IACR Transactions on Symmetric Cryptology
ISSN 2519-173X, Vol. 2022, No. 2, pp. 379–414. DOI:10.46586/tosc.v2022.i2.379-414

On the Quantum Security of OCB

Varun Maram[1§], Daniel Masny[2¶], Sikhar Patranabis[3‖] and Srinivasan Raghuraman[4]

[1] Department of Computer Science, ETH Zürich, Zürich, Switzerland
vmaram@inf.ethz.ch
[2] Meta Research, Menlo Park, USA
daniel.masny@rub.de
[3] IBM Research, Bangalore, India
sikharpatranabis@gmail.com
[4] Visa Research, Palo Alto, USA
srini131293@gmail.com

Abstract. The OCB mode of operation for block ciphers has three variants, OCB1, OCB2 and OCB3. OCB1 and OCB3 can be used as secure authenticated encryption schemes whereas OCB2 has been shown to be classically insecure (Inoue *et al.*, Crypto 2019). Even further, in the presence of quantum queries to the encryption functionality, a series of works by Kaplan *et al.* (Crypto 2016), Bhaumik *et al.* (Asiacrypt 2021) and Bonnetain *et al.* (Asiacrypt 2021) have shown how to break the existential unforgeability of the OCB modes. However, these works did not consider the confidentiality of OCB in the presence of quantum queries.

We fill this gap by presenting the first formal analysis of the IND-qCPA security of OCB. In particular, we show the first attacks breaking the IND-qCPA security of the OCB modes. Surprisingly, we are able to prove that OCB2 is IND-qCPA secure when used without associated data, while relying on the assumption that the underlying block cipher is a quantum-secure pseudorandom permutation. Additionally, we present new quantum attacks breaking the universal unforgeability of OCB. Our analysis of OCB has implications for the post-quantum security of XTS, a well-known disk encryption standard, that was considered but mostly left open by Anand et al. (PQCrypto 2016).

Keywords: OCB · IND-qCPA security · universal forgeability · Simon's Algorithm · Deutsch's Algorithm · XTS

1 Introduction

The development of large-scale quantum computers has potentially significant implications for the security of existing cryptographic algorithms. In particular, widely used public-key cryptographic algorithms based on the hardness of computing discrete-logs or factorization would suffer from devastating attacks due to Shor's algorithm [Sho99]. This has led to the advent of post-quantum cryptography (PQC), which aims to design cryptographic primitives resistant to quantum adversaries, i.e. adversaries equipped with quantum computers. The study of post-quantum secure public-key cryptography has received significant attention within the cryptographic community over the last decade [Reg02, Reg10, AKC+17], including an ongoing effort by NIST to standardize post-quantum secure public-key primitives [AAAS+19, AASA+20].

§Work done while the author was an intern at Visa Research.
¶Work done while the author was at Visa Research.
‖Work done while the author was at Visa Research.

Received: 2021-11-23 Revised: 2022-03-01 Accepted: 2022-05-01 Published: 2022-06-10

Post-Quantum Symmetric-Key Cryptography. The impact of quantum computers on symmetric-key cryptographic primitives is somewhat less understood, and was, for a long time, believed to be significantly less severe. Indeed, generic attacks against ideal symmetric-key primitives such as pseudorandom permutations (PRPs), and their real-world instantiations such as block ciphers, would only gain a quadratic speed-up due to Grover's algorithm [Gro96]. This seems to indicate that simply doubling the key length would essentially restore equivalent classical security against quantum adversaries; indeed, the community initially considered the situation of post-quantum secure symmetric-key cryptography settled based purely on this observation [Ber09]. However, a number of recent works have revisited the need for new security definitions and frameworks for handling quantum adversaries, leading to interesting observations [BDF+11, BZ13a, BZ13b, DFNS14, BJ15, GHS16, ABF+16].

In particular, a line of recent results [KM10, KM12, ATTU16, KLLN16, SS17, BHN+19, BBC+21, BLNS21] have shown that quantum computers can actually lead to devastating attacks targeting the way such ideal primitives are operated in practice, and simple techniques such as key length doubling do not prevent such *quantum* attacks. That is, a standalone block cipher is rarely used in practice because it can only encrypt messages of a fixed (and usually very short) length; real-world implementations typically extend the message space of a block cipher (and add other desirable features and/or security guarantees) by using it in conjunction with a so-called "mode of operation". Hence, in practice, it does not suffice to only consider the post-quantum security of the block cipher as a standalone primitive; the post-quantum security of the mode of operation is also of paramount importance.

Post-Quantum Security of Modes. A number of recent works [ATTU16, KLLN16, BBC+21, BLNS21] have studied the post-quantum security of modes of operation for block ciphers, message-authentication codes (MACs) and authenticated encryption (AE). The analysis is typically done in the quantum *superposition model*, where the adversary is capable of accessing a quantum encryption oracle, and of encrypting quantum states (i.e., the adversary can issue encryption queries which are quantum superpositions of different messages, to receive superpositions of different ciphertexts). While the practical significance of attacks in the superposition model is not fully settled within the community (see [ATTU16, KLLN16, BBC+21] for discussions), analyzing the post-quantum security of symmetric-key primitives in this model is receiving significant attention.

Anand *et al.* [ATTU16] studied a number of common modes of operation, namely those listed in the 2013 ENISA[1] report on recommended encryption algorithms [fNE13]: CBC, CFB, OFB, CTR, and XTS. They studied the confidentiality guarantees of these modes with respect to a security definition called "IND-qCPA security" [BZ13b], which, unlike the classical notion of IND-CPA security, allows the adversary to issue quantum encryption queries as outlined earlier. They showed that many of these modes (notably, CBC and CFB), which are classically IND-CPA secure, are rendered insecure in the IND-qCPA setting by applications of Simon's quantum period-finding algorithm [Sim94]. This clearly motivates studying the post-quantum security of modes of operation.

The OCB Mode. OCB [RBBK01, Rog04, KR11] is one of the most influential and well-studied authenticated encryption modes with three variants – OCB1 [RBBK01], OCB2 [Rog04], and OCB3 [KR11]. OCB1 and OCB3 were proven to be classically secure AE schemes under the assumption that the underlying block cipher is a classically secure strong PRP [RBBK01, BN17]. On the other hand, a breakthrough result due to Inoue *et al.* [IIMP19] exposed devastating breaks of the classical security of OCB2 when used as an AE scheme (even though OCB2 is IND-CPA secure when the underlying block cipher

[1] European Union Agency for Network and Information Security

is a classically secure PRP [Rog04]). These attacks led to the proposal of OCB2f – an alternative to OCB2 that was proven in [IIMP19] to be a classically secure AE scheme.

Post-Quantum Security of OCB. Anand *et al.* [ATTU16] did not study the post-quantum security of OCB. However, other works due to Kaplan *et al.* [KLLN16] and, more recently, Bhaumik *et al.* [BBC+21] have studied the post-quantum security of OCB as an AE mode. More concretely, their analyses focus on the authenticity guarantees provided by OCB when used as an AE in the post-quantum setting (more formally, existential unforgeability security against quantum chosen-message attacks, or EUF-qCMA security [BZ13a] in short).

Kaplan *et al.* [KLLN16] analyzed the EUF-qCMA security of each of the three variants of the OCB mode in a quantum superposition model, where the attacker has access to a superposition oracle that given a superposition of messages as input, returns the superposition of their encryptions. The key is a secret value and the nonce is chosen at random by the oracle for each query. In this attack setting, Kaplan *et al.* showed that all three variants of OCB are broken (as AE schemes) via existential forgery attacks. They also showed that OCB1 and OCB3 can be broken by additional forgery attacks that specifically target the encryption part, while ignoring the authentication tag generated as part of the output. All of these attacks are based on applications of Simon's quantum period finding algorithm [Sim94].

Bhaumik *et al.* [BBC+21] extended the aforementioned analysis by considering natural variants of OCB that are resistant to attacks due to Kaplan *et al.* and subsequently describing new ways to attack such modified OCB modes in the quantum setting. The authors of [BBC+21] concluded that the existential forgery attacks on OCB arise inherently from the fact that OCB uses an underlying tweakable block cipher that is not quantum-secure, and simple modifications are unlikely to salvage its EUF-qCMA security.

Finally, in another recent work, Bonnetain *et al.* [BLNS21] showed a new quantum attack against the EUF-qCMA security of OCB3. Unlike all other existing attacks that rely fundamentally on Simon's period finding algorithm, this attack relies on Deutsch's algorithm [Deu85], which has been much less commonly used in the quantum cryptanalysis of symmetric-key cryptoprimitives.

IND-qCPA Security of OCB. Interestingly, none of the aforementioned works analyzed the confidentiality guarantees of OCB in the quantum superposition model (i.e., IND-qCPA security [BZ13b]). Of course, forgery attacks immediately rule out any possibility of using OCB as an AE scheme in the post-quantum setting. However, we believe that investigating its IND-qCPA security remains a question of independent interest.

In particular, from a design perspective, it would be desirable to upgrade OCB to a post-quantum secure AE scheme while making only minimal changes to the original specifications. For this purpose, it seems important to understand whether OCB at least satisfies confidentiality in the post-quantum setting (in which case one could hope to make minimal changes to fix the issues related to authenticity), or is also broken via attacks on confidentiality (which indicates the need for more fundamental changes).

This motivates us to ask the following question:

Are the OCB modes IND-qCPA secure?

Post-Quantum Universal Unforgeability of OCB. Similarly, none of the aforementioned works analyzed the universal unforgeability guarantees of OCB in the quantum superposition model (more formally, the "UUF-qCMA" security of OCB). We note here that similar to the classical setting, quantum universal unforgeability attacks are seemingly harder to launch as compared to quantum existential unforgeability attacks, since the former

requires the quantum adversary to output a forgery on a (classical) message chosen by the challenger in the unforgeability security game, while the latter allows the adversary to output a forgery on any (classical) message of its choice (subject to certain restrictions that we make formal subsequently).

We again believe that investigating the universal unforgeability guarantees of OCB in the post-quantum setting is an interesting open question. In fact, universal forgeries are seemingly more devastating since they allow the adversary to forge signatures on *any* message, whereas existential forgeries might only be possible for atypical or artificial messages. This motivates us to ask the following question:

Are the OCB modes UUF-qCMA secure?

1.1 Our Contributions

In this paper, we make significant progress towards answering the aforementioned questions by presenting appropriate positive and negative results with respect to the confidentiality and universal unforgeability of OCB in the post-quantum setting. Along the way, we also present some new insights into existential forgery attacks against OCB2 and OCB2f in the post-quantum setting. We also note that our results constitute the first concrete attempt to analyze the post-quantum security of OBC2f with respect to both confidentiality and unforgeability[2], which has not been addressed in any prior work to the best of our knowledge. Our main results are summarized below.

OCB1 and OCB3 are NOT IND-qCPA Secure. We present the first quantum attacks breaking the IND-qCPA security of OCB1 and OCB3. Our attacks rely on Simon's quantum period finding algorithm, and work in the weakest attack setting where the nonces used by the challenger in the IND-qCPA security game are uniformly randomly sampled *and hidden* from the adversary. Our attacks hold even if the underlying block ciphers used by OCB1 and OCB3 are quantum-secure PRPs. This is in contrast to the results in [ATTU16], which showed that certain popular block-cipher modes of operation such as CBC, CFB, OFB and CTR are IND-qCPA secure when used with a quantum-secure PRP. Finally, our attack works even if OCB3 is used as a "pure" AE mode where the associated data (AD) is fixed to be empty always (OCB1 is a pure AE mode by default since it does not support any AD as part of the input to the encryption algorithm). As we discuss next, whether the AD is empty or not makes a difference to the IND-qCPA security of OCB2 and OCB2f; however, OCB1 and OCB3 are broken regardless.

OCB2(f) is IND-qCPA Secure for "Empty" AD. We present the first proofs of IND-qCPA security for OCB2 and OCB2f when used as pure AE modes with empty associated data (AD) and random nonces. Concretely, our proofs of IND-qCPA security hold under the assumption that the following hold simultaneously:

- The block cipher underlying the OCB2 and OCB2f modes is a quantum-secure PRP.
- The nonces used by the challenger in the IND-qCPA security game are uniformly randomly sampled (we note here that this is a commonly accepted notion of quantum security in the literature for nonce-based cryptographic schemes; e.g., see [KLLN16])[3].

[2] We note that the EUF-qCMA attacks against the original three OCB modes due to Kaplan *et. al.* [KLLN16] do apply to OCB2f in a straightforward manner.

[3] It turns out that our proofs of IND-qCPA security for OCB2 and OCB2f as pure AE modes no longer hold when the adversary can adaptively pick the (classical) nonces for the encryption queries in the IND-qCPA security game. In fact, in this strong adversarial setting, it is straightforward to design a quantum attack on the IND-qCPA security of OCB2 and OCB2f. However, we believe that such an adversarial model is unreasonably strong and does not reflect the real-world security of using OCB2, where the adversary typically cannot program the nonces used by the encryption algorithm.

We note here that our proof of IND-qCPA security of OCB2 (with random nonces) when operated as a pure AE mode does not contradict the findings of [IIMP19] with respect to the classical confidentiality of OCB2. Indeed, the authors of [IIMP19] showed that OCB2 fails to offer confidentiality in an IND-CCA sense, while simultaneously arguing that there were no obvious issues with its classical IND-CPA security, which was proven formally in [Rog04].

However, our result is surprising and somewhat non-intuitive for a different reason. To begin with, OCB2 was proven to be classically IND-CPA secure in [Rog04] under the assumption that the underlying block cipher is a classically secure tweakable block cipher. However, as already mentioned, Kaplan *et al.* [KLLN16] and Bhaumik *et al.* [BBC+21] have argued that the tweakable block cipher underlying OCB2 is not quantum-secure. In fact (as a side contribution), we actually extend their arguments to show that the tweakable block cipher underlying OCB2 remains insecure in the quantum setting even when tweaks are not allowed to repeat across queries, which is the case w.r.t. OCB2's encryption algorithm. As a result, a similar proof strategy as the classical setting does not work when proving the IND-qCPA security of OCB2.

Of course, replacing the underlying tweakable block cipher underlying OCB2 with a quantum-secure counterpart leads to a more modular analysis, as was done in [BBC+21]. However, we are interested in the IND-qCPA security of *original* OCB2 when used as a pure AE mode, and hence we do not intend to change its specification in any way. To this end, we can only rely on a weaker assumption – namely, that the block cipher underlying the OCB2 and OCB2f modes is a quantum-secure PRP. Our proof of IND-qCPA security is therefore more involved, and is inspired by the IND-qCPA security proofs for CBC and CFB presented in [ATTU16].

OCB2(f) is NOT IND-qCPA Secure for "Non-Empty" AD. We extend our analysis to present the first quantum attacks breaking the IND-qCPA security of OCB2 and OCB2f when the AD is not empty during encryption operations. Similar to our attacks on OCB1 and OCB3, these attacks again work in a weak attack setting where the nonces used by the challenger in the IND-qCPA security game are uniformly randomly sampled. However, unlike the attacks on OCB1 and OCB3 that rely fundamentally on Simon's algorithm, our attacks on OCB2 and OCB2f rely on Deutsch's algorithm [Deu85], and are inspired by the existential forgery attack against OCB3 based on Deutsch's algorithm described in [BLNS21]. Finally, our attacks hold even if the underlying block cipher is a quantum-secure PRP.

It is interesting to note that our results on the IND-qCPA security of OCB2 contrast strongly in flavor with the *classical* cryptanalysis results for OCB2 proposed in [IIMP19]. A key component of *all* the classical attacks against OCB2 proposed in [IIMP19] is that they *require* the AD to be empty. In fact,the authors of [IIMP19] suggest that a potential fix to OCB2 is to *always* keep the AD non-empty during encryption/decryption operations. This is in stark contrast to our IND-qCPA security analysis of OCB2, where we show a positive result when AD is always kept empty and a negative result–via an attack–which exploits the processing of (non-empty) AD during encryption.

OCB2(f) and OCB3 are NOT UUF-qCMA Secure. We present the first quantum attacks breaking the universal unforgeability of OCB2[4], OCB2f and OCB3 in the superposition model. Our attacks work in the weakest attack setting where the nonces used by the challenger in the unforgeability security game are uniformly randomly sampled.

Our attacks on OCB2 and OCB2f actually follow from an extension of our attacks against the IND-qCPA security of these modes when used with non-empty AD during encryption. Our attack on OCB3 can be viewed as a strengthening of the existential

[4]Though *classical* universal forgery attacks against OCB2 were already presented in [IIMP19].

forgery attack against OCB3 based on Deutsch's algorithm described in [BLNS21]. A common principle underlying both these attacks is the following: a quantum adversary can use non-empty AD inputs during encryption together with Deutsch's algorithm to gain the capability of evaluating the underlying block cipher on inputs of its choice. Given such a *raw* block-cipher access, it is straightforward for the adversary to compute universal forgeries with respect to any (nonce, AD, message) tuple in the universal unforgeability game.

OCB1 is NOT UUF-qCMA Secure for Adaptively Chosen Nonces. We show a quantum attack breaking the universal unforgeability of OCB1, albeit in a stronger adversarial model where the adversary is allowed to adaptively specify the (classical) nonces for its queries in the unforgeability security game. The requirement of a stronger attack setting stems from the fact that OCB1 is a pure AE mode that does not support any AD as part of the input to the encryption algorithm. In particular, our quantum universal forgery attacks on OCB2(f) and OCB3 do not naturally extend to OCB1.

We leave it open to extend our quantum universal forgery attack on OCB1 to the random nonce setting, or alternatively, to formally prove the UUF-qCMA security of OCB1 in the random nonce setting. Establishing either of these would formally resolve the question of whether OCB1 is inherently more resistant to quantum universal forgery attacks in the random nonce setting as compared to OCB2(f) and OCB3.

New Attacks on EUF-qCMA Security of OCB2(f). Finally, as a side contribution, we present a new quantum attack breaking the existential unforgeability (EUF-qCMA) security of OCB2 and OCB2f that specifically targets the encryption part, while ignoring the authentication tag generated as part of the output. Our attack is based on a new application of Simon's quantum period finding algorithm presented in [BBC+21], and is similar in flavor to the specialized quantum forgery attacks on OCB1 and OCB3 described in [KLLN16]. To the best of our knowledge, such an attack tailored to the specification of either OCB2 or OCB2f was not known previously (the previous attack on OCB2 in [KLLN16] explicitly targeted the authentication tag).

Our attack concretely rules out the possibility of salvaging the post-quantum authenticity of OCB2(f) by making modifications to only the underlying tag-generation component (more formally, the PMAC component). In other words, any fixes to the post-quantum authenticity of OCB2(f) must *necessarily* modify *both* the encryption and authentication components. While such a fix was proposed in [BBC+21], the necessity of a fix of this nature was not previously established to the best of our knowledge.

1.2 Discussions: Implications for Post-Quantum Security of XTS

Our analysis of OCB2 also has interesting implications for the post-quantum security of the XTS mode of operation, which is standardized by both NIST and IEEE [Dwo, oEE08]. The sole intended use of XTS is as a tweakable block cipher for encrypting data on a storage device, such as a disk [Rog11] (e.g., the tweak could be derived from the sector number on the disk and the index of the data block within the sector). XTS is structurally similar to OCB2 except for the fact that XTS uses two secret keys during encryption and decryption, unlike OCB2 which uses a single key. In fact, as pointed out by Rogaway in [Rog11], XTS is essentially the two-key version of the original single-key XEX construction from [Rog04], which also forms the core of OCB2. Consequently, our positive and negative results for the post-quantum security of OCB2 also provide some insight into the post-quantum security of XTS[5] when used as a disk encryption scheme, under appropriate assumptions.

