decaps}(c_b, \text{sk}_b)$.

3.3.2 Game-based security

We formulate a notion of game-based security for two-message oblivious transfer.

Definition 3.11 (Two-message OT: Correctness). *A two-message OT protocol $(\text{Rec}_1, \text{Sen}, \text{Rec}_2)$ satisfies correctness if for any $b \in \{0, 1\}$,*

$$\Pr \left[\begin{array}{l} \text{sk}, \text{msg}_R \leftarrow \text{Rec}_1(1^\lambda, b) \\ m = m_b : (m_0, m_1), \text{msg}_S \leftarrow \text{Sen}(\text{msg}_R) \\ m \leftarrow \text{Rec}_2(\text{sk}, \text{msg}_S) \end{array} \right] = 1 - \text{negl}(\lambda).$$

Definition 3.12 (Two-message OT: Security against malicious receiver). *A two-message OT protocol $(\text{Rec}_1, \text{Sen}, \text{Rec}_2)$ satisfies security against malicious receiver if for any QPT $\mathcal{A}$,*

$$\Pr \left[\begin{array}{l} \text{msg}_R, \sigma \leftarrow \mathcal{A}(1^\lambda) \\ m^* = m_b : (m_0, m_1), \text{msg}_S \leftarrow \text{Sen}(\text{msg}_R) \\ b \leftarrow \{0, 1\} \\ m^* \leftarrow \mathcal{A}(\text{msg}_S, b, \sigma) \end{array} \right] \leq \frac{1}{2} + \text{negl}(\lambda).$$

Definition 3.13 (Two-message OT: Security against malicious sender). *A two-message OT protocol $(\text{Rec}_1, \text{Sen}, \text{Rec}_2)$ satisfies security against malicious sender if for any QPT $\mathcal{A}$,*

$$\left| \Pr \left[\begin{array}{l} b \leftarrow \{0, 1\} \\ b^* = b : \text{sk}, \text{msg}_R \leftarrow \text{Rec}_1(1^\lambda, b) \\ b^* \leftarrow \mathcal{A}(\text{msg}_R) \end{array} \right] - \frac{1}{2} \right| = \text{negl}(\lambda).$$

3.3.3 Simulation-based security

[MR19] define the following ideal functionality $\mathcal{F}_{\text{OT}}$, which they call *endemic* security.

$\mathcal{F}_{\text{OT}}$:

- Take an input $b \in \{0, 1\}$ from the receiver.
- If the sender is corrupt, take an input m_0, m_1 from the sender. If the receiver is corrupt, take an additional input m_b from the receiver.
- All undefined m are sampled uniformly from $\{0, 1\}^\lambda$.
- Output m_b to the receiver and (m_0, m_1) to the sender.

Definition 3.14 (Two-message OT: Simulation security). *A two-message OT protocol $(\text{Rec}_1, \text{Sen}, \text{Rec}_2)$ is simulation-secure against a malicious receiver if for any family of states $|\psi\rangle_{D,A} = \{|\psi_\lambda\rangle\}_\lambda$ on two registers D, A and any QPT adversary $\mathcal{A}$, there exists a QPT simulator $\mathcal{S}$ such that for any QPT distinguisher $\mathcal{D}$,*

$$|\Pr[\mathcal{D}(D, \langle \mathcal{A}(A), \text{Sen} \rangle) = 1] - \Pr[\mathcal{D}(D, \langle \mathcal{S}(A), \mathcal{F}_{\text{OT}} \rangle) = 1]| = \text{negl}(\lambda),$$

where

- $\langle \mathcal{A}(A), \text{Sen} \rangle$ denotes the interaction between the adversary on input register A and the honest sender, and outputs $\mathcal{A}$'s output along with Sen 's output m_0, m_1 .
- $\langle \mathcal{S}(A), \mathcal{F}_{\text{OT}} \rangle$ denotes the interaction between the simulator on input register A and the ideal functionality $\mathcal{F}_{\text{OT}}$. The output includes $\mathcal{A}(A)$'s output along with the sender's part (m_0, m_1) of $\mathcal{F}_{\text{OT}}$'s output.

The protocol is simulation-secure against a malicious sender if the analogous claim holds when the adversary corrupts the sender.

Simulation of ideal functionalities. One clarification is in order about the above definition. Since the protocol we will be analyzing involves an ideal functionality, namely a random oracle, we will allow the simulator to maintain the interface of the random oracle. That is, whenever the adversary or the distinguisher queries the random oracle, the simulator may process the query and respond. In the quantum setting, an ideal functionality is specified by a channel applied to input and output registers X, Y supplied by the adversary. In this case, the simulator may take the registers X, Y , apply some operation (potentially also on its private register), and return X, Y .

Classical UC security. In Section 4 and Appendix E, [MR19] show that the Masny-Rindal OT protocol is UC-secure against *classical* adversaries when instantiated with any KEM that has negligible correctness error (Definition 3.7), computationally uniform public keys (Definition 3.8), and key indistinguishability (Definition 3.10). As a consequence, this establishes that the protocol is stand-alone simulation-secure according to our above definition restricted to classical adversaries and simulators.

3.4 OEKE

3.4.1 Syntax

Once-encrypted key exchange (OEKE) is a template for building two-message PAKE specified by the following ingredients.

- A KEM $\text{KEM} = (\text{KEM.KeyGen}, \text{KEM.Encaps}, \text{KEM.Decaps})$.
- (Potentially oracle-aided) encryption and decryption algorithms $\mathcal{E}^{O_{\text{Enc}}}, \mathcal{D}^{O_{\text{Enc}}}$ with the following syntax:
 - $\phi := \mathcal{E}^{O_{\text{Enc}}}(\text{pw}, \text{pk}; r)$ takes a password pw , a KEM public key pk , and random coins r , and outputs a message ϕ .
 - $\text{pk} := \mathcal{D}^{O_{\text{Enc}}}(\text{pw}, \phi)$ takes a password pw and a message ϕ and outputs pk .
- A (potentially oracle-aided) key derivation function $\mathcal{K}^{O_{\text{Key}}}(K) = \text{key}$.
- A (potentially oracle-aided) tag function $\mathcal{T}^{O_{\text{Tag}}}(\text{pw}, \text{pk}, \phi, c, K) = \tau$.

Given the above ingredients, the OEKE protocol is specified as follows.

- $\text{PAKE}_1(\text{pw}) \rightarrow (\text{st}, \phi)$:
 - Sample $(\text{pk}, \text{sk}) \leftarrow \text{KEM.KeyGen}$.
 - Sample r and compute $\phi = \mathcal{E}^{O_{\text{Enc}}}(\text{pw}, \text{pk}; r)$.
 - Set $\text{st} = (\text{sk}, \text{pk}, \phi)$
- $\text{PAKE}_2(\text{pw}, \phi) \rightarrow (c, \tau, \text{key})$:
 - Compute $\text{pk} = \mathcal{D}^{O_{\text{Enc}}}(\text{pw}, \phi)$.
 - Sample $(c, K) \leftarrow \text{KEM.Encaps}(\text{pk})$.
 - Set $\text{key} = \mathcal{K}^{O_{\text{Key}}}(K)$.
 - Set $\tau = \mathcal{T}^{O_{\text{Tag}}}(\text{pw}, \text{pk}, \phi, c, K)$.
- $\text{PAKE}_3(\text{pw}, \text{st}, c, \tau) \rightarrow \text{key}$:
 - Compute $K = \text{KEM.Decaps}(\text{sk}, c)$.
 - Set $\text{key} = \mathcal{K}^{O_{\text{Key}}}(K)$.
 - If $\tau = \mathcal{T}^{O_{\text{Tag}}}(\text{pw}, \text{pk}, \phi, c, K)$, output key , otherwise output $\perp$.

Besides correctness and security of KEM, we also require public key uniformity (Definition 3.8) for post-quantum security; otherwise a quantum adversary can mount an offline password attack by obtaining $\phi = \mathcal{E}(\text{pw}, \text{pk})$, computing $\text{pk}_j = \mathcal{D}(\text{pw}_j, \phi)$ for many candidate passwords pw_j , and checking which pk_j follows the public key distribution.

3.4.2 Game-based security

Next, we formulate a game-based stand-alone notion of security for two-message PAKE.

Definition 3.15 (Two-message PAKE: Correctness). *A two-message PAKE scheme $(\text{PAKE}_1, \text{PAKE}_2, \text{PAKE}_3)$ satisfies correctness if for any pw ,*

$$\Pr \left[\begin{array}{l} (\text{st}, \phi) \leftarrow \text{PAKE}_1(\text{pw}) \\ (c, \tau, \text{key}_B) \leftarrow \text{PAKE}_2(\text{pw}, \phi) \\ \text{key}_A \leftarrow \text{PAKE}_3(\text{pw}, \text{st}, c, \tau) \end{array} \right] = 1 - \text{negl}(\lambda).$$

Definition 3.16 (Two-message PAKE: Passive security). *A two-message PAKE scheme $(\text{PAKE}_1, \text{PAKE}_2, \text{PAKE}_3)$ satisfies passive security if for any pw and QPT adversary $\mathcal{A}$,*

$$\Pr \left[\begin{array}{l} (\text{st}, \phi) \leftarrow \text{PAKE}_1(\text{pw}) \\ (c, \tau, \text{key}) \leftarrow \text{PAKE}_2(\text{pw}, \phi) \\ b \leftarrow \{0, 1\} \\ \text{If } b = 0, \text{ chall} := \text{key} \\ \text{If } b = 1, \text{ chall} \leftarrow \{0, 1\}^\lambda \end{array} \right] \leq \frac{1}{2} + \text{negl}(\lambda).$$

Definition 3.17 (Two-message PAKE: Security against malicious initiator). *A two-message PAKE scheme $(\text{PAKE}_1, \text{PAKE}_2, \text{PAKE}_3)$ satisfies security against malicious initiator if for any distribution D with min-entropy γ over passwords $\mathcal{PW}$ and any QPT adversary $(\mathcal{A}_1, \mathcal{A}_2)$,*

$$\Pr \left[\begin{array}{l} \text{pw} \leftarrow D \\ (\phi^*, \sigma) \leftarrow \mathcal{A}_1(1^\lambda) \\ (c, \tau, \text{key}) \leftarrow \text{PAKE}_2(\text{pw}, \phi^*) \\ b \leftarrow \{0, 1\} \\ \text{If } b = 0, \text{ chall} := \text{key} \\ \text{If } b = 1, \text{ chall} \leftarrow \{0, 1\}^\lambda \end{array} \right] \leq \frac{1}{2} + \frac{2^{-\gamma}}{2} + \text{negl}(\lambda).$$

Definition 3.18 (Two-message PAKE: Security against malicious respondent). *A two-message PAKE scheme $(\text{PAKE}_1, \text{PAKE}_2, \text{PAKE}_3)$ satisfies security against malicious respondent if for any distribution D with min-entropy γ over passwords $\mathcal{PW}$ and any QPT adversary $(\mathcal{A}_1, \mathcal{A}_2)$,*

$$\Pr \left[\begin{array}{l} \text{pw} \leftarrow D \\ (\text{st}, \phi) \leftarrow \text{PAKE}_1(\text{pw}) \\ (c^*, \tau^*, \sigma) \leftarrow \mathcal{A}_1(\phi) \\ \text{key} \leftarrow \text{PAKE}_3(\text{pw}, \text{st}, c^*, \tau^*) \\ b \leftarrow \{0, 1\} \\ \text{If } \text{key} = \perp, \text{ chall} := \perp \\ \text{If } b = 0, \text{ chall} := \text{key} \\ \text{If } b = 1, \text{ chall} \leftarrow \{0, 1\}^\lambda \end{array} \right] \leq \frac{1}{2} + \frac{2^{-\gamma}}{2} + \text{negl}(\lambda).$$

3.4.3 UC security

Universally Composable security was introduced by Canetti [Can01] as a version of simulation-based security that provides security when the protocol is concurrently composed with other protocols. UC is similar to standalone security (as in Definition 3.14) except the distinguisher is now *interactive* (the interactive distinguisher is referred to as the environment $\mathcal{Z}$). [Unr10] formally defined UC-security for quantum protocols and adversaries and proved that secure protocol composition still holds. We adopt the same framework, so the simulator, adversary and ideal functionalities can be quantum. However, we study post-quantum protocols, so all honest parties are classical and send/receive only classical messages, and the ideal functionality being UC-realized is classical.

- On input (NewSession, sid, $P_1, P_2, \text{pw}, \text{role}$) from P_1 , send (NewSession, sid, P_1, P_2, role) to $\mathcal{S}$. Furthermore, if this is the first NewSession message for sid, or this is the second NewSession message for sid and there is a record $\langle P_2, P_1, \cdot, \cdot \rangle$, then record $\langle P_1, P_2, \text{pw}, \text{role} \rangle$ and mark it *fresh*.
 - On (TestPwd, sid, P_1, pw^*) from $\mathcal{S}$, if there is a record $\langle P_1, P_2, \text{pw}, \cdot \rangle$ marked *fresh*, then do:
 - If $\text{pw}^* = \text{pw}$, then mark the record *compromised* and send “correct guess” to $\mathcal{S}$.
 - If $\text{pw}^* \neq \text{pw}$, then mark the record *interrupted* and send “wrong guess” to $\mathcal{S}$.
 - On (NewKey, sid, $P_1, K^* \in \{0, 1\}^\lambda$) from $\mathcal{S}$, if there is a record $\langle P_1, P_2, \text{pw}, \text{role} \rangle$, and this is the first NewKey message for sid and P_1 , then output (sid, K) to P_1 , where K is defined as follows:
 - If the record is *compromised*, then set $K := K^*$.
 - If the record is *fresh*, a key (sid, K') has been output to P_2 , at which time there was a record $\langle P_2, P_1, \text{pw}, \cdot \rangle$ marked *fresh*, and $K' \neq \perp$, then set $K := K'$.
 - Otherwise if $\text{role} = \text{“respondent”}$ sample $K \leftarrow \{0, 1\}^\lambda$; if $\text{role} = \text{“initiator”}$, set $K := \perp$.
- Finally, mark the record *completed*.

Figure 3: UC PAKE functionality $\mathcal{F}_{\text{PAKE-ea}}$

A word on quantum ideal functionalities. In order to meaningfully talk about the post-quantum security of protocols in the UC framework that use a quantum accessible random oracle/ideal cipher, one must define a suitable ideal functionality. To the best of our knowledge, a formal definition for either has not appeared in the literature ([LLH24] provides a UC proof in the QROM but makes no reference to an ideal functionality). A priori, it is not obvious that such ideal functionalities exist, since in the UC framework ideal functionalities must be *efficient*, and defining efficient procedures for simulating quantum access to ideal primitives with negligible error has proven quite challenging, with a long line of recent work [Zha19, Unr23, Car26, GY25, FLM⁺26].

By [Zha19, Lemma 4] Zhandry’s compressed standard oracle is an efficient and perfectly correct simulation of quantum access to a random oracle by the unitary $|x, y\rangle \mapsto |x, y \oplus H(x)\rangle$, so it can be used to define a corresponding UC functionality for the QROM¹⁰.

Ideal ciphers are more challenging: for example, there is no known quantum analogue of [DS16] enabling efficient simulation of an ideal cipher using a random oracle. It seems plausible that the techniques of [Car26] can be applied to an ideal cipher, but this would incur substantial simulation error (perhaps mitigated by very recent work [CM26]). Two other recent works [GY25, FLM⁺26] define a procedure to perfectly simulate a broad class of oracles that includes ideal ciphers,¹¹ but it is not yet known if this ideal cipher simulation is efficient¹². However, this is not an issue for us, as we aim to show *UC-insecurity*. In other words, we show that even if such an ideal functionality existed, (O)EKE would be UC-insecure relative to it.

The PAKE functionality. Our UC PAKE functionality $\mathcal{F}_{\text{PAKE-ea}}$ is the same that was used in [Xu26] to analyze OEKE, and is given in Figure 3: it is a minor modification of the standard PAKE functionality that incorporates one-sided explicit authentication. We now give a brief overview of how it works.

The functionality involves two parties, P_1 with password pw and P_2 with password pw' . Each party is also given a *role*: the party with *role* = “initiator” sends the first protocol message. The functionality captures explicit authentication for the “initiator” party. Each execution of the protocol involves a session for P_1 and a session for P_2 , both of which have a corresponding record marked with a state:

- When a session is established (by a NewSession command), it is marked *fresh*, and will remain *fresh*

¹⁰One could also define an ideal functionality where access is given via the phase query interface $|x, y\rangle \mapsto (-1)^{y \cdot H(x)} |x, y\rangle$ but this is not useful in the post-quantum setting, as we assume classical algorithms querying the random oracle instantly measure the result in the computational basis.

¹¹Technically, this is the class of oracles sampled from a compact Lie group according to the Haar measure.

¹²[GY25, FLM⁺26] both give simulations of random permutations whose efficiency reduces to efficiency of the Clebsch-Gordan transform for permutations, which is currently an open problem.

until the simulator (ideal adversary) attacks the session.

- The simulator may attack a **fresh** P session by sending a **TestPwd** command for P , on a *password guess* pw^* . This models an *online guessing attack* in the real world, where the simulator runs the algorithm of P' on a password guess pw^* and communicates with P . If $\text{pw}^* = \text{pw}$, this is a successful attack, and the session is marked **compromised**; otherwise the session is marked **interrupted**. Note that once a session becomes **compromised** or **interrupted**, it can never return to **fresh**; this in particular means that **TestPwd** can be run only once on any specific session.
- The only way a party receives a session key is by the simulator sending a **NewKey** command, after which the session will output a key (or possibly the $\perp$ symbol for the initiator) to the party depending on the states of this session and its counter session. Subsequently the session is marked **completed** (so that **TestPwd** cannot be sent after the session ends).