[5]The classical security of XTS has been (to the best of our knowledge) analyzed by only a few works in the literature, such as [Rog11, KMV17, IM19].

IND-qCPA Security of XTS. Our proof of IND-qCPA security of OCB2 when used as a pure AE mode (i.e., with empty AD) with uniformly random nonces can be easily extended to prove the IND-qCPA security of XTS when used as a disk encryption scheme, under the assumption that each sector number is uniformly randomly chosen and that the length of messages is a multiple of the block size of the underlying block-cipher of XTS. While the practical validity of these assumptions may be debated, we note that this constitutes the first proof of IND-qCPA security for XTS in any setting; the prior work due to Anand *et al.* [ATTU16] left analyzing the IND-qCPA security of XTS as an entirely open question. Thus, our results are the first to establish some amount of confidence in the post-quantum confidentiality of XTS as a disk encryption scheme (note that XTS is a "confidentiality-only" mode by design).

On the Need for Two Keys in XTS. It turns out that our proof of IND-qCPA security for OCB2 as a pure AE mode no longer holds when the adversary can adaptively pick the (classical) nonces for the encryption queries in the IND-qCPA security game. In fact, in this strong adversarial setting, it is straightforward to design a quantum attack on the IND-qCPA security of OCB2. Interestingly, this attack does not extend to XTS even when the adversary can adaptively choose the tweaks (equivalently, nonces or disk sector numbers) in the IND-qCPA security game for XTS. More concretely, while the adaptive nonce-based attack breaks IND-qCPA security of the original single-key XEX construction from [Rog04], it fails against the two-key version of XEX that XTS is based on. This provides some insight into why it might actually be useful to have two keys for XTS in the post-quantum setting, and partially answers the question raised by Liskov and Minematsu in [LM08] on whether using two keys in XTS actually offers any advantage with respect to confidentiality over the original single-key XEX construction from [Rog04].

At the same time, we would like to point out that when it comes to security models for disk encryption schemes in the literature [KMV17, IM19], an adversary is allowed to *overwrite* data on the same sector – i.e., disk sector numbers cannot be used as nonces in this context. Hence, our above security analysis applies to XTS when interpreted as a "nonce-based" symmetric encryption scheme, with the sector numbers acting as nonces. We remark that this is the same setting considered by Anand *et al.* [ATTU16] and which will also be considered in the rest of this paper, particularly in Section 7.

1.3 Paper Outline

The rest of the paper is organized as follows. Section 2 presents preliminary background material. Section 3 describes our quantum attacks breaking the IND-qCPA security of OCB1 and OCB3. Section 4 describes our proof of IND-qCPA security for OCB2(f) with empty associated data, and our quantum attack breaking the IND-qCPA security for OCB2(f) with non-empty associated data. Section 5 describes our new quantum attack breaking the existential unforgeability (EUF-qCMA) security of OCB2(f) while specifically targeting the encryption part, and ignoring the authentication part. Section 6 presents our new quantum attacks breaking the universal unforgeability (UUF-qCMA) security of the different variants of OCB in various attack settings. Section 7 describes some implications of our IND-qCPA analysis of OCB2 for the post-quantum security of XTS. Finally, Section 8 concludes the paper and discusses some open questions.

2 Preliminaries

Notations. We use κ to denote the security parameter. For a set X, we use $x \leftarrow X$ to denote the process of sampling a uniform $x \in X$. For $n \in \mathbb{N}$, $[n] := \{1, \ldots, n\}$. For

$x \in \{0,1\}^*$, $y \in \{0,1\}^*$, we use $x \| y$ to denote the concatenation of x and y.

In the context of the OCB modes, for $x \in \{0,1\}^*$, $\tau \in \mathbb{N}$, we use $\mathsf{msb}_\tau(x)$ to denote the τ most significant bits of x. We use $\mathsf{len}(x)$ to denote the bit length of x.

Authenticated Encryption with Associated Data (AEAD). We start with the formal syntax of an AEAD scheme.

Definition 1 (Authenticated Encryption with Associated Data (AEAD))**.** An AEAD scheme is a tuple of PPT (probabilistic polynomial-time) algorithms $\Pi = (\mathsf{Enc}, \mathsf{Dec})$ and a key space $\{0,1\}^\kappa$ with the following syntax.

$\mathsf{Enc}(\mathsf{K}, \mathsf{N}, \mathsf{A}, \mathsf{M})$: Takes as input a key K, a nonce N, associated data A and a message M and outputs a ciphertext c and tag t. We simplify the syntax by sometimes denoting this process as $(\mathsf{c}, \mathsf{t}) \leftarrow \mathsf{Enc}_\mathsf{K}(\mathsf{N}, \mathsf{A}, \mathsf{M})$.

$\mathsf{Dec}(\mathsf{K}, \mathsf{N}, \mathsf{A}, \mathsf{c}, \mathsf{t})$: Takes as input a key K, a nonce N, associated data A, a ciphertext c and a tag t and outputs a message M or $\perp$. We sometimes denote the output of this decryption by $\mathsf{Dec}_\mathsf{K}(\mathsf{N}, \mathsf{A}, \mathsf{c}, \mathsf{t})$.

An AEAD is called correct if for any N, A and M,

$$\Pr[\mathsf{Dec}_\mathsf{K}(\mathsf{N}, \mathsf{A}, \mathsf{Enc}_\mathsf{K}(\mathsf{N}, \mathsf{A}, \mathsf{M})) = \mathsf{M}] \geq 1 - \mathsf{negl},$$

where $\mathsf{K} \leftarrow \{0,1\}^\kappa$.

For an AEAD scheme, we define the IND-qCPA security as follows.

Definition 2 (IND-qCPA with random nonces, adapted from [BZ13b])**.** An AEAD scheme $\Pi = (\mathsf{Enc}, \mathsf{Dec})$ is indistinguishable under chosen message attack (IND-qCPA secure) with random nonces if no efficient quantum adversary $\mathcal{A}$ can win in the following game, except with probability at most $1/2 + \mathsf{negl}$:

- **Key generation:** The challenger picks a random key $\mathsf{K} \leftarrow \{0,1\}^\kappa$ and a random bit $b \leftarrow \{0,1\}$.
- **Queries:** $\mathcal{A}$ is allowed to make two types of queries, in any order:
  - **Encryption queries:** First, the challenger picks a random nonce $\mathsf{N} \leftarrow \{0,1\}^\kappa$ and forwards it to $\mathcal{A}$. Then $\mathcal{A}$ chooses a message and AD pair (M, A), possibly in superposition. The challenger encrypts $(\mathsf{N}, \mathsf{A}, \mathsf{M})$ with the classical nonce N and returns the output (c, t) to $\mathcal{A}$.
  - **Challenge query:** Again, the challenger picks a random nonce $\mathsf{N} \leftarrow \{0,1\}^\kappa$ and forwards it to $\mathcal{A}$. Then $\mathcal{A}$ chooses two classical message and AD pairs $(\mathsf{M}_0, \mathsf{A})$, $(\mathsf{M}_1, \mathsf{A})$ of the same length and sends them to the challenger. Then the challenger encrypts $(\mathsf{N}, \mathsf{A}, \mathsf{M}_b)$ with the classical nonce N and returns the output $(\mathsf{c}^*, \mathsf{t}^*)$ to $\mathcal{A}$.
- **Guess:** $\mathcal{A}$ outputs a bit b', and wins if $b = b'$.

If $\mathcal{A}$'s winning probability in the above game is denoted by p, we also define its IND-qCPA *advantage* w.r.t. the scheme Π as $\mathsf{Adv}_\Pi^{\mathsf{IND\text{-}qCPA}}(\mathcal{A}) = \left| p - \frac{1}{2} \right|$. Hence, in more concrete terms, we have Π to be IND-qCPA secure – with random nonces – if the advantage $\mathsf{Adv}_\Pi^{\mathsf{IND\text{-}qCPA}}(\mathcal{A})$ of *any* polynomial-time quantum adversary $\mathcal{A}$ is negligible.

Comparison with IND-qCPA definitions in [BZ13b, BBC+21]. The "original" IND-qCPA notion was defined in [BZ13b] for general symmetric encryption schemes that were not *nonce-based*. In the corresponding security game, the challenger chooses classical randomness when responding to each of the adversary's encryption/challenge queries – *with the randomness kept secret*, as [BZ13b] did not model an auxiliary quantum register for randomness w.r.t. the encryption oracle in their security game. In the context of nonce-based AE schemes, this translates to the challenger picking random *and secret* nonces N. [BBC+21] strengthens this definition by allowing an adversary to choose the nonces in a non-repeating *and non-adaptive fashion* – i.e., right after the "key generation" phase, the adversary $\mathcal{A}$ sends a list of distinct nonces $\langle \mathsf{N}^{(i)} \rangle_{i \in [q]}$ to the challenger such that nonce $\mathsf{N}^{(i)}$ is to be used to process $\mathcal{A}$'s subsequent i-th (encryption or challenge) query.

Our Definition 2 above falls in-between as it is weaker than that of [BBC+21] because the adversary cannot choose the nonces, but is stronger than that of [BZ13b] because the adversary gets access to the random nonce N, chosen by the challenger, before making each query. One thing worth mentioning is that [KLLN16, Subsection 5.2] analyzes the quantum security of GMAC in a model where the adversary has access to the (quantum) oracle $\mathsf{M} \mapsto (\mathsf{N}, \mathsf{GMAC}(\mathsf{N}, \mathsf{M}))$ such that the classical nonce N is chosen randomly for each oracle query. When translating this MAC security notion to nonce-based AE schemes, this corresponds to the IND-qCPA challenger picking a random nonce N *after* the adversary makes an encryption query. Again our Definition 2 is stronger than this notion because we allow an adversary to first obtain the random nonce N from the challenger and then make an encryption query based on the nonce. At the same time, we remark that the above security model in [KLLN16] was devised in the context of cryptanalyzing schemes (in a quantum setting), i.e., the model was deliberately made to be as weak as possible which in turn made the corresponding quantum attacks in [KLLN16] strong.

Going a bit ahead, note that even though we prove the security of OCB2, with empty AD, in Section 4 w.r.t. our stronger IND-qCPA definition (compared to that of [BZ13b]), our attacks against OCB1 and OCB3 in Section 3 still apply w.r.t. the weaker IND-qCPA definition of [BZ13b] as the attacks do not require access to nonces chosen by the challenger. Finally in Section 8, we also discuss our analysis of the OCB modes in the context of other quantum security notions beyond IND-qCPA as studied in [CETU21].

We now define the EUF-qCMA and UUF-qCMA security under random nonces for an AEAD in the following way.

Definition 3 (EUF-qCMA with random nonces, adapted from [BZ13a])**.** An AEAD scheme $\Pi = (\mathsf{Enc}, \mathsf{Dec})$ is existentially unforgeable under chosen message attack (EUF-qCMA secure) with random nonces if no efficient quantum adversary $\mathcal{A}$ can win in the following game, except with a negligible probability:

- **Key generation:** The challenger picks a random key $\mathsf{K} \leftarrow \{0,1\}^\kappa$.
- **Queries:** $\mathcal{A}$ is allowed to make encryption queries as follows:
  - **Encryption queries:** First, the challenger picks a random nonce $\mathsf{N} \leftarrow \{0,1\}^\kappa$ and forwards it to $\mathcal{A}$. Then $\mathcal{A}$ chooses a message and AD pair (M, A), possibly in superposition. The challenger encrypts $(\mathsf{N}, \mathsf{A}, \mathsf{M})$ with the classical nonce N and returns the output (c, t) to $\mathcal{A}$.
- **Forgeries:** After making q encryption queries, $\mathcal{A}$ produces $q+1$ *classical* tuples $(\mathsf{N}, \mathsf{A}, \mathsf{c}, \mathsf{t})$ with any nonces Ns of its choice, and wins if for each tuple we have $\mathsf{Dec}_\mathsf{K}(\mathsf{N}, \mathsf{A}, \mathsf{c}, \mathsf{t}) \neq \perp$.

Definition 4 (UUF-qCMA with random nonces, adapted from [DDKA21])**.** An AEAD scheme $\Pi = (\mathsf{Enc}, \mathsf{Dec})$ is universally unforgeable under chosen message attack (UUF-qCMA secure) with random nonces if no efficient quantum adversary $\mathcal{A}$ can win in the following game, except with a negligible probability:

- **Key generation:** The challenger picks a random key $\mathsf{K} \leftarrow \{0,1\}^\kappa$.
- **Challenge:** The challenger picks a random nonce $\mathsf{N}^* \leftarrow \{0,1\}^\kappa$ along with a random message and AD pair $(\mathsf{M}^*, \mathsf{A}^*)$ from the corresponding spaces. It then forwards $(\mathsf{N}^*, \mathsf{A}^*, \mathsf{M}^*)$ to $\mathcal{A}$.
- **Queries:** $\mathcal{A}$ is allowed to make encryption queries as follows:
  - **Encryption queries:** First, the challenger picks a random nonce $\mathsf{N} \leftarrow \{0,1\}^\kappa$ and forwards it to $\mathcal{A}$. Then $\mathcal{A}$ chooses a message and AD pair (M, A), possibly in superposition. The challenger encrypts $(\mathsf{N}, \mathsf{A}, \mathsf{M})$ with the classical nonce N and returns the output (c, t) to $\mathcal{A}$.
- **Forgery:** After making its (polynomial-number of) encryption queries, $\mathcal{A}$ produces the *classical* tuple $(\mathsf{c}^*, \mathsf{t}^*)$, and wins if we have $\mathsf{Dec}_\mathsf{K}(\mathsf{N}^*, \mathsf{A}^*, \mathsf{c}^*, \mathsf{t}^*) = \mathsf{M}^*$.

Quantum secure PRFs and One-Way to Hiding.

Definition 5 (Quantum-secure PRF [Zha12])**.** A function $\mathsf{E}_\mathsf{K} : \{0,1\}^n \to \{0,1\}^n$ is a quantum-secure PRF if no efficient quantum adversary $\mathcal{A}$ making quantum queries can distinguish between a truly random function $H : \{0,1\}^n \to \{0,1\}^n$ and the function E_K for a uniformly random secret key $\mathsf{K} \leftarrow \{0,1\}^\kappa$.

More concretely, we define the *advantage* of $\mathcal{A}$ in distinguishing E_K from the random function H as $\mathsf{Adv}^{\mathsf{qPRF}}_{\mathsf{E}_\mathsf{K}}(\mathcal{A}) = |\Pr[1 \leftarrow \mathcal{A}^{\mathsf{E}_\mathsf{K}}] - \Pr[1 \leftarrow \mathcal{A}^H]|$, where "$\mathcal{A}^F$" denotes that $\mathcal{A}$ has quantum oracle access to $F \in \{\mathsf{E}_\mathsf{K}, H\}$. Then we have E_K to be a quantum-secure PRF if $\mathsf{Adv}^{\mathsf{qPRF}}_{\mathsf{E}_\mathsf{K}}(\mathcal{A})$ is negligible for *any* polynomial-time quantum adversary $\mathcal{A}$.

The following lemma provides a generic reduction from a hiding-style property (i.e., indistinguishability) to a one-wayness-style property (i.e., unpredictability) in the so-called *Quantum Random Oracle Model* (QROM) [BDF+11].

Lemma 1 (One-Way to Hiding (OW2H) [Unr14])**.** *Let $H : \{0,1\}^n \to \{0,1\}^n$ be a quantum random oracle. Consider an oracle algorithm A^H that makes at most q queries to H. Let B^H be an oracle algorithm that on input x does the following: picks $j \leftarrow \{1, \ldots, q\}$ and $y \leftarrow \{0,1\}^n$, runs $A^H(x, y)$ until (just before) the j-th query, measures argument of the query in the computational basis and outputs the measurement outcome (if A makes less than j queries, B outputs $\perp \notin \{0,1\}^n$). Let,*

$$P^1_A = \Pr[1 \leftarrow A^H(x, H(x)) : x \leftarrow \{0,1\}^n]$$
$$P^2_A = \Pr[1 \leftarrow A^H(x, y) : x \leftarrow \{0,1\}^n,\ y \leftarrow \{0,1\}^n]$$
$$P_B = \Pr[x \leftarrow B^H(x) : x \leftarrow \{0,1\}^n]$$

Then, we have $|P^1_A - P^2_A| \leq 2q\sqrt{P_B}$.

Tweakable Block Ciphers. A *tweakable block cipher* (TBC) with key space $\{0,1\}^\kappa$, tweak space $\{0,1\}^t$ and block-size n is a map $\tilde{\mathsf{E}} : \{0,1\}^\kappa \times \{0,1\}^t \times \{0,1\}^n \to \{0,1\}^n$ such that for every key $\mathsf{K} \in \{0,1\}^\kappa$ and every tweak $\mathsf{T} \in \{0,1\}^t$, $\mathsf{M} \mapsto \tilde{\mathsf{E}}(\mathsf{K}, \mathsf{T}, \mathsf{M})$ is a permutation on $\{0,1\}^n$. We denote the map $(\mathsf{T}, \mathsf{M}) \mapsto \tilde{\mathsf{E}}(\mathsf{K}, \mathsf{T}, \mathsf{M})$ by $\tilde{\mathsf{E}}_\mathsf{K}$. A *tweakable permutation* with tweak space $\{0,1\}^t$ and block-size n is a map $\tilde{\Pi} : \{0,1\}^t \times \{0,1\}^n \mapsto \{0,1\}^n$ such that for every tweak $\mathsf{T} \in \{0,1\}^t$, $\mathsf{M} \mapsto \tilde{\Pi}(\mathsf{T}, \mathsf{M})$ is a permutation on $\{0,1\}^n$.

Definition 6 (Quantum-secure TBC)**.** A TBC $\tilde{\mathsf{E}}_{\mathsf{K}} : \{0,1\}^t \times \{0,1\}^n \to \{0,1\}^n$ is quantum-secure if no efficient quantum adversary $\mathcal{A}$ making quantum queries can distinguish between the map $\tilde{\mathsf{E}}_{\mathsf{K}}$ for a uniformly random secret key $\mathsf{K} \leftarrow \{0,1\}^\kappa$ and a tweakable permutation $\tilde{\Pi} : \{0,1\}^t \times \{0,1\}^n \mapsto \{0,1\}^n$ drawn uniformly at random from the set of all tweakable permutations with tweak space $\{0,1\}^t$ and block-size n.

Simon's Algorithm. Simon's algorithm [Sim94] allows one to efficiently solve the following problem:

"Given quantum access to a function $f : \{0,1\}^n \to \{0,1\}^n$ for which there exists s such that: $\forall x, y\ f(x) = f(y) \iff y \in \{x, x \oplus s\}$, find s."