We now emphasize a few points about $\mathcal{F}_{\text{PAKE-ea}}$ that will be important later.

1. Notice that a simulator can only guess the password of a party once: after that the record is no longer **fresh**. We define pw^* to be the password guess for a party P in session sid if $\mathcal{S}$ sent the command $(\text{TestPwd}, \text{sid}, P, \text{pw}^*)$ when there was a record $\langle P, P', \cdot, \cdot \rangle$ marked **fresh**. If no such command has been sent, the password guess is $\perp$. Once **NewKey** has been sent to P the password guess is fixed, as $\mathcal{S}$ can no longer send **TestPwd** to P .
2. If $\mathcal{S}$ is to be a successful simulator, whenever a party P outputs a key in the real world, $\mathcal{S}$ must send **NewKey** to P in the ideal world, otherwise the real and ideal world are trivially distinguishable.
3. When $\mathcal{S}$ sends **NewKey** to the respondent P , unless the record for P has already been marked **compromised** (which requires $\mathcal{S}$ to have made a correct password guess), the key of P is chosen uniformly and independently of $\mathcal{S}$'s view, and will remain independent of $\mathcal{S}$'s view for the rest of the security experiment.

4 Attacks on simulation-based security

4.1 Masny-Rindal OT

Theorem 4.1. *Let $\Pi = (\text{Rec}_1, \text{Sen}, \text{Rec}_2)$ be the Masny-Rindal OT protocol (Section 3.3.1) instantiated with any KEM with negligible correctness error (Definition 3.7). Then Π is not post-quantum simulation-secure (Definition 3.14).*

Proof. First, we note that security against a malicious sender fails if the KEM does not have computationally uniform public keys (Definition 3.8). Indeed, given t_0, t_1 produced by the honest receiver on input bit b , the sender can recover $\text{pk}_0 = t_0 \oplus H(t_1), \text{pk}_1 = t_1 \oplus H(t_0)$. Since t_{1-b} was sampled uniformly at random by the honest receiver, pk_{1-b} is distributed uniformly at random. On the other hand, if KEM does not have computationally uniform keys, pk_b is *distinguishable* from uniformly random, which gives the sender a non-negligible advantage in distinguishing pk_b from pk_{1-b} and thus in predicting b .

Therefore, we assume that KEM has computationally uniform public keys, and to complete the proof we show that Π is not simulation-secure against a malicious receiver. In particular, we define a (mixed) state¹³ on registers D, A and an adversary distinguisher pair $\mathcal{A}, \mathcal{D}$ for which no simulator $\mathcal{S}$ can possibly exist.

The state will be a uniform mixture over two pure states $|\psi_0\rangle_{D,A}$ and $|\psi_1\rangle_{D,A}$ specified by a choice of branch $c \in \{0, 1\}$. In a slight abuse of notation, we consider the choice of the bit c to be stored in register D , so it is known to the distinguisher (but not necessarily to the adversary). These states are defined as

¹³It is straightforward to obtain a counterexample using a pure state $|\psi\rangle_{D,A}$ by *coherently* recording the choice of the branch $c \in \{0, 1\}$ in register D .

$$\begin{aligned}
|\psi_0\rangle &= \frac{1}{\sqrt{2|\mathcal{PK}|^2}} \sum_{b \in \{0,1\}, t_{1-b} \in \mathcal{PK}, \mathbf{pk}_b \in \mathcal{PK}} |\mathbf{pk}_b\rangle_{\mathbf{D}} |b\rangle_{\mathbf{B}} |t_{1-b}\rangle_{\mathbf{X}} |\mathbf{pk}_b\rangle_{\mathbf{K}}, \\
|\psi_1\rangle &= \frac{1}{\sqrt{2|\mathcal{PK}||\mathcal{R}|}} \sum_{b \in \{0,1\}, t_{1-b} \in \mathcal{PK}, r_b \in \mathcal{R}} |r_b\rangle_{\mathbf{D}} |b\rangle_{\mathbf{B}} |t_{1-b}\rangle_{\mathbf{X}} |\mathbf{pk}(r_b)\rangle_{\mathbf{K}},
\end{aligned}$$

where $\mathbf{A} = (\mathbf{B}, \mathbf{X}, \mathbf{K})$. Now, $\mathcal{A}$ takes as input register $\mathbf{A}$ and does the following.

1. Initialize a fresh register $\mathbf{Y}$ and query H on registers $\mathbf{X}, \mathbf{Y}$ to obtain the following state:

$$\begin{aligned}
\text{if } c = 0 : & \quad \frac{1}{\sqrt{2|\mathcal{PK}|^2}} \sum_{b \in \{0,1\}, t_{1-b} \in \mathcal{PK}, \mathbf{pk}_b \in \mathcal{PK}} |\mathbf{pk}_b\rangle_{\mathbf{D}} |b\rangle_{\mathbf{B}} |t_{1-b}\rangle_{\mathbf{X}} |\mathbf{pk}_b\rangle_{\mathbf{K}} |H(t_{1-b})\rangle_{\mathbf{Y}} \\
\text{if } c = 1 : & \quad \frac{1}{\sqrt{2|\mathcal{PK}||\mathcal{R}|}} \sum_{b \in \{0,1\}, t_{1-b} \in \mathcal{PK}, r_b \in \mathcal{R}} |r_b\rangle_{\mathbf{D}} |b\rangle_{\mathbf{B}} |t_{1-b}\rangle_{\mathbf{X}} |\mathbf{pk}(r_b)\rangle_{\mathbf{K}} |H(t_{1-b})\rangle_{\mathbf{Y}}.
\end{aligned}$$

2. Initialize fresh registers $\mathbf{T}_0, \mathbf{T}_1$ and apply the computation to registers $(\mathbf{B}, \mathbf{X}, \mathbf{K}, \mathbf{Y}, \mathbf{T}_0, \mathbf{T}_1)$ that produces the following state:

$$\begin{aligned}
\text{if } c = 0 : & \quad \frac{1}{\sqrt{2|\mathcal{PK}|^2}} \sum_{b \in \{0,1\}, t_{1-b} \in \mathcal{PK}, \mathbf{pk}_b \in \mathcal{PK}} |\mathbf{pk}_b\rangle_{\mathbf{D}} |b\rangle_{\mathbf{B}} |t_{1-b}\rangle_{\mathbf{X}} |\mathbf{pk}_b\rangle_{\mathbf{K}} |H(t_{1-b})\rangle_{\mathbf{Y}} |t_0\rangle_{\mathbf{T}_0} |t_1\rangle_{\mathbf{T}_1} \\
\text{if } c = 1 : & \quad \frac{1}{\sqrt{2|\mathcal{PK}||\mathcal{R}|}} \sum_{b \in \{0,1\}, t_{1-b} \in \mathcal{PK}, r_b \in \mathcal{R}} |r_b\rangle_{\mathbf{D}} |b\rangle_{\mathbf{B}} |t_{1-b}\rangle_{\mathbf{X}} |\mathbf{pk}(r_b)\rangle_{\mathbf{K}} |H(t_{1-b})\rangle_{\mathbf{Y}} |t_0\rangle_{\mathbf{T}_0} |t_1\rangle_{\mathbf{T}_1},
\end{aligned}$$

where $t_b = \mathbf{pk}_b \oplus H(t_{1-b})$ in branch $c = 0$ and $t_b = \mathbf{pk}(r_b) \oplus H(t_{1-b})$ in branch $c = 1$.

3. Query H on registers $\mathbf{X}, \mathbf{Y}$, discard $\mathbf{Y}$, and measure $\mathbf{T}_0, \mathbf{T}_1$ to obtain (t_0, t_1) . On branch $c = 0$, this yields the state

$$\frac{1}{\sqrt{2}} \sum_{b \in \{0,1\}} |t_b \oplus H(t_{1-b})\rangle_{\mathbf{D}} |b\rangle_{\mathbf{B}} |t_{1-b}\rangle_{\mathbf{X}} |t_b \oplus H(t_{1-b})\rangle_{\mathbf{K}}.$$

We leave the precise description of the resulting state on branch $c = 1$ implicit, as it will not be important for the argument.

4. Query H (classically) on t_0 and t_1 to obtain y_0 and y_1 .
5. Output (t_0, t_1) to the sender, receive (c_0, c_1) , and send $(\mathbf{B}, \mathbf{X}, \mathbf{K}, (t_0, t_1, y_0, y_1, c_0, c_1))$ to the distinguisher $\mathcal{D}$, which is a mix of quantum registers and classical information.

Finally, we describe the behavior of the distinguisher $\mathcal{D}$, who knows the value of c , and additionally receives (m_0, m_1) (from the honest sender in the real experiment, and from $\mathcal{F}_{\text{OT}}$ in the ideal experiment).

- If $c = 0$, define

$$|\psi_{\text{check}}\rangle := \frac{1}{\sqrt{2}} \sum_{b \in \{0,1\}} |t_b \oplus y_{1-b}\rangle_{\mathbf{D}} |b\rangle_{\mathbf{B}} |t_{1-b}\rangle_{\mathbf{X}} |t_b \oplus y_{1-b}\rangle_{\mathbf{K}}$$

and apply the measurement $\{|\psi_{\text{check}}\rangle\langle\psi_{\text{check}}|, I - |\psi_{\text{check}}\rangle\langle\psi_{\text{check}}|\}$ to registers $(\mathbf{D}, \mathbf{B}, \mathbf{X}, \mathbf{K})$. Output 1 if the first outcome and 0 if the second.

- If $c = 1$, measure $\mathbf{D}, \mathbf{B}$ to obtain (r_b, b) , compute $m = \text{KEM.Decaps}(\text{sk}(r_b), c_b)$, and output 1 if $m = m_b$ and 0 otherwise.

Now, suppose towards contradiction that there exists a simulator $\mathcal{S}$ for which

$$|\Pr[\mathcal{D}(\mathbf{D}, \langle \mathcal{A}(A), \text{Sen} \rangle) = 1] - \Pr[\mathcal{D}(\mathbf{D}, \langle \mathcal{S}(A), \mathcal{F}_{\text{OT}} \rangle) = 1]| = \text{negl}(\lambda).$$

Let

$$p_{\text{Real}} = \Pr[\mathcal{D}(\mathbf{D}, \langle \mathcal{A}(A), \text{Sen} \rangle) = 1], \quad p_{\text{Ideal}} = \Pr[\mathcal{D}(\mathbf{D}, \langle \mathcal{S}(A), \mathcal{F}_{\text{OT}} \rangle) = 1].$$

Under this assumption, we can establish the following two claims, which yield a contradiction. $\square$

Claim 4.2. $p_{\text{Real}} = 1 - \text{negl}(\lambda)$.

Proof. In branch $c = 0$, the distinguisher outputs 1 with probability 1 due to the definition of the state on registers (D, B, X, K) . In branch $c = 1$, the correctness of the KEM directly implies that $\Pr[m = m_b] = 1 - \text{negl}(\lambda)$, and so the distinguisher outputs 1 with probability $1 - \text{negl}(\lambda)$. $\square$

Claim 4.3. $p_{\text{Ideal}} \leq 3/4 + \text{negl}(\lambda)$.

Proof. Consider the following sequence of hybrids, where each hybrid outputs a bit.

- **Hyb₀:** This is the ideal game: $\mathcal{D}(D, \langle \mathcal{S}(A), \mathcal{F}_{\text{OT}} \rangle)$.
- **Hyb₁:** Change how we compute the output in branch $c = 1$ as follows. Let (b^*, m^*) be the input that $\mathcal{S}$ chooses to send to $\mathcal{F}_{\text{OT}}$, and let b be the bit measured by the distinguisher. Output 1 when $b = b^*$ and 0 otherwise.

We can argue $\Pr[\text{Hyb}_1 = 1] \geq \Pr[\text{Hyb}_0 = 1] - \text{negl}(\lambda)$ as follows. First, immediately outputting 1 when $b = b^*$ instead of computing and comparing m with m_b only potentially increases the probability of outputting 1 when $b = b^*$. Next, when $b \neq b^*$, we have that m_b is chosen by $\mathcal{F}_{\text{OT}}$ as a uniform string in $\{0, 1\}^\lambda$, independent of the rest of the experiment, and thus $\Pr[m = m_b] = \text{negl}(\lambda)$.

- **Hyb₂:** Replace the branch $c = 1$ state with the branch $c = 0$ state. That is, prepare the state

$$\frac{1}{\sqrt{2|\mathcal{PK}|^2}} \sum_{b \in \{0,1\}, t_{1-b} \in \mathcal{PK}, \text{pk}_b \in \mathcal{PK}} |\text{pk}_b\rangle_D |b\rangle_B |t_{1-b}\rangle_X |\text{pk}_b\rangle_K$$

regardless of whether $c = 0$ or $c = 1$.

$|\Pr[\text{Hyb}_1 = 1] - \Pr[\text{Hyb}_2 = 1]| = \text{negl}(\lambda)$ follows directly from the computational public key uniformity of the KEM. Indeed, notice that neither **Hyb₁** nor **Hyb₂** need to operate on the D register in branch $c = 1$ to compute its output bit. The distinguisher no longer uses D , and D is inaccessible to $\mathcal{S}$. Thus, this register can be held by an external challenger in the computational public key uniformity game.

- **Hyb₃:** Right before the distinguisher makes its branch-dependent measurement, post-select the registers (D, B, X, K) onto

$$|\psi_{\text{check}}\rangle = \frac{1}{\sqrt{2}} \sum_{b \in \{0,1\}} |t_b \oplus y_{1-b}\rangle_D |b\rangle_B |t_{1-b}\rangle_X |t_b \oplus y_{1-b}\rangle_K.$$

$|\Pr[\text{Hyb}_2 = 1] - \Pr[\text{Hyb}_3 = 1]| = \text{negl}(\lambda)$ follows from the assumption that $|p_{\text{Real}} - p_{\text{Ideal}}| = \text{negl}(\lambda)$, the fact that $p_{\text{Real}} = 1 - \text{negl}(\lambda)$, and Gentle Measurement applied to the measurement $\{|\psi_{\text{check}}\rangle\langle\psi_{\text{check}}|, I - |\psi_{\text{check}}\rangle\langle\psi_{\text{check}}|\}$ performed in the $c = 0$ branch.

Combining the above, we have that $\Pr[\text{Hyb}_3 = 1] \geq \Pr[\text{Hyb}_0 = 1] - \text{negl}(\lambda)$. However, it is easy to see that $\Pr[\text{Hyb}_3 = 1] = 3/4$. Indeed, with probability $1/2$, we are in branch $c = 1$, and in this branch, measuring register B of the pure state $|\psi_{\text{check}}\rangle$ yields a bit that is uniform and independent of b^* , meaning that $\Pr[b = b^*] = 1/2$. Thus, $p_{\text{Ideal}} \leq 3/4 + \text{negl}(\lambda)$. $\square$

4.2 OEKE with ideal cipher

We first consider OEKE (Section 3.4.1) where the encryption and decryption algorithms are instantiated with an ideal cipher. That is, let $(\text{IC}, \text{IC}^{-1})$ be an ideal cipher, and let $\mathcal{E}^{(\text{IC}, \text{IC}^{-1})}(\text{pw}, \text{pk}) = \text{IC}(\text{pw}, \text{pk})$ and $\mathcal{D}^{(\text{IC}, \text{IC}^{-1})}(\text{pw}, \phi) = \text{IC}^{-1}(\text{pw}, \phi)$.

Theorem 4.4. *Let Π be an OEKE protocol instantiated with (1) any KEM with negligible correctness error (Definition 3.7), (2) the IC-based encryption and decryption algorithms $\mathcal{E}, \mathcal{D}$ defined above, (3) any key derivation function $\mathcal{K}$, and (4) any tag function $\mathcal{T}$. Then Π does not post-quantum UC-realize $\mathcal{F}_{\text{PAKE-ea}}$ ¹⁴ (Figure 3).*

¹⁴By [RX23], UC-realizing the UC PAKE functionality does not imply correctness.

Proof. By the same argument made at the beginning of the proof of Theorem 4.1 (see also the discussion in Section 3.4.1), we can assume without loss of generality that KEM has post-quantum computationally uniform public keys (Definition 3.8), as otherwise Π would not satisfy security against a malicious respondent.

Now, suppose $\mathcal{S}$ is a simulator for the protocol in Figure 2 with respect to the dummy adversary. Fix $\text{pw}_0 \neq \text{pw}_1$ to be any two distinct passwords. We construct an environment $\mathcal{Z}$ that distinguishes between the real and ideal world.

Environment $\mathcal{Z}_0$

1. Sample $b \leftarrow \{0, 1\}$ and send $(\text{NewSession}, \text{sid}, P_2, P_1, \text{pw}_b, \text{respondent})$ to P_2 for some nonce sid .
2. Prepare the normalized state

$$\frac{1}{\sqrt{2|\mathcal{PK}|}} \sum_{u \in \{0, 1\}, \text{pk} \in \mathcal{PK}} |\text{pw}_u\rangle_A |\text{pw}_u\rangle_X |\text{pk}\rangle_B |\text{pk}\rangle_Y.$$

3. Query $\mathcal{E}$ on registers X, Y into a new register Z (through $\mathcal{A}$).
4. Measure Z to get ϕ and leftover state ρ_{ABXY}^0 .
5. Query $\mathcal{D}(\text{pw}_u, \phi) = \text{pk}_u$ for $u \in \{0, 1\}$.
6. Let $|\psi\rangle = \frac{1}{\sqrt{2}} \sum_{u \in \{0, 1\}} |\text{pw}_u\rangle_A |\text{pw}_u\rangle_X |\text{pk}_u\rangle_B |\text{pk}_u\rangle_Y$ and apply the measurement $M = \{\Pi_\psi, I - \Pi_\psi\}$ to ρ^0 (here Π_ψ is the projector onto $|\psi\rangle\langle\psi|$).
7. Output 1 if M accepts and otherwise output 0.

Environment $\mathcal{Z}_1$

1. Sample $b \leftarrow \{0, 1\}$ and send $(\text{NewSession}, \text{sid}, P_2, P_1, \text{pw}_b, \text{respondent})$ to P_2 for some nonce sid .
2. Prepare the normalized state $\frac{1}{\sqrt{2|\mathcal{R}|}} \sum_{u \in \{0, 1\}, r \in \mathcal{R}} |\text{pw}_u\rangle_A |\text{pw}_u\rangle_X |r\rangle_B |\text{pk}(r)\rangle_Y$ using KEM.KeyGen .
3. Query $\mathcal{E}$ on registers X, Y into a new register Z (through $\mathcal{A}$).
4. Measure Z to get ϕ and leftover state ρ_{ABXY}^1 .
5. Query $\mathcal{D}(\text{pw}_u, \phi) = \text{pk}_u$ for $u \in \{0, 1\}$.
6. Measure A, B to get pw_i, r_i ; if $\text{pw}_i \neq \text{pw}_b$ output 1.
7. Instruct $\mathcal{A}$ to send ϕ to P_2 .
 - In the real world, P_2 outputs their key key' and sends c', τ' to $\mathcal{A}$.
 - In the ideal world, $\mathcal{S}$ must send $(\text{NewKey}, \text{sid}, P_2, \cdot)$ to $\mathcal{F}_{\text{PAKE-ea}}$ and c', τ' to $\mathcal{A}$. Let pw_{guess} be $\mathcal{S}$'s password guess for (sid, P_2) .
8. Compute $\text{sk}_i = \text{sk}(r_i)$, $K = \text{KEM.Decaps}(\text{sk}_i, c')$, and $\text{key}^* = \mathcal{K}(K)$.
9. Let key' be the output of P_2 in either world. If $\text{key}^* = \text{key}'$ output 1, otherwise output 0.

Our environment $\mathcal{Z}$ simply chooses $b' \leftarrow \{0, 1\}$ and runs $\mathcal{Z}_{b'}$.

Real world. We show that $\mathcal{Z}$ outputs 1 in the real world with $1 - \text{negl}$ probability. After $\mathcal{Z}_0$ queries $\mathcal{E}$ in Step 3 the state is

$$\frac{1}{\sqrt{2|\mathcal{PK}|}} \sum_{u \in \{0,1\}, \text{pk} \in \mathcal{PK}} |\text{pw}_u\rangle_A |\text{pw}_u\rangle_X |\text{pk}\rangle_B |\text{pk}\rangle_Y |\mathcal{E}(\text{pw}_u, \text{pk})\rangle_Z.$$

After Step 5, since $\mathcal{E}$ is a keyed permutation $\rho^0 = |\psi\rangle\langle\psi|$, so $\mathcal{Z}_0$ must output 1 with probability 1.

After $\mathcal{Z}_1$ queries $\mathcal{E}$ in Step 3 the state is

$$\frac{1}{\sqrt{2|\mathcal{R}|}} \sum_{u \in \{0,1\}, r \in \mathcal{R}} |\text{pw}_u\rangle_A |\text{pw}_u\rangle_X |r\rangle_B |\text{pk}(r)\rangle_Y |\mathcal{E}(\text{pw}_u, \text{pk}(r))\rangle_Z.$$

If $\text{pw}_i = \text{pw}_b$ in Step 6 then P_2 will decrypt ϕ to obtain pk_i in Step 7. By correctness of KEM, P_2 must then output key^* as computed by $\mathcal{Z}_1$ in Step 8 with probability $1 - \text{negl}$. Therefore $\mathcal{Z}_1$ outputs 1 with probability $1 - \text{negl}$.

Ideal world. We now give a hybrid argument to show that in the ideal world, $\mathcal{Z}$ outputs 1 with probability $\leq 1 - u(\lambda)$ for some non-negligible $u(\lambda)$. Let Hyb_0 be the experiment where $\mathcal{Z}$ runs in the ideal world against some simulator $\mathcal{S}$ for the dummy adversary. In general let $\mathcal{Z}_j^i$ be $\mathcal{Z}_j$ running in experiment Hyb_i .

Hyb_1 : same as Hyb_0 , but at the end of Step 7 of $\mathcal{Z}_1$, output 1 if $\text{pwguess} = \text{pw}_i$; otherwise output 0.

Analysis of the change: We can argue $\Pr[\text{Hyb}_1 = 1] \geq \Pr[\text{Hyb}_0 = 1] - \text{negl}(\lambda)$ as follows. First, immediately outputting 1 when $\text{pwguess} = \text{pw}_i$ only potentially increases the probability of outputting 1 when $\text{pwguess} = \text{pw}_i$. Next, when $\text{pwguess} \neq \text{pw}_i$, note in Step 7 $\text{pw}_i = \text{pw}_b$ so at this point $\mathcal{S}$ has not compromised P_2 with a correct password guess. Recall that in this case $\mathcal{F}_{\text{PAKE-ea}}$ samples P_2 's key key' uniformly and independently from $\{0,1\}^\lambda$, and it remains independent of $\mathcal{S}$'s view for the rest of the experiment. In total, we have $\Pr[\text{Hyb}_1 = 1] \geq \Pr[\text{Hyb}_0 = 1] - 2^{-\lambda}$.

Hyb_2 : same as Hyb_1 , but change Step 2 of $\mathcal{Z}_1$ to be the same as $\mathcal{Z}_0$ and adjust the rest of $\mathcal{Z}_1$ accordingly.

Analysis of the change: In the previous hybrid, we made Steps 8 and 9 of $\mathcal{Z}_1$ unnecessary, which is the only place where r_i is used. Therefore we no longer need the randomness on register B and can use the same state as in $\mathcal{Z}_0$. We now have $\mathcal{Z}_1^2$ as follows:

Environment $\mathcal{Z}_1^2$

1. Sample $b \leftarrow \{0,1\}$ and send $(\text{NewSession}, \text{sid}, P_2, P_1, \text{pw}_b, \text{respondent})$ to P_2 for some nonce sid .
2. Prepare the normalized state

$$\frac{1}{\sqrt{2|\mathcal{PK}|}} \sum_{u \in \{0,1\}, \text{pk} \in \mathcal{PK}} |\text{pw}_u\rangle_A |\text{pw}_u\rangle_X |\text{pk}\rangle_B |\text{pk}\rangle_Y.$$

3. Query $\mathcal{E}$ on registers X, Y into a new register Z (through $\mathcal{A}$).
4. Measure Z to get ϕ and leftover state ρ_{ABXY}^1 .
5. Query $\mathcal{D}(\text{pw}_u, \phi) = \text{pk}_u$ for $u \in \{0,1\}$.
6. Measure A to get pw_i ; if $\text{pw}_i \neq \text{pw}_b$ output 1.
7. Instruct $\mathcal{A}$ to send ϕ to P_2 .

- In the real world, P_2 outputs their key key' and sends c', τ' to $\mathcal{A}$.
- In the ideal world, $\mathcal{S}$ must send $(\text{NewKey}, \text{sid}, P_2, \cdot)$ to $\mathcal{F}_{\text{PAKE-ea}}$ and c', τ' to $\mathcal{A}$. Let pwguess be $\mathcal{S}$'s password guess for (sid, P_2) .

8. If $\text{pwguess} = \text{pw}_i$ output 1, otherwise output 0.

The only functional change in this hybrid is that pk is now sampled uniformly, so we construct a quantum polynomial time reduction $\mathcal{B}$ that breaks computational public key uniformity given access to an $\mathcal{S}$ that distinguishes Hyb_1 and Hyb_2 . Given the public key state η_Y , $\mathcal{B}$ runs Hyb_2 with $\mathcal{S}$, playing the part of both $\mathcal{F}_{\text{PAKE-ea}}$ and the environment. $\mathcal{F}_{\text{PAKE-ea}}$ is simulated by running the functionality, and the environment is simulated by running $\mathcal{Z}^2$, with the following changes:

- If $b' = 0$ or $\text{pw}_i \neq \text{pw}_b$ then output a random bit.
- In Step 2 of $\mathcal{Z}_1$ instead prepare the state

$$\frac{1}{\sqrt{2}} \sum_{u \in \{0,1\}} |\text{pw}_u\rangle_A |\text{pw}_u\rangle_X \eta_Y.$$

Note that

- $\mathcal{B}$ must use its access to $\mathcal{F}_{\text{PAKE-ea}}$ in Step 8 of $\mathcal{Z}_1$ to check if $\text{pwguess} = \text{pw}_i$;
- Changing Step 2 of $\mathcal{Z}_1$ does not affect its execution since $\mathcal{Z}_1$ no longer uses the B register.

Let c be the choice bit in the uniformity experiment. When $c = 0$ the view of $\mathcal{S}$ in $\mathcal{B}$ is as in Hyb_2 , and when $c = 1$ the view of $\mathcal{S}$ in $\mathcal{B}$ is as in Hyb_1 . Note that the events $b' = 0$ and $\text{pw}_i = \text{pw}_b$ happen independently in $\text{Hyb}_1, \text{Hyb}_2$ and the execution of $\mathcal{B}$ (regardless of the value of c). Furthermore, when $b' = 0 \vee \text{pw}_i \neq \text{pw}_b$ then both $\mathcal{Z}^1$ and $\mathcal{Z}^2$ behave identically. Similarly, in this case $\mathcal{B}(c = 0)$ and $\mathcal{B}(c = 1)$ behave identically. Concluding,

$$\begin{aligned} & |\Pr[\mathcal{Z}^2 \text{ outputs } 1] - \Pr[\mathcal{Z}^1 \text{ outputs } 1]| \\ &= |\Pr[\mathcal{B}(c = 0) \text{ outputs } 1] - \Pr[\mathcal{B}(c = 1) \text{ outputs } 1]| \\ &= \text{Adv}(\text{PKU}_{\text{KEM}}^{\mathcal{B}}) \leq \text{negl}(\lambda). \end{aligned}$$

Hyb₃: Let W the additional state in the experiment besides ABXY : this includes $\mathcal{S}$'s workspace register, the state of $\mathcal{F}_{\text{PAKE-ea}}$ and other classical public values. Let δ^j be the state over ABXYW at the end of Step 5 of $\mathcal{Z}_j^2$. Hyb_3 is the same as Hyb_2 but δ^j is replaced with $\frac{(\Pi_\psi \otimes I_W) \delta^j (\Pi_\psi \otimes I_W)}{1-p}$ at the end of Step 5 of $\mathcal{Z}_j^2$ for $j \in \{0, 1\}$ (here $p = \Pr[M \text{ rejects in Step 6 of } \mathcal{Z}_0]$).

Analysis of the change: Since $\mathcal{Z}_0^2, \mathcal{Z}_1^2$ are the same through the end of Step 5, $\delta^0 = \delta^1 = \delta$. Since $\rho^j = \text{Tr}_W(\delta^j)$, $\rho^0 = \rho^1 = \rho$. If p is non-negligible then $\mathcal{Z}_0$ outputs 1 with probability $\leq 1 - p$ and we are done, so we may assume that p is negligible. By the Gentle Measurement Lemma,

$$|\Pr[\mathcal{Z}^3 \text{ outputs } 1] - \Pr[\mathcal{Z}^2 \text{ outputs } 1]| \leq O(\sqrt{p}).$$

We now show $\Pr[\mathcal{Z}^3 \text{ outputs } 1] \leq 7/8$. This completes the proof since by our hybrid argument

$$\Pr[\mathcal{Z}^0 \text{ outputs } 1] \leq \Pr[\mathcal{Z}^3 \text{ outputs } 1] + \text{negl} \leq 7/8 + \text{negl}.$$

Let γ be the state over ABXYW at the end of Step 5 of $\mathcal{Z}_j^3$ (by previous reasoning γ does not depend on j). By the change in Hyb_3 , there is a state σ_W such that

$$\gamma = |\psi\rangle\langle\psi|_{\text{ABXY}} \otimes \sigma_W. \quad (1)$$

We claim that the instant before $\mathcal{S}$ sends TestPw (if it does), b is uniform from its point of view. If TestPw is sent before the end of Step 5, this holds since $\mathcal{S}$'s view is independent of b at this point. If it is sent after, by Equation (1) the value of i is independent of $\mathcal{S}$'s state, and Step 7 reveals no information about i before TestPw is sent. In Step 7 we have $\text{pw}_b = \text{pw}_i$, so either way $\Pr[\text{pwguess} = \text{pw}_i] = 1/2$. Since i, b are sampled independently and b is uniform, $\mathcal{Z}^3$ reaches Step 7 of $\mathcal{Z}_1^3$ with probability $1/4$, so it must output 0 with probability $\geq 1/8$. $\square$

5 Provable game-based security

5.1 Masny-Rindal OT

Theorem 5.1. *Let $\Pi = (\text{Rec}_1, \text{Sen}, \text{Rec}_2)$ be the Masny-Rindal OT protocol (Section 3.3.1) instantiated with any KEM with negligible correctness error (Definition 3.7), (post-quantum) computationally uniform public keys (Definition 3.8), and (post-quantum) key unpredictability (Definition 3.9). Then Π satisfies correctness (Definition 3.11), (post-quantum) game-based security against malicious receiver (Definition 3.12), and (post-quantum) game-based security against malicious sender (Definition 3.13).*

Correctness follows immediately from correctness of KEM.

Proof of malicious receiver security. As an intermediate step, we first prove a weaker version where the malicious receiver has to guess both strings. A QPT adversary $(\mathcal{A}_1, \mathcal{A}_2)$ should have negligible success in the following experiment:¹⁵

- On 1^λ , $\mathcal{A}_1^H(1^\lambda)$ runs and outputs a message (t_0, t_1) and auxiliary state σ .
- For $j \in \{0, 1\}$ sample random coins $r_j \leftarrow \mathcal{R}$ and compute $\text{pk}_j = t_j \oplus H(t_{1-j}), (c_j, K_j) := \text{KEM.Encaps}(\text{pk}_j; r_j)$.
- Run $\mathcal{A}_2^H(1^\lambda, c_0, c_1, \sigma)$ to get (K_0^*, K_1^*) .
- Adversary succeeds if $K_j = K_j^*$ for $j \in \{0, 1\}$.

We now define a reduction $\mathcal{B}^{\mathcal{A}_1, \mathcal{A}_2}$ that uses measure-and-reprogram to break key unpredictability. $\mathcal{B}$ will reprogram H such that the public key $\text{pk}_{\pi(0)}$ equals the challenge public key pk^* . $\mathcal{B}$ then uses the challenge ciphertext c^* in place of $c_{\pi(0)}$ to complete the protocol, and uses $\mathcal{A}_2$'s guess for $K_{\pi(0)}^*$ as its own.

Formally, suppose $\mathcal{A}_1$ makes q_1 queries to H and let δ be the success probability of $(\mathcal{A}_1, \mathcal{A}_2)$ in the weaker malicious receiver security game described above. Let $C^H(t_0, t_1, H(t_0), H(t_1), \sigma)$ be the following quantum algorithm:

- Compute $\text{pk}_j := t_j \oplus H(t_{1-j})$ for $j \in \{0, 1\}$;
- Sample $r_0, r_1 \leftarrow \mathcal{R}$;
- Compute $(c_j, *) := \text{KEM.Encaps}(\text{pk}_j; r_j)$;
- Write $K_0^*, K_1^* \leftarrow \mathcal{A}_2^H(c_0, c_1, \sigma)$ to an output register.

Let Π_{win} be the projector onto states where $K_j^* = \text{KEM.Encaps}(\text{pk}_j; r_j)[1]$ for $j \in \{0, 1\}$. Let $V^H(t_0, t_1, H(t_0), H(t_1), \sigma)$ output 1 when the projector $(C^H)^\dagger \Pi_{\text{win}} C^H$ accepts.