Simon's algorithm solves the above problem with $O(n)$ (quantum) query complexity. For comparison, any algorithm with *classical* access to f requires $\Omega(\sqrt{2^n})$ queries. To go into some more details, Simon's algorithm recovers a vector $y \in \{0,1\}^n$ orthogonal to s, i.e., $y \cdot s = 0$, with a *single* quantum query. With $O(n)$ quantum queries, one obtains $n-1$ independent vectors $\langle y_i \rangle$ orthogonal to s w.h.p., and hence, s can be recovered by solving the corresponding system of linear equations.

Later it was shown in [KLLN16] that Simon's algorithm works as expected (i.e., recovers the hidden period s with $O(n)$ queries) even if the function f has a *few* "random collisions" x, y where $f(x) = f(y)$ but $y \notin \{x, x \oplus s\}$. To be specific, for Simon's algorithm to work properly, it is sufficient if the periodic function f does not have an "unwanted period" $t \neq s$ where $f(x \oplus t) = f(x)$ holds with probability $\geq \frac{1}{2}$ for a random choice of x. This condition suffices for cryptanalytic purposes, as shown in [KLLN16].

Deutsch's Algorithm. Deutsch's algorithm [Deu85] allows one to solve the following problem with probability 1:

"Given quantum access to a function $f : \{0,1\} \to \{0,1\}$, decide whether f is "constant" $(f(0) = f(1))$ or "balanced" $(f(0) \neq f(1))$."

Deutsch's algorithm solves the above problem with a *single* quantum query to f, whereas any algorithm with *classical* access to f requires two queries for the same success probability of 1. To be more specific, Deutsch's algorithm recovers the value $f(0) \oplus f(1)$ with a single quantum query to f, which solves the problem.

3 IND-qCPA Security Analysis of OCB1 and OCB3

3.1 Prior Quantum Superposition Attacks

By exploiting quantum access to the encryption oracle, a polynomial-time attack was proposed against the unforgeability of OCB1 and OCB3 in [KLLN16] which uses Simon's algorithm. Here, we give a high level recap of the attack for OCB1. We will use a similar strategy to attack the confidentiality of OCB1 and OCB3. For specific details of the attack, we refer to prior work.

The attack considers the following function $f_{\mathsf{N}} : \{0,1\}^n \to \{0,1\}^n$ such that $f_{\mathsf{N}}(\mathsf{M}) = \mathsf{c}[1] \oplus \mathsf{c}[2]$, where $(\mathsf{c}[1] \| \mathsf{c}[2] \| \mathsf{c}[3], \mathsf{t}) = \mathsf{OCB1.Enc}(\mathsf{K}, \mathsf{N}, \mathsf{M} \| \mathsf{M} \| 0^n)$. From Figure 1, we can see that $f_{\mathsf{N}}(\mathsf{M}) = \mathsf{E}_{\mathsf{K}}(\Delta_1 \oplus \mathsf{M}) \oplus \Delta_1 \oplus \mathsf{E}_{\mathsf{K}}(\Delta_2 \oplus \mathsf{M}) \oplus \Delta_2$, which is periodic with period $\Delta_1 \oplus \Delta_2$. It is argued in [KLLN16] that f_{N} satisfies the sufficient condition (i.e., non-existence of "unwanted periods") for Simon's algorithm to be applicable.

Now the function f_{N} can be computed in superposition with a *single*[6] quantum query to the encryption oracle $\mathsf{OCB1.Enc}(\mathsf{K}, \mathsf{N}, \cdot)$. Here we would use the techniques presented in [HS18] to "truncate out" the tag t from the output of $\mathsf{OCB1.Enc}(\mathsf{K}, \mathsf{N}, \cdot)$, as well as the techniques presented in [BBC+21] to compute a linear function of the ciphertext c

[6] This is important as the nonce N, and hence the function f_{N}, changes with each subsequent quantum query to the encryption oracle $\mathsf{OCB1.Enc}_{\mathsf{K}}(\mathsf{N}, \cdot)$.

Algorithm $\mathsf{OCB1.Enc}(\mathsf{K}, \mathsf{N}, \mathsf{M})$:

$L \leftarrow \mathsf{E}_{\mathsf{K}}(0^n)$;
$(\mathsf{M}[1], \ldots, \mathsf{M}[m]) \leftarrow \mathsf{M}$;
$\Delta_0 \leftarrow \mathsf{E}_{\mathsf{K}}(\mathsf{N} \oplus L)$;
for $i \leftarrow 1$ **to** m **do**
  $\Delta_i \leftarrow \gamma_i \cdot L \oplus \Delta_0$;
end
for $i \leftarrow 1$ **to** $m-1$ **do**
  $\mathsf{c}[i] \leftarrow \Delta_i \oplus \mathsf{E}_{\mathsf{K}}(\Delta_i \oplus \mathsf{M}[i])$;
end
$\mathsf{Pad} \leftarrow \mathsf{E}_{\mathsf{K}}(2^{-1}L \oplus \Delta_m \oplus \mathsf{len}(\mathsf{M}[m]))$;
$\mathsf{c}[m] \leftarrow \mathsf{M}[m] \oplus \mathsf{msb}_{\mathsf{len}(\mathsf{M}[m])}(\mathsf{Pad})$;
$\Sigma \leftarrow \mathsf{c}[m] \| 0^* \oplus \mathsf{Pad}$;
$\Sigma \leftarrow \mathsf{M}[1] \oplus \ldots \oplus \mathsf{M}[m-1] \oplus \Sigma$;
$\mathsf{t} \leftarrow \mathsf{E}_{\mathsf{K}}(\Delta_m \oplus \Sigma)$;
$\mathsf{t} \leftarrow \mathsf{msb}_{\tau}(\mathsf{t})$;
return (c, t)

Figure 1: OCB1 block-cipher mode. Here γ_i is the Gray encoding of integer i.

(i.e., $\mathsf{c}[1] \oplus \mathsf{c}[2]$) output by $\mathsf{OCB1.Enc}(\mathsf{K}, \mathsf{N}, \cdot)$ – all within a single quantum query. This allows us to apply Simon's algorithm on f_{N}. Namely, using a single quantum query to f_{N}, Simon's algorithm recovers a vector $y \in \{0,1\}^n$ orthogonal to the period $\Delta_1 \oplus \Delta_2$. Now note that $\Delta_1 \oplus \Delta_2 = (\gamma_1 \oplus \gamma_2) \cdot L$ is independent of the nonce N. Hence in the context of Simon's algorithm, even though the nonce N, and hence the function f_{N}, changes with each subsequent quantum query, we still obtain a random vector y orthogonal to the *fixed* period $\Delta_1 \oplus \Delta_2$. After obtaining $O(n)$ such independent orthogonal vectors, we can recover the value $\Delta_1 \oplus \Delta_2$. As shown in [KLLN16], this allows one to break the existential unforgeability of OCB1–with random nonces–in a quantum setting (more formally, the EUF-qCMA notion described in Definition 3).

3.2 IND-qCPA Insecurity of OCB1

We extend the above attack to show that OCB1 fails to confer *confidentiality* in a quantum setting as well; in other words, we break the formal notion of IND-qCPA security – with random nonces – of OCB1. In our attack, we exploit the fact that the $\mathsf{OCB1.Enc}(\mathsf{K}, \cdot, \cdot)$ algorithm processes the last block of messages differently (i.e., a "one-time-pad" type encryption is applied to $\mathsf{M}[m]$; see Fig. 1) compared to the rest of the message blocks. In the following, we sketch our attack in the IND-qCPA game w.r.t. OCB1:

1. Using $O(n)$ *quantum* encryption queries, recover the value $\Delta_1 \oplus \Delta_2$ via Simon's algorithm (similar to the EUF-qCMA attack in [KLLN16] discussed above).

2. Compute the value $L := (\Delta_1 \oplus \Delta_2) \cdot (\gamma_1 \oplus \gamma_2)^{-1}$.

3. For the challenge query, pick single-block messages $\mathsf{M}_0 = (n \oplus 2^{-1}L)$ and a random $\mathsf{M}_1 \neq \mathsf{M}_0$.

4. Upon receiving the response $(\mathsf{c}^*, \mathsf{t}^*)$ from the challenger, output bit $b' = 0$ if $\mathsf{msb}_{\tau}(\mathsf{c}^*) \oplus \mathsf{t}^* = \mathsf{msb}_{\tau}(\mathsf{M}_0)$, and output $b' = 1$ otherwise.

We now argue why the above attack succeeds with high probability. Looking at Figure 1

(Line 10), note that if M_0 was encrypted by the challenger, then

$$\mathsf{Pad} := \mathsf{E}_\mathsf{K}(2^{-1}L \oplus \Delta_1 \oplus n) = \mathsf{E}_\mathsf{K}(\mathsf{M}_0 \oplus \Delta_1), \tag{1}$$

and the checksum in Line 12 is defined as $\Sigma := \mathsf{c}^* \oplus \mathsf{Pad} = \mathsf{M}_0$. Further, for a single message block, Line 13 is ignored. Hence, we have (following Line 14,15)

$$\mathsf{t}^* := \mathsf{msb}_\tau(\mathsf{E}_\mathsf{K}(\Delta_1 \oplus \Sigma)) = \mathsf{msb}_\tau(\mathsf{E}_\mathsf{K}(\Delta_1 \oplus \mathsf{M}_0)) = \mathsf{msb}_\tau(\mathsf{Pad}), \tag{2}$$

which implies $\mathsf{t}^* := \mathsf{msb}_\tau(\mathsf{c}^* \oplus \mathsf{M}_0)$ – a condition which can be easily verified by the corresponding IND-qCPA adversary.

If M_1 was encrypted by the challenger, then we still have $\mathsf{Pad} := \mathsf{E}_\mathsf{K}(\mathsf{M}_0 \oplus \Delta_1)$ (Equation 1). But this time, we have the tag to be (Equation 2):

$$\mathsf{t}^* := \mathsf{msb}_\tau(\mathsf{E}_\mathsf{K}(\Delta_1 \oplus \Sigma)) = \mathsf{msb}_\tau(\mathsf{E}_\mathsf{K}(\Delta_1 \oplus \mathsf{M}_1)).$$

Hence, for our attack to incorrectly guess the bit $b' = 0$, we need to have "$\mathsf{msb}_\tau(\mathsf{c}^*) \oplus \mathsf{t}^* = \mathsf{msb}_\tau(\mathsf{M}_0)$", or equivalently, "$\mathsf{msb}_\tau(\mathsf{M}_1 \oplus \mathsf{M}_0) = \mathsf{msb}_\tau(\mathsf{E}_\mathsf{K}(\mathsf{M}_0 \oplus \Delta_1) \oplus \mathsf{E}_\mathsf{K}(\mathsf{M}_1 \oplus \Delta_1))$". This condition can be considered to hold with a negligible property for a sufficiently large tag length τ. Whereas, if $\tau = n$ and the condition "$\mathsf{M}_1 \oplus \mathsf{M}_0 = \mathsf{E}_\mathsf{K}(\mathsf{M}_0 \oplus \Delta_1) \oplus \mathsf{E}_\mathsf{K}(\mathsf{M}_1 \oplus \Delta_1)$" holds for a $\mathsf{M}_1 \neq \mathsf{M}_0$, it implies the existence of a *differential* in E_K. But as explained in [KLLN16], if E_K is assumed to behave as a secure PRP (which is required for the *classical* security of OCB1), then the existence of such a differential happens with a negligible probability.

There are a two aspects worth noting about our IND-qCPA attack against OCB1. First, the attack works even if the underlying block-cipher E_K is assumed to be a quantum-secure PRP. This is in contrast to the results in [ATTU16] which show that certain popular block-cipher modes of operation such as CBC, CFB, OFB and CTR are IND-qCPA secure when used with a quantum-secure PRP. Second, note that our attack is oblivious to the nonces chosen by the IND-qCPA challenger. I.e., the corresponding adversary need not have access to the nonces when making queries; hence, it can still win the IND-qCPA game regardless of the distribution of nonces.

3.3 IND-qCPA Insecurity of OCB3

We now describe an attack that breaks the formal IND-qCPA security – with random nonces – of OCB3. It uses similar ideas as our IND-qCPA attack against OCB1. The main difference is that the $\mathsf{OCB3.Enc}(\mathsf{K}, \cdot, \cdot)$ algorithm does not process the last block of messages differently (i.e., applying a "one-time-pad" type encryption) by default; it only does so if the last message block is a *partial* block, i.e., length less than n bits. Hence, in the "challenge query" phase of the IND-qCPA game, we work with messages that contain a full n-bit block and a partial block. We sketch our IND-qCPA attack against OCB3 below:

1. Using $O(n)$ *quantum* encryption queries, recover the value $\Delta_1 \oplus \Delta_2$ via Simon's algorithm (similar to the EUF-qCMA attack in [KLLN16] discussed above, but applied to OCB3).
2. Compute the value $L := (\Delta_1 \oplus \Delta_2) \cdot (4(\gamma_1 \oplus \gamma_2))^{-1}$.
3. Pick an arbitrary $\mathsf{m} \in \{0,1\}^n$ and $\mathsf{m}_0 \in \{0,1\}^{n-1}$ which satisfy the following

$$(4\gamma_1 + 1) \cdot L = (4\gamma_1 + 3) \cdot L \oplus \mathsf{m} \oplus \mathsf{m}_0 \| 1, \tag{3}$$

   or equivalently, $\mathsf{m} \oplus \mathsf{m}_0 \| 1 = 2L$.

Algorithm $\mathsf{OCB3.Enc}(\mathsf{K}, \mathsf{N}, \mathsf{A}, \mathsf{M})$:

1 $L \leftarrow \mathsf{E}_{\mathsf{K}}(0^n)$;
2 $\Delta_0 \leftarrow \mathcal{H}_{\mathsf{K}}(\mathsf{N})$;
3 $(\mathsf{M}[1], \ldots, \mathsf{M}[m]) \leftarrow \mathsf{M}$;
4 **for** $i \leftarrow 1$ **to** $m-1$ **do**
5 $\quad \Delta_i \leftarrow 4\gamma_i \cdot L \oplus \Delta_0$;
6 $\quad \mathsf{c}[i] \leftarrow \Delta_i \oplus \mathsf{E}_{\mathsf{K}}(\Delta_i \oplus \mathsf{M}[i])$;
7 **end**
8 $\delta_{\mathsf{t}} \leftarrow 0$;
9 **if** $\mathsf{len}(\mathsf{M}[m]) = n$ **then**
10 $\quad \Delta_m \leftarrow 4\gamma_m \cdot L \oplus \Delta_0$;
11 $\quad \mathsf{c}[m] \leftarrow \Delta_m \oplus \mathsf{E}_{\mathsf{K}}(\Delta_m \oplus \mathsf{M}[m])$;
12 $\quad \delta_{\mathsf{t}} \leftarrow 4\gamma_m + 2$;
13 **end**
14 **else**
15 $\quad \Delta_m \leftarrow (4\gamma_{m-1} + 1) \cdot L \oplus \Delta_0$;
16 $\quad \mathsf{Pad} \leftarrow \mathsf{E}_{\mathsf{K}}(\Delta_m)$;
17 $\quad \mathsf{c}[m] \leftarrow \mathsf{M}[m] \oplus \mathsf{msb}_{\mathsf{len}(\mathsf{M}[m])}(\mathsf{Pad})$;
18 $\quad \delta_{\mathsf{t}} \leftarrow 4\gamma_{m-1} + 3$;
19 **end**
20 $\Sigma \leftarrow \mathsf{M}[m] \| 10^*$;
21 $\Sigma \leftarrow \mathsf{M}[1] \oplus \ldots \oplus \mathsf{M}[m-1] \oplus \Sigma$;
22 $\mathsf{t} \leftarrow \mathsf{E}_{\mathsf{K}}(\Delta_0 \oplus \delta_{\mathsf{t}} \cdot L \oplus \Sigma)$;
23 **if** $\mathsf{A} \neq \epsilon$ **then**
24 $\quad \mathsf{t} \leftarrow \mathsf{t} \oplus \mathsf{HASH}_{\mathsf{E_K}}(\mathsf{A})$;
25 **end**
26 $\mathsf{t} \leftarrow \mathsf{msb}_\tau(\mathsf{t})$;
27 **return** (c, t)

Figure 2: OCB3 block-cipher mode. Here $\mathcal{H}_{\mathsf{K}}$ is a keyed hash function and $\mathsf{HASH}_{\mathsf{E_K}}$ is an algorithm that processes the associated data A for authentication. The details of $\mathcal{H}_{\mathsf{K}}$ and $\mathsf{HASH}_{\mathsf{E_K}}$ are not relevant for our attack and are therefore omitted.

4. Choose a random $\mathsf{m}_1 \in \{0,1\}^{n-1}$ such that $\mathsf{m}_1 \neq \mathsf{m}_0$.
5. For the challenge query, select $\mathsf{M}_0 = \mathsf{m} \| \mathsf{m}_0$, $\mathsf{M}_1 = \mathsf{m} \| \mathsf{m}_1$ and $\mathsf{A} = \epsilon$.
6. Upon receiving the response $(\mathsf{c}^*, \mathsf{t}^*)$ from the challenger, output bit $b' = 0$ if $\mathsf{msb}_\tau(\mathsf{c}^*[2]) \oplus \mathsf{msb}_{n-1}(\mathsf{t}^*) = \mathsf{msb}_\tau(\mathsf{m}_0)$, and output $b' = 1$ otherwise.[7]

We now argue why the above attack succeeds w.h.p. Looking at Figure 2, note that if M_0 was encrypted in the IND-qCPA game, then (according to Line 16 for $m = 2$) we have

$$\mathsf{Pad} := \mathsf{E}_{\mathsf{K}}((4\gamma_1 + 1) \cdot L \oplus \Delta_0)$$

and (according to Line 20) $\Sigma := \mathsf{m} \oplus \mathsf{m}_0 \| 1$. Hence, we have (Line 18,22,26)

$$\mathsf{t}^* := \mathsf{msb}_\tau(\mathsf{E}_{\mathsf{K}}(\Delta_0 \oplus (4\gamma_1+3) \cdot L \oplus \mathsf{m} \oplus \mathsf{m}_0 \| 1)) = \mathsf{msb}_\tau(\mathsf{E}_{\mathsf{K}}(\Delta_0 \oplus (4\gamma_1+1) \cdot L)) = \mathsf{msb}_\tau(\mathsf{Pad}),$$

where we used Equation 3 w.r.t. our choice of m and m_0. This implies $\mathsf{msb}_{n-1}(\mathsf{t}^*) = \mathsf{msb}_\tau(\mathsf{m}_0 \oplus \mathsf{c}^*[2])$, which can be easily verified by the corresponding IND-qCPA adversary.

[7] The convention being used here is that, given a bit-string $s \in \{0,1\}^\ell$, $\mathsf{msb}_k(s) = s$ for any $k \geq \ell$.