$\mathcal{B}$ is given an unpredictability challenge pk^*, c^* and will act as the measure-and-reprogram simulator for $\mathcal{A}_1^H$, with predicate V^H . In the first stage $\mathcal{B}$ receives $\pi, t_{\pi(0)}$ and sets $\Theta_{\pi(0)} \leftarrow \mathcal{PK}$; in the second stage $\mathcal{B}$ receives $t_{\pi(1)}$ and sets $\Theta_{\pi(1)} = \text{pk}^* \oplus t_{\pi(0)}$. On output t_0, t_1, σ , $\mathcal{B}$ runs $C^H(\mathbf{t}^*, \Theta)$ except that it does not generate $c_{\pi(0)}$ itself but instead uses c^* . $\mathcal{B}$ then measures and outputs $K_{\pi(0)}^*$. To analyze the success of $\mathcal{B}$, first note that V^H accepts on $\mathcal{A}_1$'s output exactly when $(\mathcal{A}_1, \mathcal{A}_2)$ wins the weaker malicious receiver security game. Additionally, $\mathcal{B}$ sets $\text{pk}^* = \text{pk}_{\pi(0)} = t_{\pi(0)} \oplus H(t_{\pi(1)}), c^* = c_{\pi(0)}$, so $K_{\pi(0)}^*$ will equal the desired challenge key if $(\mathcal{A}_1, \mathcal{A}_2)$ succeeds. The only difference in $(\mathcal{A}_1, \mathcal{A}_2)$'s view between $\mathcal{B}$ and the usual measure-and-reprogram simulator is that $\Theta_{\pi(1)}$ is not uniformly random. However, this cannot affect $\mathcal{B}$'s success probability too much without breaking public key uniformity: a reduction $\mathcal{B}'$, given a uniformity challenge pk^* , encapsulates $(c^*, K^*) \leftarrow \text{KEM.Encaps}(\text{pk}^*)$, and runs $\mathcal{B}$ as above, outputting 0 if $K_{\pi(0)}^* = K^*$ (and 1 otherwise). All together, summing over all choices of $\mathbf{t}^*$ and applying Theorem 3.5 with $t = 2$ gives

$$\text{Adv}(\text{UNP}_{\text{KEM}, \lambda}^{\mathcal{B}}) \geq \frac{\delta}{(2q_1 + 1)^4} - \text{Adv}(\text{PKU}_{\text{KEM}, \lambda}^{\mathcal{B}'}),$$

¹⁵We assume that the sender aborts if $t_0 = t_1$, which only occurs with $\text{negl}(\lambda)$ probability in an honest execution.

completing the proof. We obtain full security against a malicious receiver by using Lemma 3.3 to reduce to the weaker version. Suppose $(\mathcal{A}_1, \mathcal{A}_2)$ is a QPT adversary for malicious receiver security of Masny-Rindal OT with success probability $1/2 + \epsilon$:

- On 1^λ , $\mathcal{A}_1^H(1^\lambda)$ runs and outputs a message (t_0, t_1) and auxiliary state σ .
- For $j \in \{0, 1\}$ sample random coins $r_j \leftarrow \mathcal{R}$ and compute $\text{pk}_j = t_j \oplus H(t_{1-j}), (c_j, K_j) := \text{KEM.Encaps}(\text{pk}_j; r_j)$. Sample $b \leftarrow \{0, 1\}$.
- Run $\mathcal{A}_2^H(1^\lambda, c_0, c_1, b, \sigma)$ to get K_b^* .
- Adversary succeeds if $K_b = K_b^*$.

Without loss of generality, on input $1^\lambda, c_0, c_1, b, \sigma$, $\mathcal{A}_2^H$ runs an oracle-aided unitary U_b^H (with $1^\lambda, c_0, c_1$ hardwired in) on σ (plus some ancilla qubits), writing K_b^* into an extra register $\mathbf{X}$. Let P_b^* project onto states with

$$\mathbf{X} = \text{KEM.Encaps}(\text{pk}_b; r_b)[1]$$

and define $P_b = (U_b^H)^\dagger P_b^* U_b^H$. We now define an adversary $(\mathcal{A}'_1, \mathcal{A}'_2)$ for the weaker security:

- $\mathcal{A}'_1$: Run $\mathcal{A}_1^H$ as normal, outputting t_0, t_1, σ .
- $\mathcal{A}'_2(c_0, c_1, \sigma)$:
 1. Apply U_0^H to σ , measure $\mathbf{X}$ to obtain K_0^* , and copy K_0^* to a fresh register.
 2. Apply $U_1^H(U_0^H)^\dagger$ and measure $\mathbf{X}$ to get K_1^* . Output (K_0^*, K_1^*) .

For any fixed choice of r_0, r_1 , applying Lemma 3.3 to the projectors P_i and (in expectation over) the state σ , we have $\text{Tr}(P_0\sigma) + \text{Tr}(P_1\sigma) = 1 + 2\epsilon$ by assumption and the unitarity of U_i^H , so $\text{Tr}(P_0P_1P_0\sigma) \geq \epsilon^2$. Since P_i projects $\mathbf{X}$ to a single value, the measurement of $\mathbf{X}$ does not disturb the state conditioned on success, and the probability that $K_j^* = K_j$ for $j \in \{0, 1\}$ is exactly $\text{Tr}(P_1P_0\sigma P_0P_1) = \text{Tr}(P_0P_1P_0\sigma) \geq \epsilon^2$. By averaging over the sampling of r_0, r_1 , $(\mathcal{A}_1, \mathcal{A}_2)$ must have negligible advantage in the malicious receiver security game. $\square$

Proof of malicious sender security. The malicious sender security game for Masny-Rindal OT against QPT adversary $\mathcal{A}$ is the following:

- On 1^λ , sample $b \leftarrow \{0, 1\}$, $t_{1-b} \leftarrow \mathcal{PK}$, and $(\text{pk}_b, \text{sk}_b) \leftarrow \text{KEM.KeyGen}(1^\lambda)$.
- Set $t_b := \text{pk}_b \oplus H(t_{1-b})$.
- Send $\text{msg}_R := (t_0, t_1)$ to $\mathcal{A}$ and keep $\text{sk} := \text{sk}_b$.
- $\mathcal{A}$ returns a guess b^* : $\mathcal{A}$ wins if $b^* = b$.

Suppose $\mathcal{A}$ is an adversary for malicious sender security with success probability $\frac{1}{2} + \epsilon, \epsilon \geq 0$. We give a reduction $\mathcal{B}^{\mathcal{A}}$ that breaks uniformity of public keys. Given a challenge pk^* with choice bit c , $\mathcal{B}$ plays the part of Rec_1 but uses pk^* instead of generating its own pk_b . If $b^* = b$ then $\mathcal{B}$ outputs 1; otherwise it outputs a random bit. Note that when $c = 0$ ($\text{pk}^* \leftarrow \mathcal{PK}$) the bit b is independent of t_0, t_1 so

$$\Pr[b^* = b | c = 0] = 1/2.$$

When $c = 1$ (pk^* honestly generated) $\mathcal{B}$ simulates the malicious sender security experiment to $\mathcal{A}$, so $\Pr[b^* = b | c = 1] = 1/2 + \epsilon$, which implies $\Pr[b^* = b] = 1/2 + \epsilon/2$. We then have

$$\begin{aligned} \Pr[\text{win}] &= \Pr[\text{win} | b^* = b] \left(\frac{1}{2} + \frac{\epsilon}{2} \right) + \Pr[\text{win} | b^* \neq b] \left(\frac{1}{2} - \frac{\epsilon}{2} \right) \\ &= \Pr[\text{win} | b^* = b] \left(\frac{1}{2} + \frac{\epsilon}{2} \right) + \frac{1}{2} \left(\frac{1}{2} - \frac{\epsilon}{2} \right). \end{aligned}$$

By Bayes Theorem,

$$\Pr[\text{win} | b^* = b] = \Pr[c = 1 | b^* = b] = \frac{\frac{1}{2}(\frac{1}{2} + \epsilon)}{(\frac{1}{2} + \frac{\epsilon}{2})}.$$

All together,

$$\Pr[\text{win}] = \frac{1}{2} \left(\frac{1}{2} + \epsilon \right) + \frac{1}{2} \left(\frac{1}{2} - \frac{\epsilon}{2} \right) = \frac{1}{2} + \frac{\epsilon}{4}.$$

□

5.2 OEKE with 2-Feistel

Theorem 5.2. *Let Π be any OEKE protocol instantiated with (1) any KEM with negligible correctness error, UNP security, and computationally uniform public keys (see Section 3.2) (2) the 2-Feistel-based encryption and decryption algorithms $\mathcal{E}_{2F}, \mathcal{D}_{2F}$ (Appendix A), (3) key derivation function $\mathcal{K}$ modeled as a salted random oracle H_{key} with λ -length output,¹⁶ and (4) tag function $\mathcal{T}$ modeled as a random oracle H_{tag} with λ -length output.*

Then Π satisfies Correctness (Definition 3.15), Passive security (Definition 3.16), Security against malicious initiator (Definition 3.17), and Security against malicious respondent (Definition 3.18).

Correctness is immediate. Passive security follows from the security of the KEM; although the adversary can decrypt ϕ using pw to get pk , it still only sees pk, c, τ, w , and has to distinguish between $H_{\text{key}}(w, K)$ and a uniformly random string. Key unpredictability of KEM plus standard one-way-to-hiding shows that the adversary has negligible advantage in this game.

Proof of security against a malicious initiator. We first establish a weaker form of security that we call the *two-output* game: the malicious initiator should not be able to guess two independently encapsulated keys. Formally, a QPT adversary $(\mathcal{A}_1, \mathcal{A}_2)$ should have negligible probability of success in the following experiment:

- Sample $\text{pw}_1 \leftarrow D$, followed by $\text{pw}_2 \leftarrow D$ conditioned on $\text{pw}_2 \neq \text{pw}_1$.
- $(\phi^*, \sigma) \leftarrow \mathcal{A}_1^{H_1, H_2}(1^\lambda)$ outputs ϕ^* and auxiliary state σ .
- Sample $r_1, r_2 \leftarrow \mathcal{R}$.
- Compute $\text{pk}_i = \mathcal{D}_{2F}(\text{pw}_i, \phi^*)$ and $(c_i, K_i) := \text{KEM.Encaps}(\text{pk}_i; r_i)$ for $i \in \{1, 2\}$.
- Run $(K_1^*, K_2^*) \leftarrow \mathcal{A}_2^{H_1, H_2}(\text{pw}_1, \text{pw}_2, c_1, c_2, \sigma)$; the adversary succeeds if $K_i^* = K_i$ for $i \in \{1, 2\}$.

To show this, we will require a variant of Theorem 3.5 that works for multiple random oracles. We model this as follows: suppose we have random oracles $H_1 : A_1 \rightarrow B_1, H_2 : A_2 \rightarrow B_2$ and $\mathcal{A}_1^{H_1, H_2}$ is a quantum oracle machine given access to the H_i through the unitaries $U_i : |x, y\rangle \mapsto |x, H_i(x) \oplus y\rangle$. This adversary can be emulated by an $\mathcal{A}_2$ making the same number of queries given access to the random oracle $H : (\{1\} \times A_1) \sqcup (\{2\} \times A_2) \rightarrow B_1 \times B_2$ by querying $|i, a_i\rangle$ and only using the relevant part of the output. We denote a query $H_i(x)$ as $H(i, x)$ or (i, x) . Let $H(i, x)$ denote the intended random oracle output b_i ; we also occasionally abuse notation and denote the full output $(b_1, b_2) \in B_1 \times B_2$ by $H(i, x)$ as well. We may write an oracle algorithm $\mathcal{A}^{H_1, H_2}$ as $\mathcal{A}^H$ under this identification.

Suppose that the two-output game has non-negligible success probability. By averaging, fix distinct passwords pw_1, pw_2 for which the conditional success probability, denoted δ , is non-negligible. Define an algorithm $\mathcal{A}'_1$ as follows:

- Run $(\phi^* = (s, T), \sigma) \leftarrow \mathcal{A}_1^H(1^\lambda)$;

¹⁶Salting the random oracle means that the respondent will sample a random salt $w \leftarrow \{0, 1\}^\lambda$, set $\text{key} = H_{\text{key}}(w, K)$ as the output of the key derivation function, and send w as part of its message, meaning that the key derivation function is effectively $H_{\text{key}}(w, \cdot)$.

- Output

$$A_1 = (2, \text{pw}_1, T), A_2 = (2, \text{pw}_2, T), \phi^* = \sigma, \\ B_1 = (1, \text{pw}_1, u_1 = s \oplus H(2, \text{pw}_1, T)), B_2 = (1, \text{pw}_2, u_2 = s \oplus H(2, \text{pw}_2, T)).$$

Suppose $\mathcal{A}'_1$ makes q_1 queries to H . Let $C[r_1, r_2]^H$ be a quantum algorithm that has hard-coded random coins r_1, r_2 , additionally takes inputs

$$A_1 = (2, \text{pw}_1, T), A_2 = (2, \text{pw}_2, T), \phi^* = (s, T), \sigma, \quad (2)$$

$$B_1 = (1, \text{pw}_1, u_1), B_2 = (1, \text{pw}_2, u_2), \Theta_{A_1}^{(2)}, \Theta_{A_2}^{(2)}, \Theta_{B_1}^{(1)}, \Theta_{B_2}^{(1)}, \quad (3)$$

and does the following:

- Write 1 to an ancilla **flag** unless $u_i = s \oplus \Theta_{A_i}^{(2)}, H(A_i) = \Theta_{A_i}^{(2)}, H(B_i) = \Theta_{B_i}^{(1)}$ for all $i \in \{1, 2\}$.
- Continue execution if **flag** = 0:
- Compute $\text{pk}_i = (\Theta_{B_i}^{(1)})^{-1} \odot T$.
- Compute $(c_i, K_i) := \text{KEM.Encaps}(\text{pk}_i; r_i)$ for $i \in \{1, 2\}$.
- Run $(K_1^*, K_2^*) \leftarrow \mathcal{A}_2^{H_1, H_2}(\text{pw}_1, \text{pw}_2, c_1, c_2, \sigma)$ to an output register.
- Write 1 to **flag** unless $K_j^* = \text{KEM.Encaps}(\text{pk}_j; r_j)[1]$ for $j \in \{1, 2\}$.

Let Π_{win} be the projector onto output states where the value of **flag** equals 0. Using a purified implementation of $C[r_1, r_2]^H$, let $V[r_1, r_2]^H$ output 1 when the projector $(C[r_1, r_2]^H)^\dagger \Pi_{\text{win}} C[r_1, r_2]^H$ accepts, and 0 otherwise. We will show that

$$\Pr_{r_1, r_2 \leftarrow \mathcal{R}} [V[r_1, r_2]^H(A_1, A_2, \phi^*, \sigma, B_1, B_2) = 1 : (A_1, A_2, \phi^*, \sigma, B_1, B_2) \leftarrow \mathcal{A}'_1] = \text{negl}(\lambda).$$

Fix some choice of A_1, A_2, B_1, B_2 . Recall that the measure-and-reprogram simulator for $\mathcal{A}'_1$ will measure the points A_1, A_2, B_1, B_2 to reprogram in some order $\prec$ based on a random permutation π .

Claim 5.3. *Let E_1 be the event that $B_1 \prec A_1, B_2 \prec A_2$ and E_2 be the event that V^H outputs 1. Then $\Pr[E_1 \wedge E_2] \leq 2^{-\ell}$.*

Proof. Suppose without loss of generality that $A_1 \prec A_2$. At the point when $H(A_2)$ is reprogrammed to $\Theta_{A_2}^{(2)}$, if E_1 occurs then the values of $u_1, u_2, \Theta_{A_1}^{(2)}$ are known to the simulator, since the B_1, B_2, A_1 queries have already been made. If E_2 occurs later on we must have $u_2 \oplus \Theta_{A_2}^{(2)} = u_1 \oplus \Theta_{A_1}^{(2)}$, since these values will equal s when s is defined later on. Since $\Theta_{A_2}^{(2)}$ is a uniformly random value in $\{0, 1\}^\ell$ the claim follows. $\square$

Claim 5.3 will allow us to program a public key pk_{i^*} . Formally, we give a reduction $\mathcal{B}$ to key unpredictability of KEM as follows. Given a challenge pk^*, c^*

- $\mathcal{B}$ will act as the measure-and-reprogram simulator for $\mathcal{A}'_1$.
- $\mathcal{B}$ reprograms queries to random outputs $\Theta_i = \Theta_i^{(1)} \parallel \Theta_i^{(2)}$ until it measures $(1, \text{pw}_{i^*}, u_{i^*})$ after previously measuring $(2, \text{pw}_{i^*}, T)$ for some i^*, T . In this case, it reprograms the H_1 part of the query to be $(\text{pk}^*)^{-1} \odot T$, and the H_2 part to be a random $\Theta_{i^*}^{(2)}$. Otherwise it aborts the measure-and-reprogram and just guesses a random key $K \leftarrow \{0, 1\}^\lambda$.
- On outputs $\mathbf{x} = (A_1, A_2, B_1, B_2)$, $\mathcal{B}$ samples $r_{3-i^*} \leftarrow \mathcal{R}$ and runs $C[\{r_{3-i^*}, r_{i^*}\}]^{H(\mathbf{x}^* \Theta)}$ except that it uses c^* in place of c_{i^*} (and hence does not actually need to know the value of r_{i^*} , which was sampled by its challenger). $\mathcal{B}$ then measures and outputs $K_{i^*}^*$.