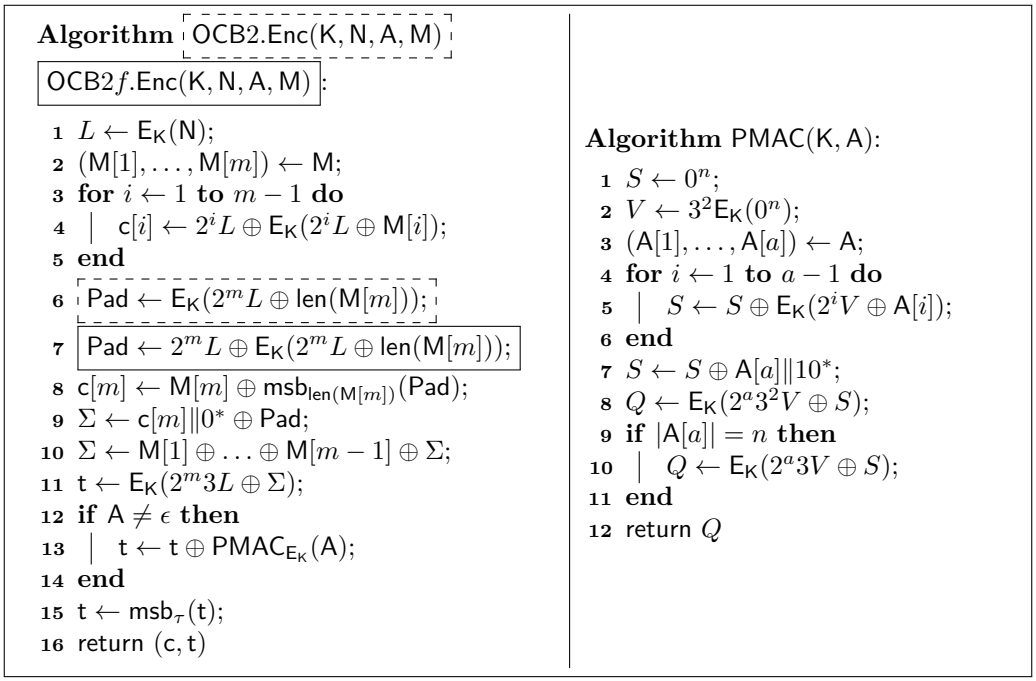

Figure 3: OCB2 and OCB2f block-cipher modes. The encryption algorithms of both schemes *only* differ in the way "Pad" values are defined: Line 6 (dashed box) in OCB2 and Line 7 (normal box) in OCB2f.

If M_1 was encrypted, then the tag is

$$\mathsf{t}^* := \mathsf{msb}_\tau(\mathsf{E_K}(\Delta_0 \oplus (4\gamma_1+3) \cdot L \oplus \mathsf{m} \oplus \mathsf{m}_1 \| 1)) = \mathsf{msb}_\tau(\mathsf{E_K}(\Delta_0 \oplus (4\gamma_1+1) \cdot L \oplus \mathsf{m}_0 \| 1 \oplus \mathsf{m}_1 \| 1)).$$

It's not hard to see that for our attack to incorrectly guess $b' = 0$, we need to have "$\mathsf{msb}_\tau(\mathsf{c}^*[2]) \oplus \mathsf{msb}_{n-1}(\mathsf{t}^*) = \mathsf{msb}_\tau(\mathsf{m}_0)$", or equivalently,

$$\begin{aligned} \text{“}\mathsf{msb}_\tau(\mathsf{m}_0 \oplus \mathsf{m}_1) = \mathsf{msb}_\tau(\mathsf{msb}_{n-1}(\mathsf{E_K}((4\gamma_1+1) \cdot L \oplus \Delta_0))) \\ \oplus \mathsf{msb}_\tau(\mathsf{msb}_{n-1}(\mathsf{E_K}((4\gamma_1+1) \cdot L \oplus \Delta_0 \oplus \mathsf{m}_0 \| 1 \oplus \mathsf{m}_1 \| 1)))\text{”} \end{aligned}$$

Using similar arguments as above w.r.t. our IND-qCPA attack against OCB1, we can consider this event to happen with a negligible probability for a random choice of m_1 and a sufficiently large τ; namely, we use that differentials do not exist in $\mathsf{E_K}$ except negligible probability when modeled as a PRP. Similar to our attack against OCB1, our IND-qCPA attack against OCB3: (1) works even if $\mathsf{E_K}$ is a quantum-secure PRP, and (2) does not require access to nonces used in the encryption algorithm.

4 IND-qCPA Security Analysis of OCB2(f)

In this section, we analyze the IND-qCPA security of OCB2 and OCB2f (an alternative to OCB2 that was proven in [IIMP19] to be a classically secure AE scheme) in various settings. For ease of exposition, the analysis is presented with respect to OCB2; however, we also explain how each of the results in this section apply analogously to OCB2f in a straightforward manner due to the fundamental structural similarities between OCB2 and OCB2f.

When considering the "pure" encryption routine of OCB2 in Figure 3, i.e. when AD is always kept empty, the value $L = \mathsf{E_K}(\mathsf{N})$ that determines the offsets changes with the

nonce. This is in contrast to OCB1 and OCB3 (see Figures 1 and 2 respectively) where $L = \mathsf{E}_\mathsf{K}(0^n)$ is fixed. This makes it difficult to extend our above IND-qCPA attacks against OCB1 and OCB3 to OCB2, since now it's not straightforward to use Simon's algorithm for recovering L.[8]

We show that this approach does not lead to a successful attack by actually proving the IND-qCPA security of OCB2, in a random nonce setting, when it is operated as a "pure" AE mode–with no input AD. Now it is well-known at this point that OCB2 is an insecure mode *classically* as shown in [IIMP19]. Namely, it was shown in [IIMP19] that OCB2 fails to offer confidentiality in an IND-CCA sense, even when operated as a pure AE mode. But at the same time, it is argued that there are no problems with OCB2 from an IND-CPA perspective. Hence, technically any positive IND-qCPA security result of OCB2 (with random nonces) when operated as a pure AE mode will not contradict the findings of [IIMP19].

Now OCB2 was shown (correctly) to be IND-CPA secure in [Rog04] where the corresponding security proof views the scheme as a TBC mode of operation. For example in Line 4 in Figure 3 in the $\mathsf{OCB2.Enc}(\mathsf{K}, \cdot)$ algorithm can be seen as a single invocation of an underlying TBC $\tilde{\mathsf{E}}_\mathsf{K}$ and would have the form "$\mathsf{c}[i] \leftarrow \tilde{\mathsf{E}}_\mathsf{K}((\mathsf{N}, i), \mathsf{M}[i])$". Here, the TBC $\tilde{\mathsf{E}}_\mathsf{K}$ uses the so-called "XEX" construction (as defined in [Rog04]) which generically turns a block-cipher (here E_K) into a TBC ($\tilde{\mathsf{E}}_\mathsf{K}$); specifically, $\tilde{\mathsf{E}}_\mathsf{K}$ is defined as $\tilde{\mathsf{E}}_\mathsf{K}((\mathsf{N}, i), \mathsf{M}) = 2^i L \oplus \mathsf{E}_\mathsf{K}(2^i L \oplus \mathsf{M})$ where $L = \mathsf{E}_\mathsf{K}(\mathsf{N})$. Similarly, Lines 6 and 11 of the $\mathsf{OCB2.Enc}$ algorithm in Figure 3 use the so-called "XE" construction for the corresponding TBCs (see [Rog04, IIMP19] for more details).

It is shown in [Rog04] that XE(X)-based TBCs are *classically* secure, according to the classical analogue of Definition 6. Hence, the IND-CPA security proof of OCB2 in [Rog04] first replaces the underlying TBC $\tilde{\mathsf{E}}_\mathsf{K}$ with a uniformly random tweakable permutation, and then argues that in this setting, the IND-CPA advantage of any adversary will be zero. However, there is a problem if we want to use the same strategy in trying to prove the IND-qCPA security of OCB2. Namely, it was shown in [KLLN16] that the XE(X) construction does not result in *quantum-secure* TBCs. Specifically, the paper describes an attack which allows one to distinguish an XE(X)-based TBC from a uniformly random tweakable permutation with $O(n)$ quantum queries, where the queries consist of classical tweaks and quantum message blocks. This distinguishing attack exploits that the tweaks are fixed across the $O(n)$ queries.

In OCB2, this is not the case, due to the design that the tweaks depend on the nonce. With every new encryption query, the nonce changes and therefore the tweak changes as well. Unfortunately, we extend the distinguishing attack of [KLLN16] to show that XE(X)-based TBCs used in OCB2 remain insecure in the quantum setting even when tweaks are not allowed to repeat across queries. We outline the attack in Section 4.1.

Hence, in order to prove the IND-qCPA security–with random nonces–of OCB2 when operated as a pure AE mode, we cannot rely on the quantum security of the underlying XE(X)-based TBCs. But fortunately, we were able to show the IND-qCPA security of OCB2 in the same setting by instead relying on the *quantum-security of the underlying block-cipher* E_K (similar to IND-qCPA security proofs of the CBC and CFB block-cipher modes of operation presented in [ATTU16]). We provide the theorem in Section 4.2.

We complete our quantum security analysis of OCB2 in Section 4.3 in which we show that OCB2 is *insecure* in the same setting (IND-qCPA with random nonces) when used as an AEAD scheme. In other words, we describe an attack that breaks the formal INDq-CPA

[8] A more advanced superposition attack is presented in [BBC+21, Subsection 3.2] that allows one to recover $L = \mathsf{E}_\mathsf{K}(\mathsf{N})$ w.r.t. OCB2 using Simon's algorithm. However, it's not clear how this can be used to attack the IND-qCPA security of OCB2–when AD is always kept empty–since the challenge queries are supposed to be *classical* in the security game. This, e.g., makes it difficult to apply the advanced superposition attack to recover $L^* = \mathsf{E}_\mathsf{K}(\mathsf{N}^*)$ where the random nonce N^* is used to encrypt the challenge query.

security – with random nonces – of OCB2 which exploits the way AD is authenticated by the encryption algorithm. This is in contrast to our IND-qCPA attacks against OCB1 and OCB3 which only targeted the pure encryption part of the corresponding schemes.

4.1 Extended Attack against the TBC Security of XE(X) in OCB2

In this section, we show that XE(X) is even quantum insecure when evaluted on each tweak at most a single time and the tweaks are the tweaks used in OCB2. In our attack, we first fix a random nonce N. We recap that $\tilde{\mathsf{E}}_\mathsf{K}$ is defined as $\tilde{\mathsf{E}}_\mathsf{K}((\mathsf{N}, i), \mathsf{M}) = 2^i L \oplus \mathsf{E}_\mathsf{K}(2^i L \oplus \mathsf{M})$ where $L = \mathsf{E}_\mathsf{K}(\mathsf{N})$. We define the function $f^{(i)} : \{0,1\}^n \to \{0,1\}^n$ such that

$$f^{(i)}(x) = \tilde{\mathsf{E}}_\mathsf{K}((\mathsf{N}, 2i-1), x) \oplus \tilde{\mathsf{E}}_\mathsf{K}((\mathsf{N}, 2i), x) = 2^{2i-1}L \oplus \mathsf{E}_\mathsf{K}(2^{2i-1}L \oplus x) \oplus 2^{2i}L \oplus \mathsf{E}_\mathsf{K}(2^{2i}L \oplus x).$$

Given quantum access to an oracle for the TBC $\tilde{\mathsf{E}}_\mathsf{K}$, we can build a quantum circuit implementing $f^{(i)}$ similar to the one described in [KLLN16, Fig. 8] w.r.t. the initial distinguishing attack. Also note that $f^{(i)}$ is periodic with period $(2^{2i-1}L \oplus 2^{2i}L)$. Following [KLLN16], we have $f^{(i)}$ satisfying the sufficient condition (i.e., non-existence of "unwanted periods") for Simon's algorithm to be applicable, based on the assumption that E_K is a secure PRP. Hence, by applying Simon's algorithm on $f^{(i)}$ with a single quantum query, we obtain a vector $y_i \in \{0,1\}^n$ orthogonal to $(2^{2i-1}L \oplus 2^{2i}L)$.

We emphasize that we're not allowed to make subsequent quantum queries to $f^{(i)}$ as a part of Simon's algorithm because of the restriction that the tweaks cannot repeat across our calls to $\tilde{\mathsf{E}}_\mathsf{K}$. Instead, we apply a *single* iteration of Simon's algorithm w.r.t. each of the n functions $\langle f^{(i)} \rangle_{1 \le i \le n}$ to obtain n independent vectors $\langle y_i \rangle_{1 \le i \le n}$, where y_i is orthogonal to $(2^{2i-1}L \oplus 2^{2i}L)$ for $1 \le i \le n$.

By solving the corresponding system of n linear equations, we can recover the value $L = \mathsf{E}_\mathsf{K}(\mathsf{N})$ w.h.p. (this is quite similar to the advanced superposition attack presented in [BBC+21, Subsection 3.2]). Upon recovering L, we choose two classical n-bit values x_1, x_2 such that $2^{2n+1}L \oplus x_1 = 2^{2n+2}L \oplus x_2$. We then make two final queries to the $\tilde{\mathsf{E}}_\mathsf{K}$ oracle, namely $z_1 \leftarrow \tilde{\mathsf{E}}_\mathsf{K}((N, 2n+1), x_1)$ and $z_2 \leftarrow \tilde{\mathsf{E}}_\mathsf{K}((N, 2n+2), x_2)$, and check if the following is satisfied: "$z_1 \oplus z_2 = 2^{2n+1}L \oplus 2^{2n+2}L$". If we have been interacting with the "real" TBC oracle for $\tilde{\mathsf{E}}_\mathsf{K}$ above, then it's not hard to see that this condition is satisfied w.h.p. However, if we instead were interacting with a quantum oracle for a uniformly random tweakable permutation $\tilde{\Pi}$, then this condition holds with a negligible probability. This results in a non-negligible distinguishing advantage against the TBC security. Further, the attack aligns with the design of OCB2 such that we cannot rely on a weaker TBC security notion of XE(X) when attempting to prove the security of OCB2.

4.2 IND-qCPA Security of OCB2 without Associated Data

In this section, we prove the IND-qCPA security of OCB2 when not used with associated data, denoted as OCB2$^-$, based on the quantum security of the underlying block cipher. Formally, we prove the following:

Theorem 1. *Let $\mathcal{A}$ be any IND-qCPA adversary against OCB2$^-$ – in a random nonce setting – making Q encryption and challenge queries in total, with m being the maximum length (in blocks) of messages in each query. Let $q = Q(m+2)$. Then there exists a quantum adversary $\mathcal{B}$ which distinguishes the underlying block-cipher E_K ($\mathsf{K} \leftarrow \{0,1\}^\kappa$) from a truly random function $H : \{0,1\}^n \to \{0,1\}^n$ while making (at most) q quantum queries, and running in about the same time as that of $\mathcal{A}$, such that*

$$\mathsf{Adv}^{\mathsf{IND\text{-}qCPA}}_{\mathsf{OCB2}^-}(\mathcal{A}) \le \mathsf{Adv}^{\mathsf{qPRF}}_{\mathsf{E}_\mathsf{K}}(\mathcal{B}) + Q(1+m)\frac{2q}{2^{n/2}} + 2Qq\sqrt{2^{-n} + \frac{2q}{2^{n/2}}}.$$

Proof. During the execution of the IND-qCPA game w.r.t. adversary $\mathcal{A}$, let q be the (maximum) number of times the block-cipher $\mathsf{E_K}$ is invoked, possibly in superposition. Note that we have $q = Q(m+2)$ since the OCB2.Enc$(\cdot)$ algorithm (see Fig. 3) involves $(m+2)$ invocations of the underlying block-cipher $\mathsf{E_K}$ when encrypting a message of length m (and empty AD). We proceed by defining a sequence of hybrid games, where we bound the difference in $\mathcal{A}$'s winning probabilities in consecutive hybrids. The convention here is that any change introduced in a particular hybrid remains in the subsequent hybrids. For the sake of simplicity, we assume that $\mathcal{A}$ asks a single challenge query. Later we show how to extend our reasoning that follows to the case of multiple challenge queries. Let m^* $(\leq m)$ be the length of messages M_0 and M_1 chosen by $\mathcal{A}$ in its challenge query.

$\mathsf{G_0}$: Is the usual IND-qCPA game w.r.t. OCB2$^-$ with random nonces.

$\mathsf{G_1}$: We replace $\mathsf{E_K}$ in the OCB2.Enc algorithm with a truly random function $H : \{0,1\}^n \to \{0,1\}^n$, when responding to $\mathcal{A}$'s queries.

$\mathsf{G}_{(2,i,0)}, i \in [Q]$: We replace Line 1 of the OCB2.Enc algorithm with "$L \leftarrow \{0,1\}^n$" when responding to the first $(Q-i+1)$ queries of $\mathcal{A}$ (which could either be an encryption query or a challenge query).

$\mathsf{G}_{(2,i,1)}, i \in [Q]$: We replace Line 6 of OCB2.Enc with "$\mathsf{Pad} \leftarrow \{0,1\}^n$" when responding to the first $(Q-i+1)$ queries of $\mathcal{A}$.

$\mathsf{G}_{(3,i)}, i \in [m^*]$: We replace the first i blocks of the challenge ciphertext c^* with "$\mathsf{c}^*[i] \leftarrow \{0,1\}^n$" (Line 4).

$\mathsf{G_4}$: We replace the tag t^* with "$\mathsf{t}^* \leftarrow \{0,1\}^n$" (Line 11 of OCB2.Enc).

In G_4, we have the values $(\mathsf{c}^*, \mathsf{t}^*)$ obtained by $\mathcal{A}$ as a response to its challenge query to be independent of the bit b. That is, $\mathcal{A}$'s view in G_4 is independent of b, and hence, its winning probability is exactly $1/2$. Below we bound the difference in $\mathcal{A}$'s winning probabilities in consecutive hybrids such that we have a corresponding upper-bound on $\mathcal{A}$'s final advantage in the IND-qCPA game G_0 as described in the theorem statement above. By "consecutive" hybrids, we mean that our proof considers the following sequence of hybrids: $\mathsf{G}_0 \to \mathsf{G}_1 \to \mathsf{G}_{(2,1,0)} \to \mathsf{G}_{(2,1,1)} \to \mathsf{G}_{(2,2,0)} \to \mathsf{G}_{(2,2,1)} \to \ldots \to \mathsf{G}_{(2,Q,0)} \to \mathsf{G}_{(2,Q,1)} \to \mathsf{G}_{(3,1)} \to \mathsf{G}_{(3,2)} \to \ldots \to \mathsf{G}_{(3,m^*)} \to \mathsf{G}_4$. It then follows that any AE scheme constructed via the OCB2 mode with an underlying quantum-secure block-cipher $\mathsf{E_K}$ is IND-qCPA secure with random nonces, when considering polynomial-time quantum adversaries $\mathcal{A}$.

Let "$\mathsf{G}_k = 1$" denote the event of $\mathcal{A}$ winning the game G_k. Also, let the 'L', 'N' and 'Pad' values used in the OCB2.Enc algorithm to respond to $\mathcal{A}$'s i-th query in any hybrid game be represented as L_i, N_i and Pad_i respectively. We now proceed to prove the following lemmas.

Lemma 2. *We have* $|\Pr[\mathsf{G}_0 = 1] - \Pr[\mathsf{G}_1 = 1]| \leq \mathsf{Adv}^{\mathsf{qPRF}}_{\mathsf{E_K}}(\mathcal{B})$ *for a quantum distinguisher* $\mathcal{B}$ *which makes (at most)* q *quantum queries and runs in about the same time as that of* $\mathcal{A}$.

Proof. The proof follows via a straightforward reduction w.r.t the quantum PRF security of $\mathsf{E_K}$. □

Before showing how to transition from G_1 to $\mathsf{G}_{(2,1,0)}$ (Lemma 4), we show how to transition from $\mathsf{G}_{(2,i,0)}$ to $\mathsf{G}_{(2,i,1)}$. Lemma 4 is slightly more complex than Lemma 3, which serves us as a warm up.