Note that $V[\{r_{3-i^*}, r_{i^*}\}]^H$ outputs 1 exactly when $(\mathcal{A}'_1, \mathcal{A}_2)$ wins the security game (with probability δ). We now argue that $V[\{r_{3-i^*}, r_{i^*}\}]^H$ still outputs 1 in $\mathcal{B}$ with noticeable probability. Assuming that a suitable i^* always exists incurs a negligible loss; by Claim 5.3,

$$\Pr[V[\{r_{3-i^*}, r_{i^*}\}]^H \text{ outputs } 1 \wedge \exists \text{ such an } i^*] \geq \Pr[V[\{r_{3-i^*}, r_{i^*}\}]^H \text{ outputs } 1] - 2^{-\ell}.$$

The reprogramming of H_1 to $(\mathbf{pk}^*)^{-1} \odot T$ is indistinguishable to $(\mathcal{A}'_1, \mathcal{A}_2)$ from reprogramming to a uniform value: a reduction $\mathcal{B}'$ to public key uniformity can, on receiving a challenge $\mathbf{pk}^*$, compute an encapsulation $(c^*, K^*) \leftarrow \text{KEM.Encaps}(\mathbf{pk}^*)$, and run $\mathcal{B}$ using $\mathbf{pk}^*, c^*$, returning 0 if $K^* = K_{i^*}^*$ and 1 otherwise. If $\mathbf{pk}^*$ is random $\mathcal{B}'$ simulates the usual measure-and-reprogram simulation, and if $\mathbf{pk}^*$ is real $\mathcal{B}'$ simulates the reduction $\mathcal{B}$. Therefore, summing over all points A_1, A_2, B_1, B_2 of the form specified in 2 & 3 and applying Theorem 3.5 when $t = 4$ we have

$$\text{Adv}(\text{UNP}_{\text{KEM}, \lambda}^{\mathcal{B}}) \geq \frac{\delta}{(2q_1 + 1)^8} - \text{Adv}(\text{PKU}_{\text{KEM}, \lambda}^{\mathcal{B}'}) - 2^{-\ell},$$

completing the proof. In fact, the above argument can be extended to the case where $\mathcal{A}_2$ is additionally given $w_i, y_i = H_{\text{key}}(w_i, K_i), \tau_i = H_{\text{tag}}(\mathbf{pw}_i, \mathbf{pk}_i, \phi^*, c_i, K_i)$ for $i \in \{1, 2\}$, and $(\mathcal{A}_1, \mathcal{A}_2)$ have quantum access to $H_{\text{key}}, H_{\text{tag}}$. To see this, we apply standard one-way-to-hiding [Unr14] within the KEM reduction, replacing the embedded challenge's tag and derived-key value by uniform strings while computing the other response honestly.

Using this, we now show a stronger version of search security: any QPT adversary $(\mathcal{A}_1, \mathcal{A}_2)$ has at most $2^{-\gamma} + \text{negl}$ chance of success in the following experiment:

- $(\phi^*, \sigma) \leftarrow \mathcal{A}_1^{H_1, H_2, H_{\text{key}}, H_{\text{tag}}}(1^\lambda)$ outputs ϕ^* and auxiliary state σ .
- Sample $\mathbf{pw} \leftarrow D, r \leftarrow \mathcal{R}, w \leftarrow \{0, 1\}^\lambda$.
- Compute $\mathbf{pk} = \mathcal{D}_{2F}(\mathbf{pw}, \phi^*), (c, K) := \text{KEM.Encaps}(\mathbf{pk}; r), y = H_{\text{key}}(w, K)$ and $\tau = H_{\text{tag}}(\mathbf{pw}, \mathbf{pk}, \phi^*, c, K)$.
- Run $K^* \leftarrow \mathcal{A}_2^{H_1, H_2, H_{\text{key}}, H_{\text{tag}}}(\mathbf{pw}, c, \tau, w, y, \sigma)$; the adversary succeeds if $K^* = K$.

We use Lemma 3.4 to reduce to the weaker version shown previously. Sample and fix a choice of $r_{\mathbf{pw}} \leftarrow \mathcal{R}$ and $w_{\mathbf{pw}} \leftarrow \{0, 1\}^\lambda$ for each password $\mathbf{pw}$. We will define a family of unitaries $\{U_{\mathbf{pw}}\}_{\mathbf{pw}}$, where each $U_{\mathbf{pw}}$ has a tuple (c, τ, w, y) hard-coded, which are deterministically derived from $\mathbf{pw}, r_{\mathbf{pw}}, w_{\mathbf{pw}}$, the random oracles, and ϕ^* . $U_{\mathbf{pw}}$ applies a coherent implementation of $\mathcal{A}_2^{H_1, H_2, H_{\text{key}}, H_{\text{tag}}}$ to $\mathbf{pw}, c, \tau, w, y, \sigma$ (plus some ancilla qubits), writing K^* into an extra register X . Let $P_{\mathbf{pw}}^*$ project onto states with

$$X = \text{KEM.Encaps}(\mathbf{pk}_{\mathbf{pw}}; r_{\mathbf{pw}})[1],$$

where $\mathbf{pk}_{\mathbf{pw}} = \mathcal{D}_{2F}(\mathbf{pw}, \phi^*)$, and define

$$P_{\mathbf{pw}} = (U_{\mathbf{pw}}^H)^\dagger P_{\mathbf{pw}}^* U_{\mathbf{pw}}^H.$$

Given $(\mathcal{A}_1, \mathcal{A}_2)$ with success probability $2^{-\gamma} + \epsilon$ we now define an adversary $(\mathcal{A}'_1, \mathcal{A}'_2)$, where $\mathbf{pw}_1, \mathbf{pw}_2$ are sampled as in the two-output game:

- $\mathcal{A}'_1(1^\lambda)$: output $(\phi^*, \sigma) \leftarrow \mathcal{A}_1^{H_1, H_2, H_{\text{key}}, H_{\text{tag}}}(1^\lambda)$.
- $\mathcal{A}'_2(\mathbf{pw}_1, \mathbf{pw}_2, \sigma)$:
 1. Apply $U_{\mathbf{pw}_1}^H$ to σ , measure X to obtain K_1^* , and copy K_1^* to a fresh register.
 2. Apply $U_{\mathbf{pw}_2}^H (U_{\mathbf{pw}_1}^H)^\dagger$ and measure X to get K_2^* .
 3. Output (K_1^*, K_2^*) .

As in Theorem 5.1, since each $P_{\mathbf{pw}}^*$ projects X to a single value, the measurement of X does not disturb the state conditioned on success. By applying Lemma 3.4 to the projectors $\{P_{\mathbf{pw}}\}_{\mathbf{pw}}$ and the state σ , averaged over $H_1, H_2, H_{\text{key}}, H_{\text{tag}}, \phi^*, \sigma, \{r_{\mathbf{pw}}, w_{\mathbf{pw}}\}_{\mathbf{pw}}$, we get that the probability $(\mathcal{A}'_1, \mathcal{A}'_2)$ outputs $K_1^* = K_1, K_2^* = K_2$ is at least ϵ^3 . This implies the existence of an adversary $(\mathcal{A}'_1, \mathcal{A}'_2)$ with the same probability of success in

the two-output game above, where $\mathcal{A}'_2$ now additionally takes $(c_1, \tau_1, w_1, y_1), (c_2, \tau_2, w_2, y_2)$ as input and hard-codes them into the descriptions of U_{pw_1}, U_{pw_2} . But above we showed that this probability must be negligible, establishing that $\epsilon = \text{negl}$.

Finally we are ready to show full indistinguishability security against a malicious initiator. Here is where we use the assumption that H_{key} is salted: a standard re-programming argument allows us to move to a hybrid (with negl distinguishing advantage) where the oracle $H_{\text{key}}(w, *)$ is sampled *independently* of K and the adversary's state after it sends its first message. We can apply Corollary 6.3.1 with $k = K, H = H_{\text{key}}(w, *)$ and ρ being the transcript/residual state. Therefore $|\Pr[\mathcal{A}_2 \text{ outputs } 1 | b = 0] - \Pr[\mathcal{A}_2 \text{ outputs } 1 | b = 1]| \leq 2^{-\gamma} + \text{negl}$, and hence the probability of winning in the full game is at most $1/2 + (2^{-\gamma} + \text{negl})/2 = 1/2 + 2^{-\gamma}/2 + \text{negl}$. $\square$

Proof of security against a malicious respondent. As an intermediate step, we first prove a weaker version of security where the malicious receiver has to guess pw given ϕ . A QPT adversary $\mathcal{A}$ should have at most $2^{-\gamma} + \text{negl}$ probability of success in the following experiment, where $\text{pw} \leftarrow D, (*, \phi) \leftarrow \text{PAKE}_1(\text{pw}); \text{pw}^* \leftarrow \mathcal{A}^{H_1, H_2}(\phi)$; and $\mathcal{A}$ wins if $\text{pw} = \text{pw}^*$. Given $\mathcal{A}$ that succeeds with $2^{-\gamma} + \epsilon$, we give a reduction $\mathcal{B}^{\mathcal{A}}$ that breaks uniformity of public keys with advantage at least $\frac{\epsilon}{4}$. Given a challenge pk^* with choice bit c , $\mathcal{B}$ plays the part of PAKE_1 but uses pk^* instead of honestly generating its own pk . If $\text{pw}^* = \text{pw}$ then $\mathcal{B}$ outputs 1; otherwise it outputs a uniformly random bit.

Claim 5.4. *When $c = 0$ ($\text{pk}^* \leftarrow \mathcal{PK}$), (pw, ϕ) are distributed as independent samples $\text{pw} \leftarrow D, \phi \leftarrow \{0, 1\}^\ell \times \mathcal{PK}$.*

Proof. Let us consider fixed H_1, H_2, pw . Recall that $\mathcal{E}_{2F}(\text{pw}, *, *)$ is a bijective map $\mathcal{PK} \times \{0, 1\}^\ell \rightarrow \{0, 1\}^\ell \times \mathcal{PK}$, with left inverse $\mathcal{D}_{2F}(\text{pw}, *)$. Therefore if w, pk are chosen uniformly, then $\phi = \mathcal{E}_{2F}(\text{pw}, \text{pk}; w)$ will be uniform as well. Since pw is fixed here, when $\text{pw} \leftarrow D$ it must be independent of ϕ . Additionally, since H_1, H_2 are fixed, this continues to hold even against an adversary that sees the entire truth tables of H_1, H_2 (so certainly against an adversary with polynomially many quantum queries to H_1, H_2). $\square$

Claim 5.4 immediately implies $p_0 := \Pr[\text{pw}^* = \text{pw} | c = 0] \leq 2^{-\gamma}$. When $c = 1$, $\mathcal{B}$ simulates an honest initiator to $\mathcal{A}^{H_1, H_2}$, so $p_1 := \Pr[\text{pw}^* = \text{pw} | c = 1] = 2^{-\gamma} + \epsilon$; all together, $\Pr[\mathcal{B} \text{ guesses } c] = \frac{1}{2} + \frac{p_1 - p_0}{4} \geq \frac{1}{2} + \frac{\epsilon}{4}$. We complete the proof of security against a malicious respondent by using Theorem 3.6 to build a password-guessing adversary from a malicious respondent adversary. Suppose $(\mathcal{A}_1, \mathcal{A}_2)$ wins the experiment of Definition 3.18 with probability $\frac{1}{2} + \frac{2^{-\gamma} + \epsilon}{2}$ for some non-negligible ϵ . Let Acc denote the event that the honest initiator accepts $\mathcal{A}_1$'s response. Since $\Pr[\text{win}] \leq \frac{1}{2} + \frac{1}{2} \Pr[\text{Acc}]$, we have $\Pr[\text{Acc}] \geq 2^{-\gamma} + \epsilon$. We now construct a password-guessing adversary that succeeds with probability at least $2^{-\gamma} + \epsilon - \text{negl}(\lambda)$, contradicting the above bound.

Suppose $\mathcal{A}_1$ makes q_1 queries to H_{tag} . Consider the following experiment Hyb_0 , where we run the honest initiator against $\mathcal{A}_1$ and output all values:

- $\text{pw} \leftarrow D$
- $(\text{sk}, \text{pk}, \phi) \leftarrow \text{PAKE}_1(\text{pw});$
- $c, \tau \leftarrow \mathcal{A}_1^{H_1, H_2, H_{\text{tag}}}(\phi);$
- Compute $K = \text{KEM.Decaps}(\text{sk}, c), \tau' = H_{\text{tag}}(\text{pw}, \text{pk}, \phi, c, K).$
- Output $\text{pw}, \text{pk}, \phi, c, K, \tau, \tau'.$

Note that $\Pr[\tau' = \tau] \geq \Pr[\text{Acc}] \geq 2^{-\gamma} + \epsilon$.

Hyb_1 : In this hybrid, we use the extractable QROM simulator of Theorem 3.6 to simulate H_{tag} . Since there are no $\mathcal{S.E}$ queries, Hyb_1 and Hyb_0 are perfectly indistinguishable.

Hyb_2 : In this hybrid, we add a $\mathcal{S.E}$ query at the end to extract $\mathcal{A}_1$'s password guess.

- $\text{pw} \leftarrow D$
- $(\text{sk}, \text{pk}, \phi) \leftarrow \text{PAKE}_1(\text{pw});$
- $c, \tau \leftarrow \mathcal{A}_1^{H_1, H_2, \mathcal{S.E}}(\phi);$

- Compute

$$\begin{aligned} K &= \text{KEM.Decaps}(\text{sk}, c), \\ \tau' &= \mathcal{S}.RO(\text{pw}, \text{pk}, \phi, c, K), \\ (\text{pw}', *, *, *, *) &= \mathcal{S}.E(\tau'). \end{aligned}$$

- Output $\text{pw}, \text{pk}, \phi, c, K, \tau, \tau', \text{pw}'$.

Since $\mathcal{S}.E$ is applied at the end, the other values are distributed identically between Hyb_2 and Hyb_1 . By Theorem 3.6, $\Pr[\tau = \tau' \wedge \text{pw} \neq \text{pw}'] \leq O(q_1^3/2^\lambda)$. Therefore in Hyb_2 ,

$$\Pr[\tau = \tau' \wedge \text{pw} = \text{pw}'] \geq 2^{-\gamma} + \epsilon - O(q_1^3/2^\lambda).$$

Hyb_3 : In this hybrid, we move $\mathcal{S}.E$ to just after $\mathcal{A}_1$ outputs.

- $\text{pw} \leftarrow D$
- $(\text{sk}, \text{pk}, \phi) \leftarrow \text{PAKE}_1(\text{pw})$;
- $c, \tau \leftarrow \mathcal{A}_1^{H_1, H_2, \mathcal{S}.RO}(\phi)$;
- Compute $(\text{pw}', *, *, *, *) = \mathcal{S}.E(\tau)$.
- Compute $K = \text{KEM.Decaps}(\text{sk}, c), \tau' = \mathcal{S}.RO(\text{pw}, \text{pk}, \phi, c, K)$.
- Output $\text{pw}, \text{pk}, \phi, c, K, \tau, \tau', \text{pw}'$.

By Theorem 3.6, in Hyb_3

$$\Pr[\tau = \tau' \wedge \text{pw} = \text{pw}'] \geq 2^{-\gamma} + \epsilon - O(q_1^3/2^\lambda) - \frac{8\sqrt{2}}{2^{\lambda/2}}.$$

Therefore if ϵ is non-negligible we have constructed an adversary guessing the password with non-negligible advantage: the adversary runs $c, \tau \leftarrow \mathcal{A}_1^{H_1, H_2, \mathcal{S}.RO}(\phi)$ (simulating H_{tag} with $\mathcal{S}$), computes $(\text{pw}', *, *, *, *) = \mathcal{S}.E(\tau)$ and outputs pw' as the password guess. $\square$

6 Advantage-tight one-way to hiding

In this section, we establish a search-to-decision reduction in the quantum random oracle model where the search upper bound transfers to a distinguishing upper bound without advantage loss (allowing for polynomially greater running time).

In fact, we consider a very general scenario specified by any sampler D that outputs (H, G, S, ρ) , where H, G are functions from $\mathcal{X} \rightarrow \mathcal{Y}$, S is a subset of the domain $S \subseteq \mathcal{X}$, and ρ is a quantum state.¹⁷ We require that $H(x) = G(x) \forall x \notin S$. Let $M_S : \mathcal{X} \rightarrow \{0, 1\}$ be the membership-checking predicate for S . For any (oracle-aided) distinguishing circuit A , we define its distinguishing advantage as

$$d_A = \left| \Pr_{(H, G, S, \rho) \leftarrow D} [A^H(\rho) = 1] - \Pr_{(H, G, S, \rho) \leftarrow D} [A^G(\rho) = 1] \right|,$$

and for any (oracle-aided) search circuit B , we define its search advantage as

$$p_B = \Pr_{(H, G, S, \rho) \leftarrow D} [x \in S : x \leftarrow B^{H, G, M_S}(\rho)].$$

Notice that we allow the search algorithm to make oracle calls to the membership checking predicate M_S .

We show that the distinguishing advantage is *exactly* upper bounded by the search advantage (plus an additive negligible factor). We take all oracle operations to be unit cost, so below QPT means that the total size of all non-oracle operations is polynomial.

¹⁷Technically we consider a family of samplers $H_\lambda, G_\lambda, S_\lambda, \rho_\lambda \leftarrow D(1^\lambda)$ indexed by the security parameter, but this is left implicit.

Theorem 6.1. *Let $\epsilon : \mathbb{N} \rightarrow [0, 1]$ and suppose that for every QPT B , $p_B \leq \epsilon(\lambda) + \text{negl}(\lambda)$. Then for every QPT distinguisher A , $d_A \leq \epsilon(\lambda) + \text{negl}(\lambda)$.*

The starting point for our proof is a decomposition of an arbitrary attacker’s distinguishing advantage as the sum of two terms: a “forward” branch that sums its amplitudes on S at each oracle query, and a “reverse” branch that, starting from the adversary’s *final* state, sums the amplitudes on S at each oracle query when running the adversary in reverse.

This decomposition is the central idea underlying the “measure-rewind-measure” technique of [KSS⁺20]. However, their search bound loses a factor that depends on the number of queries q made by the distinguisher, since their search adversary essentially chooses one (forward or reverse) query to measure at random.