Lemma 3. *We have* $|\Pr[\mathsf{G}_{(2,i,1)} = 1] - \Pr[\mathsf{G}_{(2,i,0)} = 1]| \leq \dfrac{2q}{2^{n/2}}$, *for* $i \in [Q]$.

Proof. Let m be the length of messages M (possibly) in superposition in $\mathcal{A}$'s $(Q-i+1)$-th query. Note that the only difference between hybrids $\mathsf{G}_{(2,i,0)}$ and $\mathsf{G}_{(2,i,1)}$ is the way Pad_{Q-i+1} is derived; in the former, we have $\mathsf{Pad}_{Q-i+1} \leftarrow H(2^m L_{Q-i+1} \oplus \mathsf{len}(\mathsf{M}[m]))$, and in the latter, $\mathsf{Pad}_{Q-i+1} \leftarrow \{0,1\}^n$. In the context of Lemma 1 w.r.t. the (internal) quantum random oracle H, let $x \leftarrow \{0,1\}^n$ and $y \leftarrow \{0,1\}^n$, and consider an oracle algorithm A^H making at most q queries to H such that $A^H(x, H(x))$ simulates the hybrid $\mathsf{G}_{(2,i,0)}$ towards $\mathcal{A}$ and $A^H(x,y)$ simulates $\mathsf{G}_{(2,i,1)}$. The only thing worth noting here is the way A^H handles $\mathcal{A}$'s $(Q-i+1)$-th query. A^H first generates a random nonce $\mathsf{N}_{Q-i+1} \leftarrow \{0,1\}^n$ and forwards it to $\mathcal{A}$. Upon receiving a message M (possibly) in superposition, or a pair of classical messages $(\mathsf{M}_0, \mathsf{M}_1)$, A^H sets the value $L_{Q-i+1} := 2^{-m}(x \oplus \mathsf{len}(\mathsf{M}[m]))$ (here $\mathsf{len}(\mathsf{M}[m])$ is a classical value because messages in superposition in $\mathcal{A}$'s queries are of the same length) and sets Pad_{Q-i+1} to be its second input (which is either $H(x)$ or y). Note that L_{Q-i+1} is a uniformly random value in the hybrids $\mathsf{G}_{(2,i,0)}$ and $\mathsf{G}_{(2,i,1)}$ which is independent of $\mathcal{A}$'s $(Q-i+1)$-th query. The value L_{Q-i+1} generated by A^H also has an identical distribution as in the hybrids because L_{Q-i+1} is obtained via a "one-time-pad"-like encryption with the independent and uniformly random value of x.

Again in the context of Lemma 1, it's not hard to see that $\Pr[\mathsf{G}_{(2,i,0)} = 1] = P_A^1$ and $\Pr[\mathsf{G}_{(2,i,1)} = 1] = P_A^2$. Since we have $|P_A^1 - P_A^2| \leq 2q\sqrt{P_B}$, we now show that the probability P_B is at most 2^{-n}, thereby completing our proof. Note that the corresponding oracle algorithm B^H defined by Lemma 1 essentially simulates the hybrid $\mathsf{G}_{(2,i,1)}$ towards $\mathcal{A}$ like $A^H(x,y)$ does, but instead of leading the game to its completion, B^H stops the execution of $\mathsf{G}_{(2,i,1)}$ at a point when the oracle H is invoked for the j-th time – for a random $j \in [q]$ picked by B^H beforehand – and measures the argument of the j-th H-query to see if it obtains x. We now consider three cases depending on whether the j-th H-query happens before, during, or after $\mathcal{A}$'s $(Q-i+1)$-th query.

- Before $\mathcal{A}$'s $(Q-i+1)$-th query: Note that $\mathcal{A}$'s view before it makes the $(Q-i+1)$-th query is independent of x, because x hasn't been introduced in the game $\mathsf{G}_{(2,i,1)}$'s execution yet. Hence, as $x \leftarrow \{0,1\}^n$ is an independent and uniformly random value, the probability P_B conditioned on the j-th H-query falling in this category is (at most) 2^{-n}, a negligible quantity.

- During $\mathcal{A}$'s $(Q-i+1)$-th query: Here, the j-th H-query can either correspond to Line 4 ("$\mathsf{c}[i] \leftarrow 2^i L \oplus H(2^i L \oplus \mathsf{M}[i])$"), or Line 11 ("$\mathsf{t} \leftarrow H(2^m 3L \oplus \Sigma)$"). Note that we're not taking into account the H-calls associated to computing $\mathsf{PMAC}_{\mathsf{E_K}}(\mathsf{A})$ in Line 13 ("$\mathsf{t} \leftarrow \mathsf{t} \oplus \mathsf{PMAC}_{\mathsf{E_K}}(\mathsf{A})$") because we're currently analyzing an $\mathsf{OCB2}$ mode which does not process associated data A; also we're ignoring Lines 1 and 6 because $L \leftarrow \{0,1\}^n$ and $\mathsf{Pad} \leftarrow \{0,1\}^n$ are derived uniformly at random in $\mathsf{G}_{(2,i,1)}$. When analyzing the probability P_B corresponding to each line, it helps to focus only on the quantum sub-circuit evaluating that particular line.

  If we consider the quantum sub-circuit evaluating Line 4, as described in Figure 4, then we note that measurement of the H-query (in the computational basis) commutes with the unitary gate $U_{2^i L}$, where the gate $U_{2^i L}$ acting on n qubits is defined as $U_{2^i L}|\mathsf{X}\rangle = |\mathsf{X} \oplus 2^i L\rangle$. Hence, the conditional probability P_B corresponding to Line 4 remains the same if we first measure the quantum state $|\mathsf{M}[i]\rangle$, instead of measuring the subsequent H-query $|2^i L \oplus \mathsf{M}[i]\rangle$. Now note that the $(Q-i+1)$-th query of $\mathcal{A}$ might depend on the nonce N_{Q-i+1}. However in the hybrid $\mathsf{G}_{(2,i,1)}$, the value L_{Q-i+1} is independent of N_{Q-i+1}. Hence, the view of $\mathcal{A}$ when it's making the $(Q-i+1)$-th query is also independent of L_{Q-i+1}. As a result, the probability distribution corresponding to the measurement outcome of $|\mathsf{M}[i]\rangle$ is independent of L_{Q-i+1}, and w.r.t. B^H's simulation, independent of x. So in a way, we are in a situation similar to the above case, where it is not hard to verify that the conditional probability P_B in this case is also bounded by 2^{-n}. Namely, if $|\mathsf{X}\rangle_{\mathrm{m}}$ denotes the

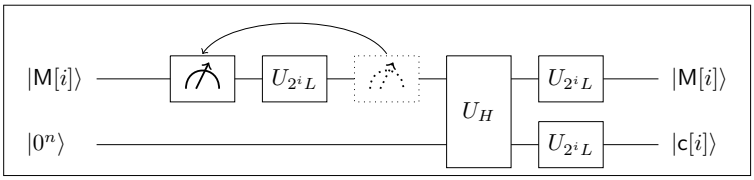

Figure 4: The quantum sub-circuit implementing Line 4 in the $\mathsf{OCB2.Enc}(\cdot)$ oracle in hybrid $\mathsf{G}_{(2,i,1)}$, namely "$\mathsf{c}[i] \leftarrow 2^iL \oplus H(2^iL \oplus \mathsf{M}[i])$" (see Fig. 3 for reference). Here, the unitary gate U_H models (internal) access to the random function $H : \{0,1\}^n \to \{0,1\}^n$ as $U_H |\mathsf{X}\rangle |\mathsf{Y}\rangle = |\mathsf{X}\rangle |\mathsf{Y} \oplus H(\mathsf{X})\rangle$, for arbitrary n-bit strings X, Y. Similarly, the unitary gate U_{2^iL} is defined as $U_{2^iL} |\mathsf{X}\rangle = |\mathsf{X} \oplus 2^iL\rangle$, for an arbitrary n-bit string X. This figure depicts the H-query measurement, which is a part of applying Lemma 1, commuting with the unitary U_{2^iL} – without affecting the probability P_B.

(classical) outcome of measuring quantum state $|\mathsf{X}\rangle$, then for $x \leftarrow \{0,1\}^n$, we have

$$\Pr[\,|2^iL_{Q-i+1} \oplus \mathsf{M}[i]\rangle_{\mathrm{m}} = x] = \Pr[2^iL_{Q-i+1} \oplus |\mathsf{M}[i]\rangle_{\mathrm{m}} = x] = \Pr[x = (2^{i-m} \oplus 1)^{-1}(\,|\mathsf{M}[i]\rangle_{\mathrm{m}} \oplus 2^{i-m}\mathsf{len}(\mathsf{M}[m])\,)] \leq 2^{-n}.$$

Coming to Line 10, we apply a similar reasoning to show that the corresponding probability P_B is at most 2^{-n}. Namely, we again note that the measurement of H-query commutes with the unitary gate U_{2^m3L}. So we instead measure the quantum state corresponding to the checksum, namely $|\Sigma\rangle$, when evaluating P_B. Here we observe that the measurement outcome of $|\Sigma\rangle$ is independent of L_{Q-i+1} because the quantum sub-circuit evaluating $|\Sigma\rangle$ does not involve the value L_{Q-i+1} at all (see Lines 8, 9 and 10 in Figure 3, and note that $\mathsf{Pad}_{Q-i+1} \leftarrow \{0,1\}^n$ is a random value in $\mathsf{G}_{(2,i,1)}$). Therefore, we again have the conditional probability P_B to be bounded by 2^{-n}.

- After $\mathcal{A}$'s $(Q-i+1)$-th query: Let the j-th H-query happen during $\mathcal{A}$'s k-th query, for any $k > (Q-i+1)$. Since $\mathcal{A}$ now has the response w.r.t. its $(Q-i+1)$-th query, its view is no longer independent of x. However, note that $\mathcal{A}$'s view when making the k-th query is still independent of the value $L_k \leftarrow \{0,1\}^n$ that is sampled in $\mathsf{G}_{(2,i,1)}$. Hence to bound the probability P_B in this case, we use a similar reasoning as the previous "During" case, where instead of using the uniformity of x, we use the uniformity of L_k. This completes the proof of the lemma.

□

Lemma 4. *We have*

- $|\Pr[\mathsf{G}_{(2,1,0)} = 1] - \Pr[\mathsf{G}_1 = 1]| \leq 2q\sqrt{2^{-n} + \frac{2q}{2^{n/2}}}$, *and*
- $|\Pr[\mathsf{G}_{(2,i,0)} = 1] - \Pr[\mathsf{G}_{(2,i-1,1)} = 1]| \leq 2q\sqrt{2^{-n} + \frac{2q}{2^{n/2}}}$, *for* $i \in [Q] \setminus \{1\}$.

Proof. In the following, we focus on the pair of hybrids $(\mathsf{G}_{(2,i-1,1)}, \mathsf{G}_{(2,i,0)})$ for $i \in [Q] \setminus \{1\}$. However, the proof also applies to the pair $(\mathsf{G}_1, \mathsf{G}_{(2,1,0)})$ in an identical fashion. Note that the only difference between hybrids $\mathsf{G}_{(2,i-1,1)}$ and $\mathsf{G}_{(2,i,0)}$ is the way L_{Q-i+1} is derived; in the former, we have $L_{Q-i+1} \leftarrow H(\mathsf{N}_{Q-i+1})$, and in the latter, $L_{Q-i+1} \leftarrow \{0,1\}^n$. In the context of Lemma 1 w.r.t. the (internal) quantum random oracle H, let $x \leftarrow \{0,1\}^n$ and $y \leftarrow \{0,1\}^n$, and consider an oracle algorithm A^H making at most q queries to H such that $A^H(x, H(x))$ simulates the hybrid $\mathsf{G}_{(2,i-1,1)}$ towards $\mathcal{A}$ and $A^H(x,y)$ simulates $\mathsf{G}_{(2,i,0)}$.

The only thing worth noting here is that, when responding to $\mathcal{A}$'s $(Q-i+1)$-th query, A^H sets its first input x to be the random nonce N_{Q-i+1} and its second input (either $H(x)$ or y) to be the value L_{Q-i+1}. Again, it's not hard to see that $\Pr[\mathsf{G}_{(2,i-1,1)} = 1] = P_A^1$ and $\Pr[\mathsf{G}_{(2,i,0)} = 1] = P_A^2$. Since we have $|P_A^1 - P_A^2| \leq 2q\sqrt{P_B}$, we establish an appropriate upper bound on the probability P_B thereby completing our proof.

Note that the corresponding oracle algorithm B^H defined by Lemma 1 essentially simulates the hybrid $\mathsf{G}_{(2,i,0)}$ towards $\mathcal{A}$ like $A^H(x,y)$ does, but instead of leading the game to its completion, B^H stops the execution of $\mathsf{G}_{(2,i,0)}$ at a point when the oracle H is invoked for the j-th time–for a random $j \leftarrow \{1,\ldots,q\}$ picked by B^H beforehand–and measures the argument of the j-th H-query to see if it obtains x. If the j-th H-query happens before or after $\mathcal{A}$'s $(Q-i+1)$-th query, then to bound the corresponding probability P_B, we use a similar reasoning as in the "Before" and "After" cases respectively in the proof of Lemma 3. But if the j-th H-query happens during $\mathcal{A}$'s $(Q-i+1)$-th query, there is a problem with extending the reasoning seen in the "During" case of the above proof. If we consider Line 11 ("$\mathsf{t} \leftarrow H(2^m 3L \oplus \Sigma)$") in particular, it need not be the case that the measurement outcome of Σ is independent of L_{Q-i+1}, because the quantum sub-circuit evaluating Σ this time *does* involve L_{Q-i+1} for *partial blocks* $\mathsf{M}[m]$ – via the value $\mathsf{Pad}_{Q-i+1} \leftarrow \mathsf{E}_{\mathsf{K}}(2^m L_{Q-i+1} \oplus \mathsf{len}(\mathsf{M}[m]))$ in Line 6 of $\mathsf{OCB2.Enc}$. We didn't have this issue in the proof of Lemma 3 because $\mathsf{Pad}_{Q-i+1} \leftarrow \{0,1\}^n$ is a random value in $\mathsf{G}_{(2,i,1)}$. However, we overcome this issue via a "nested" application of Lemma 1 as follows.

We first define an additional sequence of *sub-games*. Corresponding to each hybrid $\mathsf{G}_{(2,i,0)} (i \in [Q])$, we define a pair of sub-games $\mathsf{H}_{(2,i,0)}$ and $\mathsf{H}_{(2,i,1)}$. The only difference between $\mathsf{G}_{(2,i,0)}$ and $\mathsf{H}_{(2,i,0)}$ is their respective winning conditions. Each $\mathsf{G}_{(2,i,0)}$ in essence is an IND-qCPA game where the adversary $\mathcal{A}$ wins if it outputs the bit b, or in other words, if $\mathcal{A}$ can tell which of M_0 or M_1 was encrypted by the challenger w.r.t. its challenge query. Whereas in game $\mathsf{H}_{(2,i,0)}$, the challenger first picks a uniformly random $j \in [q]$ and the uniformly random nonce $\mathsf{N}_{Q-i+1} \leftarrow \{0,1\}^n$. It then perfectly simulates the game $\mathsf{G}_{(2,i,0)}$ towards the adversary $\mathcal{A}$ until the point when oracle H is invoked for the j-th time when processing $\mathcal{A}$'s queries; $\mathcal{A}$ now wins the game $\mathsf{H}_{(2,i,0)}$ if measuring the argument of the j-th H-query results in N_{Q-i+1}. And the only difference between $\mathsf{H}_{(2,i,0)}$ and $\mathsf{H}_{(2,i,1)}$ is that in the latter, we have $\mathsf{Pad}_{Q-i+1} \leftarrow \{0,1\}^n$ (whereas in the former, we have $\mathsf{Pad}_{Q-i+1} \leftarrow \mathsf{E}_{\mathsf{K}}(2^m L_{Q-i+1} \oplus \mathsf{len}(\mathsf{M}[m]))$).

Let "$\mathsf{H}_k = 1$" denote the event of $\mathcal{A}$ winning the game H_k. Then we have the probability P_B described above to be $P_B = \Pr[\mathsf{H}_{(2,i,0)} = 1]$. We now show the following.

Claim. *We have* $\Pr[\mathsf{H}_{(2,i,0)} = 1] \leq 2^{-n} + \dfrac{2q}{2^{n/2}}$, *for* $i \in [Q]$.

Proof. In the context of Lemma 1 w.r.t. the oracle H, let $\tilde{x} \leftarrow \{0,1\}^n$ and $\tilde{y} \leftarrow \{0,1\}^n$, and consider an oracle algorithm C^H making at most q queries to H such that $C^H(\tilde{x}, H(\tilde{x}))$ simulates the sub-game $\mathsf{H}_{(2,i,0)}$ towards $\mathcal{A}$ and $C^H(\tilde{x},\tilde{y})$ simulates $\mathsf{H}_{(2,i,1)}$. Here, like in the proof of Lemma 3, C^H sets the value $L_{Q-i+1} := 2^{-m}(\tilde{x} \oplus \mathsf{len}(\mathsf{M}[m]))$ and sets Pad_{Q-i+1} to be its second input (either $H(\tilde{x})$ or $\tilde{y}$); it's not hard to see that L_{Q-i+1} is an independent and uniformly random value w.r.t. $\mathcal{A}$'s $(Q-i+1)$-th query, as is the case in sub-games $\mathsf{H}_{(2,i,0)}$ and $\mathsf{H}_{(2,i,1)}$. Again in the context of Lemma 1, we have $\Pr[\mathsf{H}_{(2,i,0)} = 1] = P_C^1$, $\Pr[\mathsf{H}_{(2,i,1)} = 1] = P_C^2$, and $|P_C^1 - P_C^2| \leq 2q\sqrt{P_D}$ where D^H is the corresponding oracle algorithm defined by Lemma 1 w.r.t. C^H. Hence, towards proving the above claim, it's sufficient to show that the probabilities $\Pr[\mathsf{H}_{(2,i,1)} = 1]$ and P_D are at most 2^{-n}.