Here, we move this factor of q from the advantage bound to the *running time* of the search adversary by using quantum rewinding. In particular, we define a unitary U that runs the search adversary suggested by the above decomposition *coherently* over the choice of branch $b \in \{\text{forward}, \text{reverse}\}$ and query $t \in [q]$. We then define an “input” projector Π_{in} that projects onto the clean starting space of this (purified) adversary, and a “success” projector Π_{good} that projects onto the set S . Then, by alternating the projectors Π_{in} and $U^\dagger \Pi_{\text{good}} U$ sufficiently many times, one can amplify *almost all* of the adversary’s input state that is responsible for its distinguishing advantage (which, by the decomposition stated above, lies in the span of right singular vectors of $\Pi_{\text{good}} U \Pi_{\text{in}}$ above a certain threshold) onto the image of Π_{good} . Measuring the output register then produces a string in S .

The quantum rewinding step is captured by the below amplitude amplification lemma, which follows readily from [GSLW19, Theorem 26].

Lemma 6.2 ([GSLW19]). *Let U be a unitary, let Π_{in} and Π_{good} be projectors, and write*

$$\Pi_{\text{good}} U \Pi_{\text{in}} = \sum_j \sigma_j |u_j\rangle\langle v_j|, \quad P_{\geq \gamma} := \sum_{j: \sigma_j \geq \gamma} |v_j\rangle\langle v_j|.$$

For any $\gamma > 0$ and $\delta \in (0, 1/2)$, there exists a unitary $\text{Amp}_{\gamma, \delta}$ using a total of $O(\log(1/\delta)/\gamma)$ calls to $U, U^\dagger, \Pi_{\text{in}}, \Pi_{\text{good}}$ such that for any ρ supported on the image of Π_{in} ,

$$\text{Tr}(\Pi_{\text{good}} \text{Amp}_{\gamma, \delta}(\rho)) \geq (1 - \delta)^2 \text{Tr}(P_{\geq \gamma} \rho).$$

Given this lemma, we are ready to prove Theorem 6.1.

Proof. (of Theorem 6.1) Let A be any QPT distinguisher that makes q oracle calls and has advantage d_A .

Notation. Fix a sample H, G, S, ρ from D and consider a purified implementation of A that operates on a register Z that holds its input state ρ along with any auxiliary registers required for the purification of the strategy. We also define a clock register R_{clock} spanned by $|0\rangle, |1\rangle, \dots, |q\rangle$ and initialized to $|0\rangle$. Finally, let $|\psi\rangle_{Z, R_{\text{pure}}, R_{\text{clock}}}$ denote an arbitrary purification of ρ using purification register R_{pure} , tensored with $|0\rangle_{R_{\text{clock}}}$. Note that A never touches R_{pure} or R_{clock} .

Using the purified implementation of A , we write

$$\begin{aligned} A_H &= A_q O_H A_{q-1} \dots A_1 O_H A_0 \\ A_G &= A_q O_G A_{q-1} \dots A_1 O_G A_0, \end{aligned}$$

where O_H is an oracle query to H and O_G is an oracle query to G . For $t \in [q]$, define

$$\begin{aligned} F_t &= A_{t-1} O_G A_{t-2} \dots A_1 O_G A_0, \\ R_t &= A_q O_H A_{q-1} \dots A_{t+1} O_H A_t. \end{aligned}$$

Next, define W_q to be a unitary satisfying

$$W_q |0\rangle_{R_{\text{clock}}} = \frac{1}{\sqrt{q}} \sum_{t=1}^q |t\rangle_{R_{\text{clock}}},$$

define forward and reverse history-state unitaries

$$U_{\text{for}} := \left(|0\rangle\langle 0|_{\text{R}_{\text{clock}}} \otimes I + \sum_{t=1}^q |t\rangle\langle t|_{\text{R}_{\text{clock}}} \otimes F_t \right) (W_q \otimes I_{Z, \text{R}_{\text{pure}}}),$$

$$U_{\text{rev}} := \left(|0\rangle\langle 0|_{\text{R}_{\text{clock}}} \otimes I + \sum_{t=1}^q |t\rangle\langle t|_{\text{R}_{\text{clock}}} \otimes R_t^\dagger \right) (W_q \otimes I_{Z, \text{R}_{\text{pure}}}),$$

and define projectors

$$\Pi_{\text{in}} := |0\rangle\langle 0|_{\text{R}_{\text{clock}}} \otimes I_{Z, \text{R}_{\text{pure}}}, \quad \Pi_{\text{good}} := \sum_{t=1}^q |t\rangle\langle t|_{\text{R}_{\text{clock}}} \otimes \Pi_S \otimes I_{\text{R}_{\text{pure}}},$$

where Π_S is the projection $\sum_{x \in S} |x\rangle\langle x|$ applied to the query register of A . Note that for any state $|\psi\rangle$ in the image of Π_{in} ,

$$\|\Pi_{\text{good}} U_{\text{for}} |\psi\rangle\|^2 = \frac{1}{q} \sum_{t \in [q]} \|\Pi_S F_t |\psi\rangle\|^2 \quad (4)$$

and

$$\|\Pi_{\text{good}} U_{\text{rev}} |\psi\rangle\|^2 = \frac{1}{q} \sum_{t \in [q]} \|\Pi_S R_t^\dagger |\psi\rangle\|^2. \quad (5)$$

Finally, let O be the single-qubit sub-register of Z on which A writes its output bit, and define

$$\Pi_{\text{acc}} := |1\rangle\langle 1|_{\text{O}} \otimes I.$$

Approach. As explained above, we will now construct a unitary U_A based on the description of A such that, for some threshold r , any right singular vector of $\Pi_{\text{good}} U_A \Pi_{\text{in}}$ with singular value below $1/r$ contributes negligibly to the distinguishing advantage of A . Moreover, we want the complexity of implementing U_A to be comparable to A . If this can be arranged, then we can apply Lemma 6.2 to amplify the overwhelming majority of A 's distinguishing advantage into the subspace Π_{good} , which produces a successful search answer.

The idea for U_A will be derived from re-writing the expression for A 's distinguishing advantage as follows.

$$\begin{aligned} & \left| \Pr[A^H(|\psi\rangle) = 1] - \Pr[A^G(|\psi\rangle) = 1] \right| \\ &= \left| \langle \psi | \left(A_H^\dagger \Pi_{\text{acc}} A_H - A_G^\dagger \Pi_{\text{acc}} A_G \right) | \psi \rangle \right| \\ &= \left| \langle \psi | \left(A_H^\dagger \Pi_{\text{acc}} A_H - A_H^\dagger \Pi_{\text{acc}} A_G + A_H^\dagger \Pi_{\text{acc}} A_G - A_G^\dagger \Pi_{\text{acc}} A_G \right) | \psi \rangle \right| \\ &= \left| \langle \psi | \left(A_H^\dagger \Pi_{\text{acc}} \Delta + \Delta^\dagger \Pi_{\text{acc}} A_G \right) | \psi \rangle \right| \\ &\leq \|\Delta |\psi\rangle\| + \|\Delta^\dagger \Pi_{\text{acc}} A_G |\psi\rangle\|, \end{aligned}$$

where

$$\Delta := A_H - A_G = \sum_{t \in [q]} R_t (O_H - O_G) F_t.$$

We have that

$$\begin{aligned}
\|\Delta |\psi\rangle\| &\leq \sum_{t \in [q]} \|R_t(O_H - O_G)F_t |\psi\rangle\| \\
&\leq 2 \sum_{t \in [q]} \|\Pi_S F_t |\psi\rangle\| \\
&\leq 2\sqrt{q} \left(\sum_{t \in [q]} \|\Pi_S F_t |\psi\rangle\|^2 \right)^{1/2} \\
&= 2\sqrt{q} (q \|\Pi_{\text{good}} U_{\text{for}} |\psi\rangle\|^2)^{1/2} \\
&= 2q \|\Pi_{\text{good}} U_{\text{for}} |\psi\rangle\|,
\end{aligned}$$

where the last inequality is Cauchy-Schwarz and the first equality is Equation (4). Similarly,

$$\begin{aligned}
\|\Delta^\dagger \Pi_{\text{acc}} A_G |\psi\rangle\| &\leq \sum_{t \in [q]} \|F_t^\dagger (O_H - O_G)^\dagger R_t^\dagger \Pi_{\text{acc}} A_G |\psi\rangle\| \\
&\leq 2 \sum_{t \in [q]} \|\Pi_S R_t^\dagger \Pi_{\text{acc}} A_G |\psi\rangle\| \\
&\leq 2\sqrt{q} \left(\sum_{t \in [q]} \|\Pi_S R_t^\dagger \Pi_{\text{acc}} A_G |\psi\rangle\|^2 \right)^{1/2} \\
&= 2q \|\Pi_{\text{good}} U_{\text{rev}} \Pi_{\text{acc}} A_G |\psi\rangle\|,
\end{aligned}$$

using Equation (5) and the fact that A_G and Π_{acc} preserve Π_{in} .

To summarize, we can write

$$\begin{aligned}
&|\Pr[A^H(|\psi\rangle) = 1] - \Pr[A^G(|\psi\rangle) = 1]| \\
&\leq \|\Delta |\psi\rangle\| + \|\Delta^\dagger \Pi_{\text{acc}} A_G |\psi\rangle\| \\
&\leq 2q (\|\Pi_{\text{good}} U_{\text{for}} |\psi\rangle\| + \|\Pi_{\text{good}} U_{\text{rev}} \Pi_{\text{acc}} A_G |\psi\rangle\|).
\end{aligned} \tag{6}$$

In other words, up to the $2q$ scaling, one component of the distinguishing advantage comes from a “forward” branch $\Pi_{\text{good}} U_{\text{for}} |\psi\rangle$ and the other comes from a “reverse” branch $\Pi_{\text{good}} U_{\text{rev}} \Pi_{\text{acc}} A_G |\psi\rangle$. Similar observations have been used in [KSS⁺20, GLX24].

The unitary U_A . We use this decomposition to inform the specification of the unitary U_A . First, expand the register Z to include single-qubit registers B and F initialized to $|0\rangle$. B will be a “branch” register, allowing both the forward and reverse branches to be run coherently, and F will be a register that allows for a unitary implementation of Π_{acc} , that is,

$$U_{\text{acc}} := I_F \otimes (I - \Pi_{\text{acc}}) + X_F \otimes \Pi_{\text{acc}}.$$

Redefine $|\psi\rangle$ to include $|00\rangle_{BF}$, redefine Π_{in} as

$$\Pi_{\text{in}} := |00\rangle\langle 00|_{BF} \otimes |0\rangle\langle 0|_{\text{R}_{\text{clock}}} \otimes I,$$

and redefine $\Pi_{\text{good}} = \Pi_{\text{good}} \otimes I_{BF}$. Then, define

$$U_A := (|0\rangle\langle 0|_B \otimes I_F \otimes U_{\text{for}} + |1\rangle\langle 1|_B \otimes (I_F \otimes U_{\text{rev}}) U_{\text{acc}} (I_F \otimes A_G)) (H_B \otimes I),$$

where H is the Hadamard gate.

The singular value cutoff. Write a singular value decomposition

$$\Pi_{\text{good}} U_A \Pi_{\text{in}} = \sum_j \sigma_j |u_j\rangle\langle v_j|,$$

and, fixing an r , define

$$P := \sum_{j: \sigma_j \geq 1/(8r)} |v_j\rangle\langle v_j|.$$

We can now establish the central claim of the proof, showing that A 's distinguishing advantage comes fully from the P -image of its input $|\psi\rangle$, up to error q/r .

Claim 6.3.

$$|\Pr[A^H(|\psi\rangle) = 1] - \Pr[A^G(|\psi\rangle) = 1]| \leq \|P|\psi\rangle\|^2 + \frac{q}{r}.$$

Proof. Let

$$L := \Pi_{\text{good}} U_A \Pi_{\text{in}}$$

and define

$$E := |00\rangle\langle 00|_{\text{BF}} \otimes |0\rangle\langle 0|_{\text{Rclock}} \otimes \left(A_H^\dagger \Pi_{\text{acc}} A_H - A_G^\dagger \Pi_{\text{acc}} A_G \right).$$

We first compare E with L . Let $|\phi\rangle$ be an arbitrary state (not necessarily in the image of Π_{in}) and define $|\phi_{\text{in}}\rangle := \Pi_{\text{in}}|\phi\rangle$. Define

$$\begin{aligned} f(\phi) &:= \|\Pi_{\text{good}} U_{\text{for}} |\phi_{\text{in}}\rangle\|, \\ b(\phi) &:= \|\Pi_{\text{good}} U_{\text{rev}} \Pi_{\text{acc}} A_G |\phi_{\text{in}}\rangle\|. \end{aligned}$$

Then

$$\begin{aligned} \|E|\phi\rangle\| &= \left\| \left(A_H^\dagger \Pi_{\text{acc}} A_H - A_G^\dagger \Pi_{\text{acc}} A_G \right) |\phi_{\text{in}}\rangle \right\| \\ &\leq \|\Delta |\phi_{\text{in}}\rangle\| + \|\Delta^\dagger \Pi_{\text{acc}} A_G |\phi_{\text{in}}\rangle\| \\ &\leq 2q(f(\phi) + b(\phi)), \end{aligned}$$

using Equation (6) applied to the possibly unnormalized vector $|\phi_{\text{in}}\rangle$.

By the definition of U_A , we have

$$\begin{aligned} L|\phi\rangle &= \frac{1}{\sqrt{2}} \left(|00\rangle_{\text{BF}} \Pi_{\text{good}} U_{\text{for}} |\phi_{\text{in}}\rangle \right. \\ &\quad + |10\rangle_{\text{BF}} \Pi_{\text{good}} U_{\text{rev}} (I - \Pi_{\text{acc}}) A_G |\phi_{\text{in}}\rangle \\ &\quad \left. + |11\rangle_{\text{BF}} \Pi_{\text{good}} U_{\text{rev}} \Pi_{\text{acc}} A_G |\phi_{\text{in}}\rangle \right), \end{aligned}$$

so discarding the second summand we have

$$\|L|\phi\rangle\|^2 \geq \frac{1}{2} (f(\phi)^2 + b(\phi)^2).$$

Thus, for any input state $|\phi\rangle$,

$$\|E|\phi\rangle\| \leq 2q(f(\phi) + b(\phi)) \leq 2q\sqrt{2}\sqrt{f(\phi)^2 + b(\phi)^2} \leq 4q\|L|\phi\rangle\|.$$

This establishes that

$$\|E(I - P)\| \leq 4q\|L(I - P)\| \leq \frac{q}{2r},$$

where the second inequality follows by definition of P using threshold $\gamma = 1/(8r)$. Therefore,

$$\|E - PEP\| = \|(I - P)E + PE(I - P)\| \leq \|(I - P)E\| + \|E(I - P)\| \leq \frac{q}{r},$$

using the fact that E is Hermitian.

Finally, since $|\psi\rangle$ is supported on the image of Π_{in} ,

$$\begin{aligned} & |\Pr[A^H(|\psi\rangle) = 1] - \Pr[A^G(|\psi\rangle) = 1]| \\ &= |\langle\psi| E |\psi\rangle| \\ &\leq |\langle\psi| PEP |\psi\rangle| + \|E - PEP\| \\ &\leq \langle\psi| P |\psi\rangle + \frac{q}{r} \\ &= \|P |\psi\rangle\|^2 + \frac{q}{r}, \end{aligned}$$

where the second inequality uses the fact that, since E is the difference of two projectors, $-I \preceq E \preceq I$. $\square$

Amplifying and completing the proof. Apply Lemma 6.2 with $\Pi_{\text{good}}, U = U_A, \Pi_{\text{in}}, \gamma = 1/(8r)$, and $\delta = 2^{-\lambda}$. This yields an amplified unitary **Amp** such that

$$\|\Pi_{\text{good}} \mathbf{Amp} |\psi\rangle\|^2 \geq (1 - \text{negl}(\lambda)) \|P |\psi\rangle\|^2.$$

Define the search adversary B to run **Amp** on the expanded register Z and R_{clock} and then measure the query address to obtain a response x . B can be implemented using $O(\lambda r)$ calls to $U_A, U_A^\dagger, \Pi_{\text{good}}$, and Π_{in} , and thus with the same number of calls to A and Π_S .

Hence, taking the expectation over $(H, G, S, \rho) \leftarrow D$, and applying Claim 6.3, we have that

$$\begin{aligned} d_A &= \left| \mathbb{E}_{(H, G, S, \rho) \leftarrow D} [\Pr[A^H(|\psi\rangle) = 1] - \Pr[A^G(|\psi\rangle) = 1]] \right| \\ &\leq \mathbb{E}_{(H, G, S, \rho) \leftarrow D} \left| \Pr[A^H(|\psi\rangle) = 1] - \Pr[A^G(|\psi\rangle) = 1] \right| \\ &\leq \mathbb{E}_{(H, G, S, \rho) \leftarrow D} \|P |\psi\rangle\|^2 + \frac{q}{r} \\ &\leq \frac{p_B}{(1 - \text{negl}(\lambda))} + \frac{q}{r} \\ &\leq \epsilon(\lambda) + \frac{q}{r} + \text{negl}(\lambda). \end{aligned}$$

Finally, assume for contradiction that there exists a polynomial $s(\lambda)$ such that on infinitely many security parameters, $d_A > \epsilon(\lambda) + \frac{1}{s(\lambda)}$. Then setting $r = 2qs(\lambda) = \text{poly}(\lambda)$ we obtain a contradiction, completing the proof. $\square$

Now we use Theorem 6.1 to analyze the following scenario. Let $(k, \rho) \leftarrow D$ be an arbitrary sampler that outputs $k \in \mathcal{X}$ and a state ρ . Let $H : \mathcal{X} \rightarrow \mathcal{Y}$ be a uniformly random function with $|\mathcal{Y}| = \lambda^{\omega(1)}$, sampled independently of D . We have the following corollary.

Corollary 6.3.1. *Let $\epsilon : \mathbb{N} \rightarrow [0, 1]$ and suppose that for every QPT B ,*

$$\Pr_{(k, \rho) \leftarrow D, H} [B^H(\rho, H(k)) = k] \leq \epsilon(\lambda) + \text{negl}(\lambda).$$

Then for every QPT A ,

$$\left| \Pr_{(k, \rho) \leftarrow D, H} [A^H(\rho, H(k)) = 1] - \Pr_{(k, \rho) \leftarrow D, H, u \leftarrow \mathcal{Y}} [A^H(\rho, u) = 1] \right| \leq \epsilon(\lambda) + \text{negl}(\lambda).$$

Proof. First, given k and H , define

$$y := H(k), \quad S := H^{-1}(y), \quad G(x) := \begin{cases} a & \text{if } H(x) = y \\ H(x) & \text{otherwise} \end{cases},$$

where $a \leftarrow \mathcal{Y}$. Then since G and M_S are efficiently implementable given y, a , and oracle access to H , and H is collision-resistant whenever $|\mathcal{Y}| = \lambda^{\omega(1)}$, we have that

$$\Pr_{(k,\rho) \leftarrow D, H, G} [B^{H, G, M_S}(\rho, y) \in S \setminus \{k\}] = \text{negl}(\lambda),$$

where G is sampled as described above. Combining with the assumption in the corollary statement, we get

$$\Pr_{(k,\rho) \leftarrow D, H, G} [B^{H, G, M_S}(\rho, y) \in S] \leq \epsilon(\lambda) + \text{negl}(\lambda).$$

So by Theorem 6.1, we have that

$$\left| \Pr_{(k,\rho) \leftarrow D, H, G} [A^H(\rho, H(k)) = 1] - \Pr_{(k,\rho) \leftarrow D, H, G} [A^G(\rho, H(k)) = 1] \right| \leq \epsilon(\lambda) + \text{negl}(\lambda).$$

Now, define

$$\tilde{G}(x) := \begin{cases} a, & \text{if } x = k \\ H(x) & \text{otherwise} \end{cases},$$

and note that G and $\tilde{G}$ only differ on $S \setminus \{k\}$. Again due to collision-resistance of H and Theorem 6.1 with $\epsilon(\lambda) = 0$ (though here standard one-way to hiding would suffice), we have that

$$\left| \Pr_{(k,\rho) \leftarrow D, H, G} [A^G(\rho, H(k)) = 1] - \Pr_{(k,\rho) \leftarrow D, H, \tilde{G}} [A^{\tilde{G}}(\rho, H(k)) = 1] \right| \leq \text{negl}(\lambda).$$

Finally, note that the distribution of $k, \rho, H(k), \tilde{G}$ is *identical* to the distribution of k, ρ, u, H in the experiment in the statement of the corollary. This completes the proof. $\square$

7 Acknowledgments

JB would like to thank Saikrishna Badrinarayan, Daniel Masny, and Pratyay Mukherjee for a previous collaboration related to the post-quantum security of the Masny-Rindal OT protocol, leading to insights that guided the direction of this work. JJ would like to thank Lawrence Roy and Naman Kumar for helpful discussions.

AI usage. The main contributions of this work were developed without AI assistance. AI tools were used to find the central idea behind the proof of Theorem 6.1 (advantage-tight one-way to hiding) and to aid with proofreading, checking of technical details, and polishing.

References

- [ABJ25] Afonso Arriaga, Manuel Barbosa, and Stanislaw Jarecki. Noic: Pake from kem without ideal ciphers. *Cryptology ePrint Archive*, 2025.
- [ABJ26] Afonso Arriaga, Manuel Barbosa, and Stanislaw Jarecki. Tempo: An ml-kem to pake compiler resilient to timing attacks. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2026.
- [AHU19] Andris Ambainis, Mike Hamburg, and Dominique Unruh. Quantum security proofs using semi-classical oracles. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCs*, pages 269–295. Springer, Cham, August 2019.
- [ARU14] Andris Ambainis, Ansis Rosmanis, and Dominique Unruh. Quantum attacks on classical proof systems: The hardness of quantum rewinding. In *55th FOCS*, pages 474–483. IEEE Computer Society Press, October 2014.

- [BCM⁺18] Zvika Brakerski, Paul Christiano, Urmila Mahadev, Umesh Vazirani, and Thomas Vidick. A cryptographic test of quantumness and certifiable randomness from a single quantum device. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 320–331. IEEE, 2018.
- [BCP03] Emmanuel Bresson, Olivier Chevassut, and David Pointcheval. Security proofs for an efficient password-based key exchange. In Sushil Jajodia, Vijayalakshmi Atluri, and Trent Jaeger, editors, *ACM CCS 2003*, pages 241–250. ACM Press, October 2003.
- [BCP⁺23] Hugo Beguinet, Céline Chevalier, David Pointcheval, Thomas Ricosset, and Mélissa Rossi. GeT a CAKE: Generic transformations from key encapsulation mechanisms to password authenticated key exchanges. In Mehdi Tibouchi and Xiaofeng Wang, editors, *ACNS 2023, Part II*, volume 13906 of *LNCS*, pages 516–538. Springer, Cham, June 2023.
- [BDF⁺11] Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, and Mark Zhandry. Random oracles in a quantum world. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 41–69. Springer, Berlin, Heidelberg, December 2011.
- [BM92] Steven M. Bellovin and Michael Merritt. Encrypted key exchange: Password-based protocols secure against dictionary attacks. In *1992 IEEE Symposium on Security and Privacy*, pages 72–84. IEEE Computer Society Press, May 1992.
- [BPR00] Mihir Bellare, David Pointcheval, and Phillip Rogaway. Authenticated key exchange secure against dictionary attacks. In Bart Preneel, editor, *EUROCRYPT 2000*, volume 1807 of *LNCS*, pages 139–155. Springer, Berlin, Heidelberg, May 2000.
- [BR93] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In Dorothy E. Denning, Raymond Pyle, Ravi Ganesan, Ravi S. Sandhu, and Victoria Ashby, editors, *ACM CCS 93*, pages 62–73. ACM Press, November 1993.
- [Can01] Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *42nd FOCS*, pages 136–145. IEEE Computer Society Press, October 2001.
- [Car26] Joseph Carolan. Compressed permutation oracles. In *Proceedings of the 58th Annual ACM Symposium on Theory of Computing*, pages 150–161, 2026.
- [CGLS24] Alexandru Cojocaru, Juan A. Garay, Qipeng Liu, and Fang Song. Improved quantum lifting by coherent measure-and-reprogram. In Kai-Min Chung and Yu Sasaki, editors, *ASIACRYPT 2024, Part IX*, volume 15492 of *LNCS*, pages 158–184. Springer, Singapore, December 2024.
- [CHK⁺05] Ran Canetti, Shai Halevi, Jonathan Katz, Yehuda Lindell, and Philip D. MacKenzie. Universally composable password-based key exchange. In Ronald Cramer, editor, *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 404–421. Springer, Berlin, Heidelberg, May 2005.
- [CHL⁺25] Alexandru Cojocaru, Minki Hhan, Qipeng Liu, Takashi Yamakawa, and Aaram Yun. Quantum lifting for invertible permutations and ideal ciphers. In Yael Tauman Kalai and Seny F. Kamara, editors, *Advances in Cryptology – CRYPTO 2025, Part II*, volume 16001 of *Lecture Notes in Computer Science*, pages 481–512. Springer, 2025.
- [CM26] Joseph Carolan and Christian Majenz. Compressed permutation oracles revisited, 2026.
- [Cry19] Crypto Forum Research Group. CFRG PAKE selection. <https://github.com/cfrg/pake-selection>, 2019. Accessed: 2026-09-30.
- [DFM20] Jelle Don, Serge Fehr, and Christian Majenz. The measure-and-reprogram technique 2.0: Multi-round Fiat-Shamir and more. In Daniele Micciancio and Thomas Ristenpart, editors, *CRYPTO 2020, Part III*, volume 12172 of *LNCS*, pages 602–631. Springer, Cham, August 2020.

- [DFMS19] Jelle Don, Serge Fehr, Christian Majenz, and Christian Schaffner. Security of the Fiat-Shamir transformation in the quantum random-oracle model. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 356–383. Springer, Cham, August 2019.
- [DFMS22] Jelle Don, Serge Fehr, Christian Majenz, and Christian Schaffner. Online-extractability in the quantum random-oracle model. In Orr Dunkelman and Stefan Dziembowski, editors, *EUROCRYPT 2022, Part III*, volume 13277 of *LNCS*, pages 677–706. Springer, Cham, May / June 2022.
- [DS16] Yuanxi Dai and John P. Steinberger. Indifferentiability of 8-round Feistel networks. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part I*, volume 9814 of *LNCS*, pages 95–120. Springer, Berlin, Heidelberg, August 2016.
- [DS23] Marcel Dall’Agnol and Nicholas Spooner. On the necessity of collapsing for post-quantum and quantum commitments. In *18th Conference on the Theory of Quantum Computation, Communication and Cryptography*, 2023.
- [Eur24] European Telecommunications Standards Institute. Impact of quantum computing on cryptographic security proofs. ETSI Technical Report TR 103 965 V1.1.1, ETSI, December 2024.
- [FLM⁺26] Ben Foxman, Alex Lombardi, Fermi Ma, Barak Nehoran, and John Wright. Quantum lazy sampling and path recording for any group. *arXiv preprint arXiv:2606.30281*, 2026.
- [FO99] Eiichi Fujisaki and Tatsuaki Okamoto. Secure integration of asymmetric and symmetric encryption schemes. In Michael J. Wiener, editor, *CRYPTO’99*, volume 1666 of *LNCS*, pages 537–554. Springer, Berlin, Heidelberg, August 1999.
- [GLX24] Jiangxia Ge, Heming Liao, and Rui Xue. Measure-rewind-extract: Tighter proofs of one-way to hiding and CCA security in the quantum random oracle model. In Kai-Min Chung and Yu Sasaki, editors, *ASIACRYPT 2024, Part IV*, volume 15487 of *LNCS*, pages 3–34. Springer, Singapore, December 2024.
- [GSLW19] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019, page 193–204, New York, NY, USA, 2019. Association for Computing Machinery.
- [GY25] Dmitry Grinko and Satoshi Yoshida. Quantum simulation of random unitaries from clebsch-gordan transforms. *arXiv preprint arXiv:2509.26623*, 2025.
- [HHKR25] Kathrin Hövelmanns, Andreas Hülsing, Mikhail Kudinov, and Silvia Ritsch. Cake requires programming-on the provable post-quantum security of (o) cake. *Cryptology ePrint Archive*, 2025.
- [JRX25] Jake Januzelli, Lawrence Roy, and Jiayu Xu. Under what conditions is encrypted key exchange actually secure? In Serge Fehr and Pierre-Alain Fouque, editors, *EUROCRYPT 2025, Part II*, volume 15602 of *LNCS*, pages 451–481. Springer, Cham, May 2025.
- [KSS⁺20] Veronika Kuchta, Amin Sakzad, Damien Stehlé, Ron Steinfeld, and Shifeng Sun. Measure-rewind-measure: Tighter quantum random oracle model proofs for one-way to hiding and CCA security. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part III*, volume 12107 of *LNCS*, pages 703–728. Springer, Cham, May 2020.
- [LLH24] You Lyu, Shengli Liu, and Shuai Han. Universal composable password authenticated key exchange for the post-quantum world. In Marc Joye and Gregor Leander, editors, *EUROCRYPT 2024, Part VII*, volume 14657 of *LNCS*, pages 120–150. Springer, Cham, May 2024.

- [LMQW22] Alex Lombardi, Ethan Mook, Willy Quach, and Daniel Wichs. Post-quantum insecurity from LWE. In Eike Kiltz and Vinod Vaikuntanathan, editors, *Theory of Cryptography – TCC 2022, Part I*, volume 13747 of *Lecture Notes in Computer Science*, pages 3–32. Springer, 2022.
- [MR19] Daniel Mansy and Peter Rindal. Endemic oblivious transfer. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, CCS ’19*, page 309–326, New York, NY, USA, 2019. Association for Computing Machinery.
- [MRR20] Ian McQuoid, Mike Rosulek, and Lawrence Roy. Minimal symmetric PAKE and 1-out-of-N OT from programmable-once public functions. In Jay Ligatti, Xinming Ou, Jonathan Katz, and Giovanni Vigna, editors, *ACM CCS 2020*, pages 425–442. ACM Press, November 2020.
- [PS96] David Pointcheval and Jacques Stern. Provably secure blind signature schemes. In Kwangjo Kim and Tsutomu Matsumoto, editors, *ASIACRYPT’96*, volume 1163 of *LNCS*, pages 252–265. Springer, Berlin, Heidelberg, November 1996.
- [RX23] Lawrence Roy and Jiayu Xu. A universally composable PAKE with zero communication cost - (and why it shouldn’t be considered UC-secure). In Alexandra Boldyreva and Vladimir Kolesnikov, editors, *PKC 2023, Part I*, volume 13940 of *LNCS*, pages 714–743. Springer, Cham, May 2023.
- [Unr10] Dominique Unruh. Universally composable quantum multi-party computation. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 486–505. Springer, Berlin, Heidelberg, May / June 2010.
- [Unr14] Dominique Unruh. Revocable quantum timed-release encryption. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 129–146. Springer, Berlin, Heidelberg, May 2014.
- [Unr23] Dominique Unruh. Towards compressed permutation oracles. In Jian Guo and Ron Steinfeld, editors, *ASIACRYPT 2023, Part IV*, volume 14441 of *LNCS*, pages 369–400. Springer, Singapore, December 2023.
- [Ver22] Sebastian Reynaldo Verschoor. Quantum information in security protocols. 2022.
- [VJW⁺25] Jelle Vos, Stanislaw Jarecki, Christopher A Wood, Cathie Yun, Steve Myers, and Yannick Sierra. A hybrid asymmetric password-authenticated key exchange in the random oracle model. *Cryptology ePrint Archive*, 2025.
- [VJW26] Jelle Vos, Stanislaw Jarecki, and Christopher A. Wood. Hybrid post-quantum password authenticated key exchange. <https://datatracker.ietf.org/doc/draft-vos-cfrg-pqpake/>, 2026. Internet-Draft, IETF.
- [Xu26] Jiayu Xu. The most efficient protocol for PAKE: What exact stuff do you need to hash at the end? *Cryptology ePrint Archive*, Paper 2026/1331, 2026.
- [YZ21] Takashi Yamakawa and Mark Zhandry. Classical vs quantum random oracles. In Anne Canteaut and François-Xavier Standaert, editors, *EUROCRYPT 2021, Part II*, volume 12697 of *LNCS*, pages 568–597. Springer, Cham, October 2021.
- [YZ22] Takashi Yamakawa and Mark Zhandry. Verifiable quantum advantage without structure. In *63rd FOCS*, pages 69–74. IEEE Computer Society Press, October / November 2022.
- [Zha19] Mark Zhandry. How to record quantum queries, and applications to quantum indistinguishability. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 239–268. Springer, Cham, August 2019.

A Generalizing the attack

For the sake of clarity and concreteness, Theorem 4.4 shows an attack against a specific instantiation of ideal cipher-based OEKE. However, our attack is quite general, and applies *mutatis mutandis* to all known variants of EKE and OEKE.

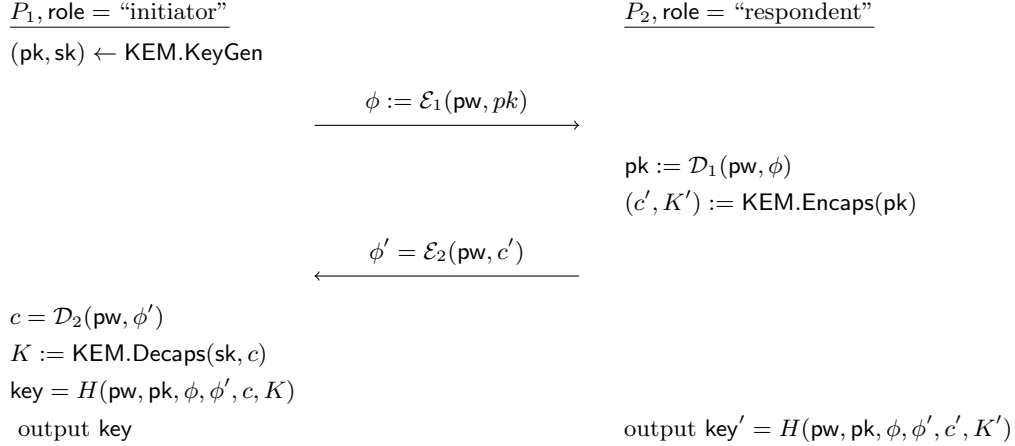


Figure 4: A typical instantiation of EKE using a KEM KEM , public key ideal cipher $\mathcal{E}_1, \mathcal{D}_1$ and ciphertext ideal cipher $\mathcal{E}_2, \mathcal{D}_2$.