Coming to $\Pr[\mathsf{H}_{(2,i,1)} = 1]$, we can now show that the probability is at most 2^{-n} by applying a similar reasoning as in the proof of Lemma 3; here we no longer have the issue with extending arguments from the "During" case of Lemma 3's proof–as we discussed initially when analyzing P_B above–because now $\mathsf{Pad}_{Q-i+1} \leftarrow \{0,1\}^n$ is independent of the value L_{Q-i+1}. We can also show that P_D is at most 2^{-n} in a quite similar fashion to that

of our analysis of "P_B" in the proof of Lemma 3. The only difference worth noting here is that the oracle algorithm "B^H" in Lemma 3's proof stops the simulation of $\mathsf{G}_{(2,i,1)}$ at a random j-th H-query from $\mathcal{A}$, whereas D^H stops the simulation of $\mathsf{H}_{(2,i,1)}$ at a random j-th H-query. But this does not affect the distribution of H-queries from $\mathcal{A}$ in both cases, because note that the values used in the $\mathsf{OCB2.Enc}$ algorithm to process $\mathcal{A}$'s queries (e.g., the 'L', 'N', 'Pad' values) are defined exactly the same in both $\mathsf{G}_{(2,i,1)}$ and $\mathsf{H}_{(2,i,1)}$ (the only difference between the two games are their respective winning conditions). This completes the proof of the above claim. □

After establishing the upper bound on probability P_B ($= \Pr[\mathsf{H}_{(2,i,0)} = 1]$) above, we now have

$$\begin{aligned} |\Pr[\mathsf{G}_{(2,i,0)} = 1] - \Pr[\mathsf{G}_{(2,i-1,1)} = 1]| &= |P_A^1 - P_A^2| \\ &\leq 2q\sqrt{P_B} \leq 2q\sqrt{2^{-n} + \frac{2q}{2^{n/2}}}, \end{aligned}$$

which completes the proof of Lemma 4. □

Lemma 5. *We have*

- $|\Pr[\mathsf{G}_{(3,1)} = 1] - \Pr[\mathsf{G}_{(2,Q,1)} = 1]| \leq \dfrac{2q}{2^{n/2}}$,
- $|\Pr[\mathsf{G}_{(3,i)} = 1] - \Pr[\mathsf{G}_{(3,i-1)} = 1]| \leq \dfrac{2q}{2^{n/2}}$, *for* $i \in [m^* - 1] \setminus \{1\}$,
- $\Pr[\mathsf{G}_{(3,m^*)} = 1] = \Pr[\mathsf{G}_{(3,m^*-1)} = 1]$, *and*
- $|\Pr[\mathsf{G}_4 = 1] - \Pr[\mathsf{G}_{(3,m^*)} = 1]| \leq \dfrac{2q}{2^{n/2}}$.

Proof. The only difference between hybrids $\mathsf{G}_{(3,i-1)}$ and $\mathsf{G}_{(3,i)}$, for $2 \leq i \leq m^* - 1$, is the way $\mathsf{c}^*[i]$ (the i-th block of ciphertext c^*) is derived w.r.t. $\mathcal{A}$'s challenge query; in the former, we have in Line 4 $\mathsf{c}^*[i] \leftarrow 2^i L \oplus H(2^i L \oplus \mathsf{M}_b[i])$, and in the latter, $\mathsf{c}^*[i] \leftarrow \{0,1\}^n$. In the context of Lemma 1 w.r.t. the random function H, let $x \leftarrow \{0,1\}^n$ and $y \leftarrow \{0,1\}^n$, and consider an oracle algorithm A^H making at most q queries to H such that $A^H(x, H(x))$ simulates the hybrid $\mathsf{G}_{(3,i-1)}$ towards $\mathcal{A}$ and $A^H(x, y)$ simulates $\mathsf{G}_{(3,i)}$. The only thing worth noting here is the way A^H handles $\mathcal{A}$'s challenge query. A^H first generates a random nonce $\mathsf{N} \leftarrow \{0,1\}^n$ and forwards it to $\mathcal{A}$. Upon receiving a pair of classical messages $(\mathsf{M}_0, \mathsf{M}_1)$, A^H sets the value $L := 2^{-i}(x \oplus \mathsf{M}_b[i])$ (here the bit $b \leftarrow \{0,1\}^n$ is sampled by A^H at the beginning of its interaction with $\mathcal{A}$) and sets $c^*[i]$ to be the xor of $2^i L$ and its second input (which is either $H(x)$ or y). Note that L is a uniformly random value in the hybrids $\mathsf{G}_{(3,i-1)}$ and $\mathsf{G}_{(3,i)}$ which is independent of $\mathcal{A}$'s challenge query. The value L generated by A^H also has an identical distribution as in the hybrids because L is obtained via a "one-time-pad"-like encryption with the independent and uniformly random x. Coming to the value $\mathsf{c}^*[i]$ generated by $A^H(x, y)$, namely $\mathsf{c}^*[i] \leftarrow 2^i L \oplus y$, it has the same distribution as "$c^*[i]$" derived in $\mathsf{G}_{(3,i)}$, namely "$\mathsf{c}^*[i] \leftarrow \{0,1\}^n$", again because of the one-time-pad encryption with the independent "key" $y \leftarrow \{0,1\}^n$.

In the context of Lemma 1, we have $\Pr[\mathsf{G}_{(3,i-1)} = 1] = P_A^1$ and $\Pr[\mathsf{G}_{(3,i)} = 1] = P_A^2$. Since we have $|P_A^1 - P_A^2| \leq 2q\sqrt{P_B}$, we show that the probability P_B is at most 2^{-n} to complete the proof w.r.t. $2 \leq i \leq m^* - 1$. Note that the corresponding oracle algorithm B^H defined by Lemma 1 simulates the hybrid $\mathsf{G}_{(3,i)}$ towards $\mathcal{A}$ like $A^H(x, y)$ does, with the difference being that B^H stops the execution of $\mathsf{G}_{(3,i)}$ when the oracle H is invoked for the j-th time–for a random $j \leftarrow \{1, \ldots, q\}$ picked by B^H beforehand–and measures the argument of the j-th H-query to see if it obtains x. If the j-th H-query happens before, during, or after $\mathcal{A}$'s challenge query, then to show that the corresponding probability P_B

is at most 2^{-n}, we use a similar reasoning as in the "Before", "During" and "After" cases respectively in the proof of Lemma 3; note that when extending the arguments from the "During" case, we don't have to resort to the "commuting-measurement" strategy, because the values "$M_b[i]$" and "Σ" w.r.t. $\mathcal{A}$'s challenge query are classical – i.e., the analysis will be simpler. It's also not hard to see that the above reasoning extends to the pair of hybrids $\mathsf{G}_{(2,Q,1)}$ and $\mathsf{G}_{(3,1)}$ in an identical manner.

Regarding the hybrids $\mathsf{G}_{(3,m^*-1)}$ and $\mathsf{G}_{(3,m^*)}$, the only difference is that in the former, we have $\mathsf{c}[m^*] \leftarrow \mathsf{M}_b[m^*] \oplus \mathsf{msb}_{\mathsf{len}(\mathsf{M}_b[m^*])}(\mathsf{Pad})$, and in the latter, $\mathsf{c}[m^*] \leftarrow \{0,1\}^n$. But we can see that these hybrids are information-theoretically indistinguishable because $\mathsf{Pad} \leftarrow \{0,1\}^n$ is derived independently of $\mathcal{A}$'s challenge query. So in the former hybrid, $\mathsf{c}[m^*]$ is obtained via a one-time-pad encryption of $\mathsf{M}_b[m^*]$ where the first "$\mathsf{len}(\mathsf{M}_b[m^*])$" bits of Pad are used as the key (note that this key is not used anywhere else in $\mathsf{G}_{(3,m^*-1)}$ and $\mathsf{G}_{(3,m^*)}$, particularly in computing Σ where we can instead use the last "$n - \mathsf{len}(\mathsf{M}_b[m^*])$" bits of Pad). Hence the distribution of $\mathsf{c}[m^*]$ is identical in both hybrids.

Finally, coming to the hybrids $\mathsf{G}_{(3,m^*)}$ and G_4, the only difference is with the derivation of tag t^*: in the former, $\mathsf{t}^* \leftarrow H(2^{m^*}3L \oplus \Sigma)$, and in the latter, $\mathsf{t}^* \leftarrow \{0,1\}^n$. This is equivalent to reprogramming the (internal) quantum random oracle H at the point $x \leftarrow \{0,1\}^n$, where $x = 2^{m^*}3L \oplus \Sigma$. We can hence use Lemma 1 to argue about the indistinguishability of hybrids $\mathsf{G}_{(3,m^*)}$ and G_4 in a similar fashion as above, where we considered the pair $(\mathsf{G}_{(3,i-1)}, \mathsf{G}_{(3,i)})$ for $2 \leq i \leq m^* - 1$; as part of applying Lemma 1, note that we don't have to consider the case when the j-th H-query happens during $\mathcal{A}$'s challenge query, because in G_4, the function H is not invoked at any point when responding to the challenge query. This finishes the proof. Additionally, note that since the values $(\mathsf{c}^*, \mathsf{t}^*)$ are independent of the challenger's bit b, we have $\Pr[\mathsf{G}_4 = 1] = 1/2$.

□

Now we put together the bounds established w.r.t. consecutive hybrids in the above lemmas as follows. For the sake of brevity, we denote $p[\mathsf{G}_k] = \Pr[\mathsf{G}_k = 1]$. We hence have the IND-qCPA advantage of adversary $\mathcal{A}$ w.r.t. OCB2$^-$ to be

$$\mathsf{Adv}^{\mathsf{IND\text{-}qCPA}}_{\mathsf{OCB2}^-}(\mathcal{A}) = \left|p[\mathsf{G}_0] - \frac{1}{2}\right| = |p[\mathsf{G}_0] - p[\mathsf{G}_4]| \leq |p[\mathsf{G}_0] - p[\mathsf{G}_1]| + |p[\mathsf{G}_1] - p[\mathsf{G}_{(2,1,0)}]| +$$
$$\sum_{i=1}^{Q} \left(|p[\mathsf{G}_{(2,i,0)}] - p[\mathsf{G}_{(2,i,1)}]|\right) + \sum_{i=2}^{Q} \left(|p[\mathsf{G}_{(2,i-1,1)}] - p[\mathsf{G}_{(2,i,0)}]|\right) + |p[\mathsf{G}_{(2,Q,1)}] - p[\mathsf{G}_{(3,1)}]| +$$
$$\sum_{i=2}^{m^*} \left(|p[\mathsf{G}_{(3,i-1)}] - p[\mathsf{G}_{(3,i)}]|\right) + |p[\mathsf{G}_{(3,m^*)}] - p[\mathsf{G}_4]|$$
$$\leq \mathsf{Adv}^{\mathsf{qPRF}}_{\mathsf{E_K}}(\mathcal{B}) + (Q + m^*) \cdot \frac{2q}{2^{n/2}} + Q \cdot 2q\sqrt{2^{-n} + \frac{2q}{2^{n/2}}}.$$

Note that our above analysis considered the adversary $\mathcal{A}$ making a single challenge query of block-length m^*, wherein we used the sub-sequence of hybrids $(\mathsf{G}_{(3,1)}, \mathsf{G}_{(3,2)}, \ldots, \mathsf{G}_{(3,m^*)}, \mathsf{G}_4)$ to effectively make the response $(\mathsf{c}^*, \mathsf{t}^*)$ to that particular challenge query independent of bit b. This resulted in the term "$m^* \cdot \frac{2q}{2^{n/2}}$" – following Lemma 5 – in the above bounds on $\mathsf{Adv}^{\mathsf{IND\text{-}qCPA}}_{\mathsf{OCB2}^-}(\mathcal{A})$. If $\mathcal{A}$ made *another* challenge query of block-length m^{**}, a similar sub-sequence of hybrids will result in an *additional* bound "$m^{**} \cdot \frac{2q}{2^{n/2}}$" on $\mathsf{Adv}^{\mathsf{IND\text{-}qCPA}}_{\mathsf{OCB2}^-}(\mathcal{A})$. Now note that the adversary $\mathcal{A}$ can make at-most Q challenge queries, each of block-length at-most m (i.e., m^*, $m^{**} \leq m$) following our assumption in the statement of Theorem 1. Hence, we will obtain a *total* bound of "$Q \cdot m \cdot \frac{2q}{2^{n/2}}$" for Q challenge queries, replacing the term "$m^* \cdot \frac{2q}{2^{n/2}}$" for a single challenge query above. Hence, we finally have the upper

bound on $\mathcal{A}$'s IND-qCPA advantage w.r.t. OCB2$^-$ to be

$$\mathsf{Adv}^{\mathsf{IND\text{-}qCPA}}_{\mathsf{OCB2}^-}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{qPRF}}_{\mathsf{E_K}}(\mathcal{B}) + Q(1+m)\cdot\frac{2q}{2^{n/2}} + Q\cdot 2q\sqrt{2^{-n}+\frac{2q}{2^{n/2}}}.$$

□

We consider it an interesting open question to improve the above concrete security bounds, thereby resulting in a tighter proof of IND-qCPA security for OCB2 when used without AD.

Extension to OCB2f. We know that OCB2 is an insecure mode *classically* as shown in [IIMP19]. As already mentioned earlier, a subsequent fix was proposed in [IIMP19] with the resulting scheme termed as "OCB2f". The only (slight) difference between the schemes OCB2 and OCB2f is the way "Pad" values are computed; see Figure 3.

We now explain how our aforementioned IND-qCPA security analysis of OCB2, as formalized in Theorem 1, extends analogously to OCB2f – with empty AD – in a straightforward manner.

Corollary 1. *If the underlying block-cipher* $\mathsf{E_K}$ *is a quantum-secure PRF, then* OCB2*f mode results in an AE (without AD) scheme which is IND-qCPA secure with random nonces.*

Proof sketch. Note that to extend our positive IND-qCPA security analysis of OCB2 above to OCB2f in an analogous manner, it is sufficient to only modify accordingly the parts of Theorem 1's proof that depend on the *exact* "Pad" values of the corresponding scheme (the sequence of hybrids $\langle \mathsf{G}_i \rangle$ remains the same).

Upon closer inspection, note that the only places that need some (slight) modification are in the proofs of Lemma 3 and Claim 4.2 above, specifically in the way we use Lemma 1 to justify replacing OCB2's "Pad" values with uniformly random n-bit strings. Namely, in a formal IND-qCPA security proof for OCB2f similar to that of Theorem 1 (with random nonces and empty AD), towards replacing OCB2f's new "Pad" values with random n-bit strings in hybrids $\mathsf{G}_{(2,i,1)}$ and $\mathsf{H}_{(2,i,1)}$, the oracle algorithms A^H (in the proof of Lemma 1) and C^H (in the proof of Claim 4.2) respectively will now set the value $\mathsf{Pad}_{(Q-i+1)}$ to be their second input "*xored with* $2^m L_{(Q-i+1)}$" (in contrast to just setting $\mathsf{Pad}_{(Q-i+1)}$ to be their second input as in the corresponding proofs above w.r.t. OCB2). Note again that if A^H's (respectively, C^H's) second input is the uniformly random and independent value y, then we have $\mathsf{Pad}_{(Q-i+1)}$ to also be a uniformly random value just as in the hybrid $\mathsf{G}_{(2,i,1)}$ (respectively, $\mathsf{H}_{(2,i,1)}$) because of a one-time-pad encryption with the independent "key" y.

It's also not hard to see that the rest of Theorem 1's proof steps above apply to OCB2f in an identical fashion because they work with uniformly random "Pad" values; hence, the remainder of the proof works irrespective of the exact way "Pad" values are initially computed in the corresponding schemes. □

4.3 IND-qCPA Insecurity of OCB2 with Associated Data

Theorem 1 shows that OCB2 is IND-qCPA secure with random nonces when used as a "pure" AE scheme like OCB1 – i.e., not taking any associated data (AD) as input. We now show that OCB2 is *insecure* in the same setting (IND-qCPA with random nonces) when used as an AEAD scheme by exploiting the way AD is authenticated by the encryption algorithm.

Before we describe our IND-qCPA attack, let's consider a quantum forgery attack against OCB2 adapted from [KLLN16] that uses Simon's algorithm. The attack exploits the processing of AD by $\mathsf{OCB2.Enc}(\mathsf{K},\cdot)$ as well. Specifically, we consider the following function

$f_{\mathsf{N}} : \{0,1\}^n \to \{0,1\}^n$ such that $f_{\mathsf{N}}(\mathsf{A}) = \mathsf{OCB2.Enc}_{\mathsf{K}}(\mathsf{N}, \mathsf{A}\|\mathsf{A}\|0^n, \epsilon)$. From Figure 3, we can see that

$$f_{\mathsf{N}}(\mathsf{A}) = \mathsf{E}_{\mathsf{K}}(3L) \oplus \mathsf{E}_{\mathsf{K}}(2^3 3V \oplus \mathsf{E}_{\mathsf{K}}(2V \oplus \mathsf{A}) \oplus \mathsf{E}_{\mathsf{K}}(2^2 V \oplus \mathsf{A})),$$

where $L = \mathsf{E}_{\mathsf{K}}(\mathsf{N})$ and $V = 3^2\mathsf{E}_{\mathsf{K}}(0^n)$. Note that $f_{\mathsf{N}}(\cdot)$ is periodic with period $2V \oplus 2^2V$. Hence, similar to quantum superposition attacks in [KLLN16] against OCB1 and OCB3 discussed above, we can recover the value $2V \oplus 2^2V$–and therefore, $\mathsf{E}_{\mathsf{K}}(0^n)$–with $O(n)$ quantum queries to the OCB2 encryption oracle using Simon's algorithm.

Now coming to our IND-qCPA attack against OCB2, on a high-level, we extend the recent (existential) forgery attacks against OCB3 presented in [BLNS21] that use Deutsch's algorithm to gain *raw block-cipher access* w.r.t. OCB2; namely, the ability to evaluate $\mathsf{E}_{\mathsf{K}}(\mathsf{inp})$ on arbitrary inputs inp of our choice when we only have quantum access to $\mathsf{OCB2.Enc}(\mathsf{K}, .)$. This is sufficient to break the IND-qCPA security of OCB2.

Towards evaluating $\mathsf{E}_{\mathsf{K}}(\mathsf{inp})$, we first recover the value $\mathsf{E}_{\mathsf{K}}(0^n)$ using Simon's algorithm, as discussed above. Then we fix two single-block AD inputs $\alpha_0 := 2{\cdot}3V$ and $\alpha_1 := 2{\cdot}3V \oplus \mathsf{inp}$, where $V = 3^2\mathsf{E}_{\mathsf{K}}(0^n)$. Now similar to the attack strategy in [BLNS21], we consider the function $f^{(i)} : \{0,1\} \to \{0,1\}$ such that $f^{(i)}(b) = i$th bit of $\{\mathsf{OCB2.Enc}(\mathsf{K}, \mathsf{N}, \alpha_b, \epsilon)\}$, for a random nonce N. From Figure 3, we have $f^{(i)}(b) = i$th bit of $\{\mathsf{E}_{\mathsf{K}}(3L) \oplus \mathsf{E}_{\mathsf{K}}(2 \cdot 3V \oplus \alpha_b)\}$, where $L = \mathsf{E}_{\mathsf{K}}(\mathsf{N})$. Note that the function $f^{(i)}$ can be computed in superposition with a *single* quantum query to the oracle $\mathsf{OCB2.Enc}(\mathsf{K}, \cdot)$.[9] Here we again use the techniques of [HS18, BBC+21] to truncate the other $(n-1)$ bits of the output of $\mathsf{OCB2.Enc}(\mathsf{K}, \cdot)$. This allows us to apply Deutsch's algorithm on $f^{(i)}$, where with a single quantum query, we can recover the value

$$\begin{aligned} f^{(i)}(0) \oplus f^{(i)}(1) &= i\text{th bit of } \{\mathsf{E}_{\mathsf{K}}(3L) \oplus \mathsf{E}_{\mathsf{K}}(0^n)\} \oplus i\text{th bit of } \{\mathsf{E}_{\mathsf{K}}(3L) \oplus \mathsf{E}_{\mathsf{K}}(\mathsf{inp})\} \\ &= i\text{th bit of } \{\mathsf{E}_{\mathsf{K}}(0^n) \oplus \mathsf{E}_{\mathsf{K}}(\mathsf{inp})\}. \end{aligned}$$

We can then recover all n bits of $\{\mathsf{E}_{\mathsf{K}}(0^n) \oplus \mathsf{E}_{\mathsf{K}}(\mathsf{inp})\}$ by applying Deutsch's algorithm to each of the n functions $\langle f^{(i)} \rangle_{i \in [n]}$; note that even though the random nonce N may change with each subsequent application of Deutsch's algorithm, the value $\{\mathsf{E}_{\mathsf{K}}(0^n) \oplus \mathsf{E}_{\mathsf{K}}(\mathsf{inp})\}$ is independent of N. The knowledge of $\mathsf{E}_{\mathsf{K}}(0^n)$ thus allows us to obtain $\mathsf{E}_{\mathsf{K}}(\mathsf{inp})$.

We now sketch our IND-qCPA attack against OCB2 below.

1. Using $O(n)$ *quantum* encryption queries, recover the value $2V \oplus 2^2V$–and hence, $\mathsf{E}_{\mathsf{K}}(0^n)$–via Simon's algorithm (similar to the quantum forgery attack in [KLLN16] against OCB2 discussed above).

2. For the challenge query, pick arbitrary single-block messages M_0, M_1 such that $\mathsf{M}_0 \neq \mathsf{M}_1$ and select $\mathsf{A} = \epsilon$; record the nonce N used by the challenger to encrypt either M_0 or M_1.

3. Evaluate the value $L = \mathsf{E}_{\mathsf{K}}(N)$ using n quantum encryption queries via Deutsch's algorithm (namely, the raw block-cipher access discussed above). Similarly, evaluate the value $\mathsf{Pad} = \mathsf{E}_{\mathsf{K}}(2L \oplus n)$.

4. Output $b' = 0$ if $\mathsf{c}^* = \mathsf{M}_0 \oplus \mathsf{Pad}$, and output $b' = 1$ otherwise; here c^* is the ciphertext corresponding to the challenge query.

It's not hard to see that the above attack succeeds w.h.p., since we're essentially recomputing the values (e.g., L and Pad) involved in processing the challenge query in the IND-qCPA game.

[9] Given a black-box quantum circuit for $\mathsf{OCB2.Enc}_{\mathsf{K}}(\cdot)$, the quantum circuit to evaluate $f^{(i)}$ will require a unitary gate that maps b to α_b, similar to the quantum forgery attacks against CBC-MAC in [KLLN16]; see [KLLN16, Figures 2.1 and 10].

Also there is another thing worth pointing out on how our results relate to the *classical* cryptanalysis of OCB2 in [IIMP19]. Namely, a key component of *all* classical attacks against OCB2 proposed in [IIMP19] is that they query an empty AD. Therefore, another fix to OCB2 suggested in [IIMP19] is to *always* keep the AD to be non-empty during encryption/decryption operations. This is in stark contrast to our IND-qCPA security analysis of OCB2 where we show a positive result when AD is always kept empty and a negative result – via an attack – which exploits the processing of (non-empty) AD during encryption.

Extension to OCB2f. When extending our above IND-qCPA attack against OCB2 to OCB2f, the *only* difference will be in Step 3 where we now evaluate the "Pad" value to be "$2L \oplus \mathsf{E}_{\mathsf{K}}(2L \oplus n)$"; the rest of the steps stay the same. The main reason is that the application of Simon's and Deutsch's algorithms above in Steps 1 and 3 respectively extends to OCB2f *as-it-is*. This is because those steps query the encryption oracle on empty messages and non-empty AD, and the encryption algorithms of OCB2 and OCB2f process AD in an identical fashion via the PMAC subroutine (see Figure 3).

4.4 IND-qCPA Insecurity of OCB2 for Adaptively Chosen Nonces

We conclude our IND-qCPA security analysis in this section by describing another way to break the confidentiality of OCB2, this time used as a pure AE scheme – i.e., our attack does not exploit the processing of AD by the $\mathsf{OCB2.Enc}_{\mathsf{K}}(\cdot)$ algorithm, in contrast to the attack presented in Subsection 4.3. Namely, we consider a stronger adversarial setting where the adversary can *adaptively* pick the classical nonces for the encryption queries in the IND-qCPA security game.

On a high-level, we adapt a recent superposition attack presented in [BBC+21, Subsection 3.2] that allows one to recover $L = \mathsf{E}_{\mathsf{K}}(\mathsf{N})$ used in the encryption routine of OCB2 via Simon's algorithm.[10] Similar to the attack strategy in [BBC+21], consider the function

$$g : \{0,1\}^{(2n+2)n+\tau} \rightarrow \{0,1\}^{(n+2)n+\tau}$$
$$g(\mathsf{c}_1, \mathsf{c}_2, \ldots \mathsf{c}_{2n+1}, \mathsf{c}_{2n+2}, \mathsf{t}) = (\mathsf{c}_1, \mathsf{c}_2 \oplus \mathsf{c}_3, \mathsf{c}_4 \oplus \mathsf{c}_5, \ldots, \mathsf{c}_{2n} \oplus \mathsf{c}_{2n+1}, \mathsf{c}_{2n+2}, \mathsf{t}).$$

Here, the c_is are n-bit blocks and t is a τ-bit block. Now consider the following function $f_{\mathsf{N}} : \{0,1\}^{n^2} \rightarrow \{0,1\}^{(n+2)n+\tau}$ such that

$$f_{\mathsf{N}}(\mathsf{M}_1, \ldots \mathsf{M}_n) = g \circ \mathsf{OCB2.Enc}(\mathsf{K}, \mathsf{N}, \epsilon, 0^n \| \mathsf{M}_1 \| \mathsf{M}_1 \| \ldots \| \mathsf{M}_n \| \mathsf{M}_n \| 0^n)$$

for some randomly chosen nonce N, where the M_is are n-bit blocks. Following the discussion in Subsection 3.1, and in [BBC+21, Subsection 3.2] w.r.t. the advanced superposition attack, it's not hard to see that f_{N} is periodic with n linearly independent periods $\langle s_i \rangle_{i \in [n]}$ such that $s_i = ((0^n)^{(i-1)}, 2^{2i}L \oplus 2^{2i+1}L, (0^n)^{(n-i)})$, where $L = \mathsf{E}_{\mathsf{K}}(\mathsf{N})$. Also, we can see that g is a linear function, i.e., $g(\mathsf{c} \oplus \mathsf{c}') = g(\mathsf{c}) \oplus g(\mathsf{c}')$ for any two valid inputs c, c'. Hence we can use [BBC+21, Lemma 2] (i.e., computing a linear function of a quantum oracle's output) to compute f_{N} in superposition with a single quantum query to the $\mathsf{OCB2.Enc}(\mathsf{K}, \mathsf{N}, \cdot)$ oracle. This allows us to apply Simon's algorithm on f_{N}, where we use similar arguments as that of [BBC+21] towards showing the non-existence of "unwanted periods" in f_{N}. So with a single quantum query to f_{N}, Simon's algorithm recovers a vector $y \in \{0,1\}^{n^2}$ orthogonal to *each* of the n periods $\langle s_i \rangle_{i \in [n]}$. If we write y as $y = (y_1, y_2, \ldots, y_n)$, then we can solve the resulting n equations – namely, "$y_i \cdot (2^{2i}L \oplus 2^{2i+1}L) = 0$", for $i \in [n]$ – to recover the value L. Also note that as part of Simon's algorithm, when we measure the quantum register corresponding to the output of $f_{\mathsf{N}}(\cdot)$, we recover the *fixed* classical values of the

[10] However, the authors of [BBC+21] do not expand on how the knowledge of $L = \mathsf{E}_{\mathsf{K}}(\mathsf{N})$ allows one to break certain quantum-security notions (e.g., IND-qCPA, EUF-qCMA) w.r.t. the OCB modes.

first and last ciphertext blocks – namely, $\mathsf{c}_1 = 2^1L \oplus \mathsf{E}_\mathsf{K}(2^1L)$ and $\mathsf{c}_{2n+2} = \mathsf{E}_\mathsf{K}(2^{2n+2}L \oplus n)$ – along with the tag $\mathsf{t} = \mathsf{msb}_\tau(\mathsf{E}_\mathsf{K}(2^{2n+2}3L))$. This tag is also fixed as the checksum Σ of all messages in superposition that are queried w.r.t. the $\mathsf{OCB2.Enc}(\mathsf{K}, \mathsf{N}, \cdot)$ oracle is zero.

The knowledge of these classical values – in particular, L and c_1 – will allow us to break the IND-qCPA security of OCB2, in a setting with *adaptively chosen* (non-repeating) nonces, with just a single quantum encryption query. We sketch our IND-qCPA attack below:

1. Pick a random nonce N and use a single *quantum* encryption query to recover the *classical* values $L = \mathsf{E}_\mathsf{K}(\mathsf{N})$, $\mathsf{c}_1 = 2^1L \oplus \mathsf{E}_\mathsf{K}(2^1L)$, $\mathsf{c}_{2n+2} = \mathsf{E}_\mathsf{K}(2^{2n+2}L \oplus n)$ and $\mathsf{t} = \mathsf{msb}_\tau(\mathsf{E}_\mathsf{K}(2^{2n+2}3L))$ via Simon's algorithm, as discussed above.

2. For the challenge query, select the nonce N^* to be $\mathsf{N}^* = 2^1L$.[11] By doing so, we know the corresponding initial offset L^* to be $L^* = \mathsf{E}_\mathsf{K}(\mathsf{N}^*) = \mathsf{c}_1 \oplus \mathsf{N}^*$, where c_1 is one of the values recovered in the previous step.

3. Define $\mathsf{m}_0 = 2^1L^* \oplus \mathsf{N} \in \{0,1\}^n$ and a random $\mathsf{m}_1 \in \{0,1\}^n$ such that $\mathsf{m}_1 \neq \mathsf{m}_0$, where N is the nonce used in the quantum encryption query above. Also choose an arbitrary $\mathsf{m} \in \{0,1\}^n$. Select the two 2-block messages in the challenge query as $\mathsf{M}_0 = \mathsf{m}_0\|\mathsf{m}$, $\mathsf{M}_1 = \mathsf{m}_1\|\mathsf{m}$ and $\mathsf{A} = \epsilon$.

4. Upon receiving the response $(\mathsf{c}^*, \mathsf{t}^*)$ from the challenger, output bit $b' = 0$ if we have $\mathsf{c}^*[1] \oplus 2^1L^* = L$, and output $b' = 1$ otherwise. Here L is another value recovered in the first step.

It's not hard to see that the above attack succeeds w.h.p. Looking at Figure 3 (Line 4), if M_0 was encrypted by the challenger, then we have

$$\mathsf{c}^*[1] := 2^1L^* \oplus \mathsf{E}_\mathsf{K}(2^1L^* \oplus \mathsf{m}_0) = 2^1L^* \oplus \mathsf{E}_\mathsf{K}(\mathsf{N}) = 2^1L^* \oplus L.$$

And if M_1 was encrypted by the challenger, we have

$$\mathsf{c}^*[1] := 2^1L^* \oplus \mathsf{E}_\mathsf{K}(2^1L^* \oplus \mathsf{m}_1) \neq 2^1L^* \oplus \mathsf{E}_\mathsf{K}(2^1L^* \oplus \mathsf{m}_0)\ (= 2^1L^* \oplus L),$$

where the inequality follows from the fact that E_K is a permutation.

Extension to OCB2f. It's again not hard to see that the above attack applies to OCB2f in a similar fashion. The *only* difference will be that, since OCB2f computes the value Pad differently (see Figure 3), we recover a different value of c_{2n+2} in Step 1: namely, $\mathsf{c}_{2n+2} = 2^{2n+2}L \oplus \mathsf{E}_\mathsf{K}(2^{2n+2}L \oplus n)$. Otherwise, the remaining steps of the above attack extend to OCB2f *as-it-is*. This is because those steps mainly focus on the way the *non-final* message blocks (i.e., all blocks of the message except the last block) are encrypted by the corresponding scheme, and OCB2 and OCB2f encrypt the non-final message blocks in an identical manner.

Hence, in conjunction with Theorem 1, the above attack necessitates the use of random nonces in OCB2(f) – in addition to not allowing AD as input – if we are aiming for confidentiality in a quantum setting.

[11] Note that if E_K behaves as a secure PRP, then the probability that "$\mathsf{N} = \mathsf{N}^*$", or equivalently, "$\mathsf{N} = 2^1\mathsf{E}_\mathsf{K}(\mathsf{N})$" can be considered to be negligible w.r.t. the random nonce N.

5 EUF-qCMA Security Analysis of OCB2(f)

In Subsection 4.3, we have described an unforgeability (EUF-qCMA) attack against OCB2 from [KLLN16] which exploits the processing of AD by the scheme. This is in contrast to the (alternative) EUF-qCMA attacks against OCB1 and OCB3 in [KLLN16] (described in Subsection 3.1) which only target the pure encryption parts of the respective schemes.

We now provide an unforgeability attack that breaks the EUF-qCMA security–with random nonces – of OCB2(f) when it is operated as a pure AE mode (i.e., with no input AD). On a high-level, we again adapt the advanced superposition attack presented in [BBC+21, Subsection 3.2] which allows one to recover $L = \mathsf{E}_\mathsf{K}(\mathsf{N})$ used in $\mathsf{OCB2.Enc}_\mathsf{K}(\mathsf{N}, \cdot)$ via Simon's algorithm. Since we know that OCB2 fails to satisfy existential unforgeability in a *classical* setting, as shown in [IIMP19], we will be describing our attack w.r.t. OCB2f in the following.

The attack starts along similar lines as the first step of our IND-qCPA attack sketched above in Subsection 4.4. The only difference is that we instead consider the function $f_\mathsf{N} : \{0,1\}^{n^2} \to \{0,1\}^{(n+2)n+\tau}$ such that

$$f_\mathsf{N}(\mathsf{M}_1, \ldots \mathsf{M}_n) = g \circ \mathsf{OCB2f.Enc}(\mathsf{K}, \mathsf{N}, \epsilon, n \| \mathsf{M}_1 \| \mathsf{M}_1 \| \ldots \| \mathsf{M}_n \| \mathsf{M}_n \| n)$$

for some randomly chosen nonce N (and the M_is are n-bit blocks). Following our discussion in Subsection 4.4, it's not hard to see that by applying Simon's algorithm on f_N using a single quantum query, we recover the classical values of $L = \mathsf{E}_\mathsf{K}(\mathsf{N})$, $\mathsf{c}_1 = 2^1 L \oplus \mathsf{E}_\mathsf{K}(2^1 L \oplus n)$, $\mathsf{c}_{2n+2} = n \oplus 2^{2n+2} L \oplus \mathsf{E}_\mathsf{K}(2^{2n+2} L \oplus n)$ and $\mathsf{t} = \mathsf{msb}_\tau(\mathsf{E}_\mathsf{K}(2^{2n+2} 3L))$.

The knowledge of these classical values will allow us to break the EUF-qCMA security of OCB2f, in a random nonce setting, with just a single encryption query (i.e., $q = 1$). Towards computing the forgeries, we additionally exploit the fact that the encryption algorithm of OCB2f processes the last block of messages differently (i.e., a "one-time-pad" type encryption is applied). We now sketch our EUF-qCMA attack below:

1. Upon receiving a random nonce N, use a single *quantum* encryption query to recover the *classical* values $L = \mathsf{E}_\mathsf{K}(\mathsf{N})$, $\mathsf{c}_1 = 2^1 L \oplus \mathsf{E}_\mathsf{K}(2^1 L \oplus n)$, $\mathsf{c}_{2n+2} = n \oplus 2^{2n+2} L \oplus \mathsf{E}_\mathsf{K}(2^{2n+2} L \oplus n)$ and $\mathsf{t} = \mathsf{msb}_\tau(\mathsf{E}_\mathsf{K}(2^{2n+2} 3L))$ via Simon's algorithm, as discussed above.

2. Compute two forgeries w.r.t. the same nonce N and empty AD as follows:

   Select a single-block message $\mathsf{M}' = 2^{2n+2} 3L \oplus 2 \cdot 3L$. It's not hard to see that the (one-time-pad) ciphertext c' corresponding to M' w.r.t. the $\mathsf{OCB2f.Enc}(\mathsf{K}, \cdot)$ algorithm is $\mathsf{c}' = \mathsf{M}' \oplus 2^1 L \oplus \mathsf{E}_\mathsf{K}(2^1 L \oplus n) = \mathsf{M}' \oplus \mathsf{c}_1$. Similarly, we have the corresponding tag

   $$\mathsf{t}' = \mathsf{msb}_\tau(\mathsf{E}_\mathsf{K}(2 \cdot 3L \oplus \mathsf{M}')) = \mathsf{msb}_\tau(\mathsf{E}_\mathsf{K}(2^{2n+2} 3L)) = \mathsf{t}.$$

   Therefore, output $(\mathsf{N}, , \mathsf{c}', \mathsf{t}')$ as the first forgery.

   Now select another single-block message $\mathsf{M}'' = 2^1 L \oplus n \oplus 2 \cdot 3L$. It's not hard to see that the (one-time-pad) ciphertext c'' corresponding to M'' is $\mathsf{c}'' = \mathsf{M}'' \oplus 2^1 L \oplus \mathsf{E}_\mathsf{K}(2^1 L \oplus n) = \mathsf{M}'' \oplus \mathsf{c}_1$. Similarly, we have the corresponding tag

   $$\mathsf{t}'' = \mathsf{msb}_\tau(\mathsf{E}_\mathsf{K}(2 \cdot 3L \oplus \mathsf{M}'')) = \mathsf{msb}_\tau(\mathsf{E}_\mathsf{K}(2^1 L \oplus n)) = \mathsf{msb}_\tau(\mathsf{c}_1 \oplus 2^1 L).$$

   Therefore, output $(\mathsf{N}, , \mathsf{c}'', \mathsf{t}'')$ as the second forgery.

The above attack succeeds w.h.p., since we have $\mathsf{OCB2f.Dec}(\mathsf{K}, \mathsf{N}, \varepsilon, \mathsf{c}', \mathsf{t}') = \mathsf{M}' \neq \perp$ and $\mathsf{OCB2f.Dec}(\mathsf{K}, \mathsf{N}, \varepsilon, \mathsf{c}'', \mathsf{t}'') = \mathsf{M}'' \neq \perp$.

6 UUF-qCMA Security Analysis of OCB Modes

As discussed above, it was shown in [KLLN16] that *all* 3 versions of OCB do not offer authentication in a quantum setting – in the sense of *existential unforgeability*. Specifically, polynomial-time attacks that use Simon's algorithm were presented in [KLLN16] which break the formal notion of EUF-qCMA security (as defined in [BZ13a]) of each of the OCB modes by exploiting quantum access to their respective encryption oracles. In this section, we briefly describe more powerful quantum attacks targeting authentication – namely, *universal forgeries* – w.r.t. certain versions of OCB, thereby breaking their respective UUF-qCMA security (see Definition 4).