EKE. Encrypted Key Exchange (shown in Figure 4) uses the same first message as OEKE, but instead of sending an authentication tag in the second message, the KEM ciphertext is also encrypted under the password using an ideal cipher. Since our attack only requires the environment to send a first message, it also works against EKE: the only difference is that $\mathcal{Z}_1$ must now query $\mathcal{D}_2(pw_i, \phi')$ to obtain the KEM ciphertext for key derivation.

2-Feistel/Tempo. Modern variants of OEKE do not use a full ideal cipher: the most popular alternative is a 2-Feistel network (also known as a Programmable Once Public Function) [MRR20, JRX25, ABJ25]. Let $H_1 : \mathcal{PW} \times \{0, 1\}^\ell \rightarrow \mathcal{PK}$, $H_2 : \mathcal{PW} \times \mathcal{PK} \rightarrow \{0, 1\}^\ell$ be random oracles, where $\mathcal{PK}$ is the space of public keys and an abelian group with operation $\odot$, and $\mathcal{PW}$ is the space of passwords ($\ell = \omega(\log \lambda)$; one can take $\ell = \lambda$ for simplicity). We define the 2-Feistel based encryption and decryption algorithms as follows.

- $\mathcal{E}_{2F}(pw, pk; w) = (s, T)$, where $R = H_1(pw, w)$, $T = pk \odot R$, $m = H_2(pw, T)$, $s = m \oplus w$.
- $\mathcal{D}_{2F}(pw, (s, T)) = pk$, where $m = H_2(pw, T)$, $w = s \oplus m$, $pk = H_1(pw, w)^{-1} \odot T$.

One can easily check that $\mathcal{D}_{2F}(pw, \mathcal{E}_{2F}(pw, pk; w)) = pk$ (w is recovered by the computation of $\mathcal{D}_{2F}$ as well).

The 2-Feistel is a more efficient alternative to an ideal cipher, since in general one needs eight rounds of a Feistel network to be indistinguishable from an IC [DS16]. Conversely, the 2-Feistel is a weaker primitive than an IC:

1. Encryption now uses additional randomness so there are many possible encryptions of pk under a given pw ;
2. 2F ciphertexts are partially malleable: given a ciphertext (s, T) that is an encryption of pk under pw , an adversary can produce (s', T') that is an encryption of $pk \odot \Delta$ under pw for adversarially chosen Δ .

These differences make the security proof more complex but still possible (see [JRX25, ABJ25] for details). The Tempo protocol [ABJ26] instantiates the above 2-Feistel OEKE template to use ML-KEM. The only difference is it splits public keys into two pieces $t \in \mathcal{PK}_1, \zeta \in \{0, 1\}^{\ell'}$ via an efficient and invertible procedure

Split, where ζ is uniformly random when $\mathbf{pk}$ is honestly generated. The protocol initiator 2-Feistel encrypts t only, sending ζ in the clear; the respondent decrypts and applies $\mathbf{Split}^{-1}$ to get a public key. Additionally, the security argument relies on a **KeyGen2** function that generates $(\mathbf{sk}, \mathbf{pk})$ from ζ : when ζ is chosen uniformly $\mathbf{KeyGen2}(\zeta)$ is guaranteed to be distributed identically to an honest key pair.

Theorem A.1. *Let Π be an OEKE protocol instantiated with (1) any KEM with negligible correctness error (Definition 3.7), (2) the 2-Feistel-based encryption and decryption algorithms $\mathcal{E}_{2F}, \mathcal{D}_{2F}$ defined above, or the Tempo algorithms, (3) any key derivation function $\mathcal{K}$, and (4) any tag function $\mathcal{T}$. Then Π does not post-quantum UC-realize $\mathcal{F}_{\text{PAKE-ea}}$.*

Proof. Consider the following environment interacting with 2-Feistel OEKE;

Environment $\mathcal{Z}_0$

1. Sample $b \leftarrow \{0, 1\}$ and send $(\text{NewSession}, \text{sid}, P_2, P_1, \mathbf{pw}_b)$ to P_2 for some nonce sid .
2. Prepare the normalized state

$$\frac{1}{\sqrt{2|\mathcal{PK}|2^\ell}} \sum_{\substack{u \in \{0,1\}, \mathbf{pk} \in \mathcal{PK}, \\ w \in \{0,1\}^\ell}} |w\rangle_C |\mathbf{pw}_u\rangle_A |\mathbf{pw}_u\rangle_X |pk\rangle_B |pk\rangle_Y.$$

3. Query H_1 on registers X, C into a new register Z_1 (through $\mathcal{A}$) to compute

$$\frac{1}{\sqrt{2|\mathcal{PK}|2^\ell}} \sum_{\substack{u \in \{0,1\}, \mathbf{pk} \in \mathcal{PK}, \\ w \in \{0,1\}^\ell}} |w\rangle_C |\mathbf{pw}_u\rangle_A |\mathbf{pw}_u\rangle_X |pk\rangle_B |pk\rangle_Y \\ |pk \odot H_1(\mathbf{pw}_u, w)\rangle_{Z_1}.$$

4. Query H_2 on registers X, Z_1 into a new register Z_2 (through $\mathcal{A}$) to compute

$$\frac{1}{\sqrt{2|\mathcal{PK}|2^\ell}} \sum_{\substack{u \in \{0,1\}, \mathbf{pk} \in \mathcal{PK}, \\ w \in \{0,1\}^\ell}} |w\rangle_C |\mathbf{pw}_u\rangle_A |\mathbf{pw}_u\rangle_X |pk\rangle_B |pk\rangle_Y \\ |pk \odot H_1(\mathbf{pw}_u, w)\rangle_{Z_1} |w \oplus H_2(\mathbf{pw}_u, Z_1)\rangle_{Z_2}.$$

5. Measure Z_2, Z_1 to get (s, T) and leftover state ρ_{CABXY}^0 .

6. For each $u \in \{0, 1\}$, compute

$$m_u = H_2(\mathbf{pw}_u, T), w_u = s \oplus m_u, \mathbf{pk}_u = H_1(\mathbf{pw}_u, w_u)^{-1} \odot T.$$

7. Let $|\psi\rangle = \frac{1}{\sqrt{2}} \sum_{u \in \{0,1\}} |w_u\rangle_C |\mathbf{pw}_u\rangle_A |\mathbf{pw}_u\rangle_X |\mathbf{pk}_u\rangle_B |\mathbf{pk}_u\rangle_Y$ and apply the measurement $M = \{|\psi\rangle\langle\psi|, I - |\psi\rangle\langle\psi|\}$ to ρ^0 .

8. Output 1 if M accepts: otherwise output 0.

The above $\mathcal{Z}_0$ works exactly the same as for ideal cipher OEKE: the key point is that given a candidate password $\mathbf{pw}$, decryption of (s, T) under $\mathbf{pw}$ not only recovers the public key but also the randomness w used to encrypt, so the post-measurement state $|\psi\rangle$ can be computed. Put another way, the computation of the first message is classically reversible given $\mathbf{pw}$. $\mathcal{Z}_1$ is modified in the same way: since the only difference between $\mathcal{Z}_0$ and $\mathcal{Z}_1$ through Step 6 is how public keys are sampled, we can again reduce to public key uniformity.

The case of Tempo is analogous:

Environment $\mathcal{Z}_0$

1. Sample $b \leftarrow \{0, 1\}$ and send $(\text{NewSession}, \text{sid}, P_2, P_1, \text{pw}_b)$ to P_2 for some nonce sid .
2. Prepare the normalized state

$$\frac{1}{\sqrt{2|\mathcal{PK}_1|2^{\ell+\ell'}}} \sum_{\substack{u \in \{0,1\}, t \in \mathcal{PK}_1, \\ w \in \{0,1\}^\ell, \zeta \in \{0,1\}^{\ell'}} |w\rangle_{\mathcal{C}} |\zeta\rangle_{\mathcal{W}} |\text{pw}_u\rangle_{\mathcal{A}} |\text{pw}_u\rangle_{\mathcal{X}} |t\rangle_{\mathcal{B}} |t\rangle_{\mathcal{Y}}.$$

3. Query H_1 on registers $\mathcal{X}, \mathcal{W}, \mathcal{C}$ into a new register $\mathcal{Z}_1$ (through $\mathcal{A}$) to compute

$$\frac{1}{\sqrt{2|\mathcal{PK}_1|2^{\ell+\ell'}}} \sum_{\substack{u \in \{0,1\}, t \in \mathcal{PK}_1, \\ w \in \{0,1\}^\ell, \zeta \in \{0,1\}^{\ell'}} |w\rangle_{\mathcal{C}} |\zeta\rangle_{\mathcal{W}} |\text{pw}_u\rangle_{\mathcal{A}} |\text{pw}_u\rangle_{\mathcal{X}} |t\rangle_{\mathcal{B}} |t\rangle_{\mathcal{Y}} \\ |H_1(\text{pw}_u, \zeta, w) \odot t\rangle_{\mathcal{Z}_1}.$$

4. Query H_2 on registers $\mathcal{X}, \mathcal{W}, \mathcal{Z}_1$ into a new register $\mathcal{Z}_2$ (through $\mathcal{A}$) to compute

$$\frac{1}{\sqrt{2|\mathcal{PK}_1|2^{\ell+\ell'}}} \sum_{\substack{u \in \{0,1\}, t \in \mathcal{PK}_1, \\ w \in \{0,1\}^\ell, \zeta \in \{0,1\}^{\ell'}} |w\rangle_{\mathcal{C}} |\zeta\rangle_{\mathcal{W}} |\text{pw}_u\rangle_{\mathcal{A}} |\text{pw}_u\rangle_{\mathcal{X}} |t\rangle_{\mathcal{B}} |t\rangle_{\mathcal{Y}} \\ |H_1(\text{pw}_u, \zeta, w) \odot t\rangle_{\mathcal{Z}_1} |w \oplus H_2(\text{pw}_u, \zeta, \mathcal{Z}_1)\rangle_{\mathcal{Z}_2}.$$

5. Measure $\mathcal{Z}_2, \mathcal{Z}_1, \mathcal{W}$ to get s, T, ζ and leftover state $\rho_{\mathcal{CABXY}}^0$.

6. For each $u \in \{0, 1\}$, compute

$$m_u = H_2(\text{pw}_u, \zeta, T), w_u = s \oplus m_u, t_u = H_1(\text{pw}_u, \zeta, w_u)^{-1} \odot T.$$

7. Let $|\psi\rangle = \frac{1}{\sqrt{2}} \sum_{u \in \{0,1\}} |w_u\rangle_{\mathcal{C}} |\text{pw}_u\rangle_{\mathcal{A}} |\text{pw}_u\rangle_{\mathcal{X}} |t_u\rangle_{\mathcal{B}} |t_u\rangle_{\mathcal{Y}}$ and apply the measurement $M = \{|\psi\rangle\langle\psi|, I - |\psi\rangle\langle\psi|\}$ to ρ^0 .

8. Output 1 if M accepts: otherwise output 0.

In $\mathcal{Z}_1$, we still prepare a superposition over all ζ , but instead compute t via $t = \text{Split}(\text{KeyGen2}(1^\lambda, \zeta)[1])[0]$. Thus we rely on the assumption that when ζ is chosen uniformly, an efficient QPT adversary cannot distinguish $(t \leftarrow \mathcal{PK}_1, \zeta \in \{0, 1\}^{\ell'})$ and $\zeta \in \{0, 1\}^{\ell'}$ with t as above. This is implied by the $\text{UNI} - \text{PK}^+$ assumption in [ABJ26], and is necessary for the post-quantum security of Tempo by the same logic as the necessity of public key uniformity for OEKE: otherwise an adversary can mount an offline attack by decrypting the first message under many candidate passwords.

□

B Modified Measure-and-Reprogram

We sketch why V can be given access to H in Theorem 3.5. First we introduce some notation from [DFM20]. Let $|\phi_i\rangle$ be defined as $\mathcal{A}$'s state right before making its $i + 1$ st query—with the special case $|\phi_q\rangle$ denoting the final output state—to which we add the superscript $\mathcal{O}$ when all previous queries have been answered using $\mathcal{O}$. Next, we use $\mathcal{A}_{i \rightarrow j}^{\mathcal{O}}$ to denote the unitary that brings $\mathcal{A}$ from $|\phi_i\rangle$ to $|\phi_j\rangle$, using $\mathcal{O}$ from the i -th query on. Finally, we use the shorthand $X := |x\rangle\langle x|$. The measure-and-reprogram proof starts with the following lemma:

Lemma B.1. *Let $\mathcal{A}$ be a q -query oracle quantum algorithm. Then, for any function $H : \mathcal{X} \rightarrow \mathcal{Y}$, any $x \in \mathcal{X}, \Theta \in \mathcal{Y}$, and any projector $\Pi_{x,\Theta}$, it holds that*

$$\mathbb{E}_{i,b} \left[\left\| (X \otimes \Pi_{x,\Theta}) \left(\mathcal{A}_{i+b \rightarrow q}^{H(x*\Theta)} \right) (\mathcal{A}_{i \rightarrow i+b}^H) X |\phi_i^H\rangle \right\|^2 \right] \geq \frac{\left\| (X \otimes \Pi_{x,\Theta}) |\phi_q^{H(x*\Theta)}\rangle \right\|^2}{(2q+1)^2}.$$

Here the expectation is over a random pair $(i, b) \in (\{0, \dots, q-1\} \times \{0, 1\}) \cup (q, 0)$. The projector $\Pi_{x, \Theta}$ will function as V later on. Notice that the expectation is only over the randomness of the measure-and-reprogram simulator, and it holds for any *fixed* $x, \Theta, \Pi_{x, \Theta}, H$. Therefore the same lemma holds for a projector $\Pi_{x, \Theta}^{H(x * \Theta)}$ that gets access to the reprogrammed random oracle.

The next lemma is proven by induction using the previous lemma as a base case:

Lemma B.2. *Let $\mathcal{A}$ be a q -query oracle quantum algorithm. Then, for any function $H : \mathcal{X} \rightarrow \mathcal{Y}$, any $\mathbf{x} \in \mathcal{X}^n$, $\Theta \in \mathcal{Y}^n$, and any projector $\Pi_{\mathbf{x}, \Theta}$, it holds that*

$$\frac{\|(|\mathbf{x}\rangle\langle\mathbf{x}| \otimes \Pi_{\mathbf{x}, \Theta}) \mathcal{A}^{H * \Theta_{\mathbf{x}}} |\phi_0\rangle\|^2}{(2q+1)^{2n}} \leq \mathbb{E}_{\mathbf{r}} \left[\left\| (|\mathbf{x}\rangle\langle\mathbf{x}|_A \otimes \Pi_{\mathbf{x}, \Theta}) S_{\mathbf{r}}^H(\mathcal{A}) |\phi_0\rangle \right\|^2 \right].$$

The notation required to define $S_{\mathbf{r}}^H(\mathcal{A})$ formally is quite complex so we omit it, only noting that $S_{\mathbf{r}}^H(\mathcal{A})$ is the simulator that chooses n queries $\mathbf{r} = (r_1, \dots, r_n)$ of $\mathcal{A}$ to H to measure and reprogram. The key observation is that the lemma once again holds for any fixed $H, \mathbf{x}, \Theta, \Pi_{\mathbf{x}, \Theta}$ so it also holds for a projector $\Pi_{\mathbf{x}, \Theta}^{H(\mathbf{x} * \Theta)}$.

Finally, we come to the proof of [DFM20, Theorem 6], the statement of which is the same as Theorem 3.5 but V is not given oracle access. The proof proceeds by taking the expectation of Lemma B.2 over random H, Θ

$$\begin{aligned} & \mathbb{E}_{H, \Theta} \frac{\|(|\mathbf{x}\rangle\langle\mathbf{x}| \otimes \Pi_{\mathbf{x}, \Theta}) \mathcal{A}^{H * \Theta_{\mathbf{x}}} |\phi_0\rangle\|^2}{(2q+1)^{2n}} \\ & \leq \mathbb{E}_{\mathbf{r}, H, \Theta} \left[\left\| (|\mathbf{x}\rangle\langle\mathbf{x}|_A \otimes \Pi_{\mathbf{x}, \Theta}) S_{\mathbf{r}}^H(\mathcal{A}) |\phi_0\rangle \right\|^2 \right] \end{aligned}$$

and noting that after a change of variables the left hand side equals

$$\mathbb{E}_H \frac{\|(|\mathbf{x}\rangle\langle\mathbf{x}| \otimes \Pi_{\mathbf{x}, H(\mathbf{x})}) \mathcal{A}^H |\phi_0\rangle\|^2}{(2q+1)^{2n}}.$$

If we repeat the argument with our new projector $\Pi_{\mathbf{x}, \Theta}^{H(\mathbf{x} * \Theta)}$ the same change of variables occurs: the left hand side becomes

$$\mathbb{E}_H \frac{\left\| \left(|\mathbf{x}\rangle\langle\mathbf{x}| \otimes \Pi_{\mathbf{x}, H(\mathbf{x})}^H \right) \mathcal{A}^H |\phi_0\rangle \right\|^2}{(2q+1)^{2n}}.$$

Note that here it is important that we give $\Pi_{\mathbf{x}, \Theta}$ access to the reprogrammed H , *not* the original H : otherwise $\Pi_{\mathbf{x}, \Theta}$ and $\mathcal{A}$ would be accessing different oracles. This makes intuitive sense since $\Pi_{\mathbf{x}, \Theta}$ is applied after the oracle is reprogrammed. With that the proof of the main measure-and-reprogram theorem goes through the same way.