If we first consider OCB2, then recall that towards attacking the IND-qCPA security of the scheme in Subsection 4.3, we were able to evaluate the underlying block-cipher E_K on arbitrary inputs of our choice. This raw block-cipher access allows us to compute universal forgeries w.r.t. *any* tuple $(\mathsf{N}^*, \mathsf{A}^*, \mathsf{M}^*)$ efficiently (polynomial-time): we simply execute the $\mathsf{OCB2.Enc}(\mathsf{K}, \cdot)$ algorithm on input $(\mathsf{N}^*, \mathsf{A}^*, \mathsf{M}^*)$ *locally* by emulating all block-cipher evaluations involved via this raw access, in the setting with random nonces. This is quite similar to the *classical* universal forgery attacks presented in [IIMP19] against OCB2. However, our quantum attack also succeeds in computing universal forgeries w.r.t. the fixed version of OCB2, namely OCB2f, proposed in [IIMP19]. Similarly, it's not hard to extend the "Deutsch's-algorithm-based" quantum forgery attack against OCB3 described in [BLNS21, Subsection 3.1] to obtain raw block-cipher access, just as we showed for OCB2 in Subsection 4.3. This access again leads to universal forgery attacks against OCB3 in a quantum setting (with random nonces).

When it comes to OCB1 however, note that the scheme does not take AD as input. Hence, our quantum universal forgery attacks against OCB2(f) and OCB3, which first obtain raw block-cipher access by exploiting the way AD is authenticated in their respective encryption algorithms, do not extend to OCB1. In other words, OCB1 surprisingly appears to be immune to universal forgeries in a quantum setting *with random nonces* in contrast to OCB2 and OCB3. We leave the task of formally proving/disproving the UUF-qCMA security of OCB1 with random nonces as an open question. However, if we give an adversary the ability to choose the classical nonces when querying the quantum encryption oracle (with a restriction that the nonces do not repeat in subsequent queries and cannot be equal to the challenge nonce N^*), then there is a way to compute universal forgeries w.r.t. OCB1. Looking at the $\mathsf{OCB1.Enc}(\mathsf{K}, \cdot)$ algorithm in Figure 1, note that w.r.t. an input nonce N, if we already know the values of $L = \mathsf{E}_\mathsf{K}(0^n)$ and $\Delta_0 = \mathsf{E}_\mathsf{K}(\mathsf{N} \oplus L)$, then we can obtain raw-block cipher access by carefully picking our message M as follows: to evaluate $\mathsf{E}_\mathsf{K}(\mathsf{inp})$, we pick $\mathsf{M} := \mathsf{m}_0 \| \mathsf{m}_1$ such that $\mathsf{m}_0 = \Delta_1 \oplus \mathsf{inp}$, where $\Delta_1 = \gamma_1 \cdot L \oplus \Delta_0$, and m_1 is an arbitrary single-block message. By querying (N, M) to the encryption oracle and obtaining the resulting ciphertext and tag (c, t), note that we simply have $\mathsf{E}_\mathsf{K}(\mathsf{inp}) = \mathsf{c}[1] \oplus \Delta_1$. This results in a universal forgery attack against OCB1 along similar lines to that against OCB2 in [IIMP19]. We give a sketch of the attack below. Given a target nonce N^* and message M^*, we compute the forged ciphertext and tag $(\mathsf{c}^*, \mathsf{t}^*) \leftarrow \mathsf{OCB1.Enc}(\mathsf{K}, \mathsf{N}^*, \mathsf{M}^*)$ as follows:

1. Using $O(n)$ *quantum* encryption queries, w.r.t. non-repeating nonces of our choice that are not equal to N^*, recover the value L via Simon's algorithm (similar to the IND-qCPA attack against OCB1 in Subsection 3.2).

2. Select the "initial" nonce $\mathsf{N}^{(0)} := L$ such that we have the corresponding initial offset $\Delta_0^{(0)}$ to be $\Delta_0^{(0)} = \mathsf{E}_\mathsf{K}(\mathsf{N}^{(0)} \oplus L) = L$. Subsequently, pick sufficiently many (we denote this number with k and remark that it is sufficient for k to be the length of the message for the forgery attack) non-repeating nonces $\langle \mathsf{N}^{(i)} \rangle_{i \in [k]}$ where $\mathsf{N}^{(i)} \neq \mathsf{N}^*$.

3. Choose a $(k+1)$-block classical message M such that $\mathsf{M}[i] := \Delta_i^{(0)} \oplus (\mathsf{N}^{(i)} \oplus L)$, where $\Delta_i^{(0)} = \gamma_i \cdot L \oplus \Delta_0^{(0)}$, for $i \in [k]$. Pick the last message block $\mathsf{M}[k+1]$ arbitrarily.

Algorithm $\mathsf{XTS.Enc}(\mathsf{K}, \mathsf{T}, \mathsf{M})$:

$(\mathsf{K}_1, \mathsf{K}_2) \leftarrow \mathsf{K}$;
$L \leftarrow \mathsf{E}_{\mathsf{K}_2}(\mathsf{T})$;
$(\mathsf{M}[1], \ldots, \mathsf{M}[m]) \leftarrow \mathsf{M}$;
for $i \leftarrow 1$ **to** $m-1$ **do**
  $\mathsf{c}[i] \leftarrow \alpha^i L \oplus \mathsf{E}_{\mathsf{K}_1}(\alpha^i L \oplus \mathsf{M}[i])$;
end
if $\mathsf{len}(\mathsf{M}[m]) = n$; **then**
  $\mathsf{c}[m] \leftarrow \alpha^m L \oplus \mathsf{E}_{\mathsf{K}_1}(\alpha^m L \oplus \mathsf{M}[m])$;
end
else
  $\mathsf{c}[m] \leftarrow \mathsf{msb}_{\mathsf{len}(\mathsf{M}[m])}(\mathsf{c}[m-1])$;
  $X \leftarrow \mathsf{lsb}_{n-\mathsf{len}(\mathsf{M}[m])}(\mathsf{c}[m-1])$;
  $Y \leftarrow \mathsf{M}[m] \| X$;
  $\mathsf{c}[m-1] \leftarrow \alpha^m L \oplus \mathsf{E}_{\mathsf{K}_1}(\alpha^m L \oplus Y)$;
end
return c;

Figure 5: XTS block-cipher mode. Here T is the index of sector on which the encryption of data M is to be stored. In the context of OCB2, T can be seen as a "nonce". Also, α is a generator of the field $\mathbb{F}_2^n$.

4. Query $(\mathsf{N}^{(0)}, \mathsf{M})$ to the $\mathsf{OCB1.Enc}(\mathsf{K}, \cdot)$ oracle and obtain the resulting ciphertext and tag (c, t). Recover the initial offset values $\Delta_0^{(i)}$ w.r.t. each of the nonces $\mathsf{N}^{(i)}$ as $\Delta_0^{(i)} = \mathsf{E}_{\mathsf{K}}(\mathsf{N}^{(i)} \oplus L) = \mathsf{c}[i] \oplus \Delta_i^{(0)}$.
5. Emulate all block-cipher evaluations involved in execution of $\mathsf{OCB1.Enc}(\mathsf{K}, \mathsf{N}^*, \mathsf{M}^*)$ by subsequently querying the encryption oracle with nonces $\mathsf{N}^{(i)}$ and appropriately chosen messages $\mathsf{M}^{(i)}$. Raw block-cipher access w.r.t. OCB1 is possible since we know the value L from Step 1 and initial offset values $\Delta_0^{(i)}$ of nonces $\mathsf{N}^{(i)}$ from Step 4.

7 Post-Quantum Security of XTS

XTS is a block-cipher mode of operation for encrypting data on sector-level storage devices such as HDD. It has been standardized by IEEE [oEE08] and recommended by NIST (with AES as the underlying block-cipher) [Dwo] for storage encryption. As mentioned in Subsection 1.2, XTS is structurally quite similar to OCB2; see Figure 5 for a detailed specification of XTS' encryption.

When compared to OCB2's encryption (described in Figure 3), a main difference is that XTS uses *two* independent secret keys K_1, K_2: the latter is used to compute the value L as "$L \leftarrow \mathsf{E}_{\mathsf{K}_2}(\mathsf{T})$" and the former is used as a key in subsequent block-cipher invocations for encrypting each individual message block $\mathsf{M}[i]$. Another difference is the way the last message block $\mathsf{M}[m]$ is processed: in contrast to the "one-time-pad" type encryption applied by OCB2, XTS uses the so-called *ciphertext stealing* technique (see Fig. 5).

If we have XTS only encrypting messages whose last blocks are of n bits, thereby allowing us to ignore the effect of ciphertext stealing, then it's relatively straightforward to extend our formal proof of IND-qCPA security for OCB2–when operated as a pure AE mode–to XTS in a setting where the "nonces" T are chosen uniformly at random; i.e., when the XTS-encrypted data are written on *random* sectors of the corresponding storage device. We note that in practice, sectors are typically chosen according to a predefined order when writing data (as is the case, for example, in the *sequential write* feature of HDDs).

So one can interpret our following positive IND-qCPA security result for XTS to be of theoretical interest wherein XTS is treated as a general tweakable encryption primitive. At the same time, it is worth pointing out that the setting with random sector indices T is not completely theoretical as it is closely related to the *random write* feature of HDDs (in contrast to *sequential write*). Hence, it would be interesting to assess the practical benefits of such a random write feature of sector-level storage devices in a post-quantum setting. We also remark that our above assumption of writing data on random disk sectors implicitly forbids an adversary to *overwrite* data on the same sector – a setting that is typically allowed in security models for disk encryption schemes in the literature [KMV17, IM19], as discussed in Subsection 1.2.

Now to give a sketch of the IND-qCPA security proof for XTS following that of Theorem 1, we proceed with a sequence of hybrids starting with the IND-qCPA game w.r.t. XTS with random "nonces". In the next hybrid, we replace $\mathsf{E}_{\mathsf{K}_2}$ in the $\mathsf{XTS.Enc}$ algorithm with a truly random function $H_2 : \{0,1\}^n \to \{0,1\}^n$, owing to the quantum security of $\mathsf{E}_{\mathsf{K}_2}$. This allows us to generate the values L in the next hybrid w.r.t. the adversary's queries in the IND-qCPA game as "$L \leftarrow \{0,1\}^n$". Here we apply Lemma 1 in a similar fashion as in the proof of Lemma 4. But our analysis will be much simpler since we only need to consider Line 2 ("$L \leftarrow H_2(\mathsf{T})$") of the $\mathsf{XTS.Enc}$ algorithm when measuring H_2-queries as a part of applying Lemma 1. In the next hybrid, we replace $\mathsf{E}_{\mathsf{K}_1}$ in the $\mathsf{XTS.Enc}$ algorithm with another truly random function $H_1 : \{0,1\}^n \to \{0,1\}^n$, this time relying on the quantum security of $\mathsf{E}_{\mathsf{K}_1}$. Finally, we replace each block of the challenge ciphertext c^* with uniformly random values, similar to the sub-sequence of hybrids "$\mathsf{G}_{(3,i)}, i \in [m^*]$" in the proof of Theorem 1. Here we apply Lemma 1 in a similar fashion as in the proof of Lemma 5. Again our analysis will be simpler because, in contrast to OCB2, all individual message blocks in the challenge query are processed in a uniform manner – note that we're ignoring ciphertext stealing in this setting. Hence, it's not hard to obtain the following based on our IND-qCPA security analysis of OCB2, as formalized in Theorem 1:

Corollary 2. *The* XTS *mode results in an IND-qCPA secure disk-encryption scheme if*

- *the underyling block-cipher* E *is a quantum-secure PRF,*
- *the sector index* T *is chosen uniformly at random when encrypting a new message,*
- *and the length of messages is a multiple of* n *bits (the block size of* E*) .*

However, note that in Subsection 4.4, we showed a way to break the IND-qCPA security of OCB2 if an adversary can adaptively choose the nonces. Surprisingly, the attack does not extend to XTS in a straightforward manner when the adversary can pick the (non-repeating) sector indices T adaptively when making queries in the IND-qCPA game. To go into some more details, by applying Simon's algorithm in a similar way as in Subsection 4.4 towards breaking the IND-qCPA security of XTS, we can evaluate $\mathsf{E}_{\mathsf{K}_2}(\cdot)$ *only on a single input*; namely, we can only recover the value $L = \mathsf{E}_{\mathsf{K}_2}(\mathsf{T})$ where T is the sector index chosen by the IND-qCPA adversary in the corresponding encryption query. But the knowledge of this value – along with arbitrary evaluations of $\mathsf{E}_{\mathsf{K}_1}(\cdot)$ – does not appear to enable us to choose the "nonce" T^* for the challenge query in a way which allows us to win the IND-qCPA security game, in contrast to our attack in Subsection 4.4. The reason, at-least on a higher level, seems to be the fact that the "nonces" T and the message blocks $\mathsf{M}[i]$s are processed by *separate* block-cipher instances in $\mathsf{XTS.Enc}$: "$\mathsf{E}_{\mathsf{K}_2}(\cdot)$" is used for the former and "$\mathsf{E}_{\mathsf{K}_1}(\cdot)$" is used for the latter. We leave the task of formally proving/disproving the IND-qCPA security of XTS in this adaptive setting as an open question.

But if we instead consider a variant of XTS that uses a *single* block-cipher key, i.e., $\mathsf{K}_1 = \mathsf{K}_2 = \mathsf{K}$, then note that a single block-cipher instance "$\mathsf{E}_{\mathsf{K}}(\cdot)$" is used throughout

the encryption algorithm, similar to OCB2.Enc. This allows us to extend our *adaptive* IND-qCPA attack against OCB2 from Subsection 4.4 to this "single-key" variant of XTS in a straightforward manner. Again note that the attack works even if E_K were to be a quantum-secure PRP. We thereby partially answer the question raised by Liskov and Minematsu in [LM08] on whether XTS offers any advantage over its "single-key" variant: the answer seems to lie with the ability of these schemes in providing confidentiality in a quantum setting.

8 Conclusion and Open Questions

In this paper, we made significant progress towards improving our understanding of the post-quantum security of OCB with respect to confidentiality and universal unforgeability. Our work gives rise to interesting open questions, which we summarize here.

UUF-qCMA Security of OCB1 with Random Nonces. Our universal unforgeability attack on OCB1 necessarily requires the quantum adversary to adaptively choose the nonces in the UUF-qCMA security game. We leave it open to extend our attack on OCB1 to the random nonce setting, or alternatively, to formally prove the post-quantum universal unforgeability of OCB1 in the random nonce setting. Establishing either of these would formally resolve the question of whether OCB1 is inherently more resistant to quantum universal forgery attacks in the random nonce setting as compared to OCB2(f) and OCB3.

IND-qCPA Security of XTS for Arbitrary-Length Messages. Our proof of IND-qCPA security of OCB2 when used as a pure AE mode (i.e., with empty AD) with uniformly random nonces can be extended to prove the IND-qCPA security of XTS when used as a disk encryption scheme, under the assumption that each sector number is uniformly randomly chosen and that the length of messages is a multiple of the block size of the underlying block-cipher of XTS. We leave it open to extend the analysis of IND-qCPA security of XTS for the setting where the length of messages can be arbitrary, i.e. not necessarily a multiple of the block size of the underlying block-cipher (while still assuming that each sector number is uniformly randomly chosen). We also leave it open to extend the IND-qCPA security analysis of XTS to the setting where sector numbers can be chosen adaptively (and in a non-repeating manner) by an adversary.

Quantum Analysis of OCB Confidentiality beyond IND-qCPA. Note that in this paper, we considered the notion of IND-qCPA security with regards to confidentiality. But there are other stronger quantum security notions in the literature. More recently, [CETU21] showed that the so-called notions of "Left-or-Right" qIND-qCPA security in [GHS16] and "Real-or-Random" RoP-qsCPA security in [MS16] – which are individually strictly stronger than IND-qCPA security – together imply *all* possible quantum IND-CPA security notions.

Now note that our IND-qCPA attacks against the OCB modes in Sections 3 and 4 trivially extend to the above notions of qIND-qCPA and RoP-qsCPA security, as these notions are stronger than IND-qCPA security. However, an interesting open question here would be to analyze the quantum security of the OCB2 mode – without AD – w.r.t. these two notions, either deriving a positive result by possibly extending our IND-qCPA security proof in Subsection 4.2 or a negative result via an efficient attack.

Acknowledgments

It is our pleasure to thank Pratyay Mukherjee for helpful discussions during the early stages of this work. We also thank Xavier Bonnetain and the anonymous reviewers of

ToSC 2022 for their constructive comments and suggestions on earlier drafts of this paper.

References

[AAAS+19] Gorjan Alagic, Gorjan Alagic, Jacob Alperin-Sheriff, Daniel Apon, David Cooper, Quynh Dang, Yi-Kai Liu, Carl Miller, Dustin Moody, Rene Peralta, et al. *Status report on the first round of the NIST post-quantum cryptography standardization process.* US Department of Commerce, National Institute of Standards and Technology âĂę, 2019.

[AASA+20] Gorjan Alagic, Jacob Alperin-Sheriff, Daniel Apon, David Cooper, Quynh Dang, John Kelsey, Yi-Kai Liu, Carl Miller, Dustin Moody, Rene Peralta, et al. Status report on the second round of the nist post-quantum cryptography standardization process. *US Department of Commerce, NIST*, 2020.

[ABF+16] Gorjan Alagic, Anne Broadbent, Bill Fefferman, Tommaso Gagliardoni, Christian Schaffner, and Michael St. Jules. Computational security of quantum encryption. In Anderson C. A. Nascimento and Paulo Barreto, editors, *ICITS 16*, volume 10015 of *LNCS*, pages 47–71. Springer, Heidelberg, August 2016.

[AKC+17] Reza Azarderakhsh, Brian Koziel, Matt Campagna, Brian LaMacchia, Craig Costello, Patrick Longa, Luca De Feo, Michael Naehrig, Basil Hess, Joost Renes, Amir Jalali, Vladimir Soukharev, David Jao, and David Urbanik. Supersingular Isogeny Key Encapsulation, 2017.

[ATTU16] Mayuresh Vivekanand Anand, Ehsan Ebrahimi Targhi, Gelo Noel Tabia, and Dominique Unruh. Post-quantum security of the CBC, CFB, OFB, CTR, and XTS modes of operation. In Tsuyoshi Takagi, editor, *Post-Quantum Cryptography - 7th International Workshop, PQCrypto 2016*, pages 44–63. Springer, Heidelberg, 2016.

[BBC+21] Ritam Bhaumik, Xavier Bonnetain, André Chailloux, Gaëtan Leurent, María Naya-Plasencia, André Schrottenloher, and Yannick Seurin. QCB: efficient quantum-secure authenticated encryption. In Mehdi Tibouchi and Huaxiong Wang, editors, *ASIACRYPT 2021*, volume 13090 of *LNCS*, pages 668–698. Springer, 2021.

[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, Heidelberg, December 2011.

[Ber09] Daniel J Bernstein. Introduction to post-quantum cryptography. In *Post-quantum cryptography*, pages 1–14. Springer, 2009.

[BHN+19] Xavier Bonnetain, Akinori Hosoyamada, María Naya-Plasencia, Yu Sasaki, and André Schrottenloher. Quantum attacks without superposition queries: The offline Simon's algorithm. In Steven D. Galbraith and Shiho Moriai, editors, *ASIACRYPT 2019, Part I*, volume 11921 of *LNCS*, pages 552–583. Springer, Heidelberg, December 2019.

[BJ15] Anne Broadbent and Stacey Jeffery. Quantum homomorphic encryption for circuits of low T-gate complexity. In Rosario Gennaro and Matthew J. B. Robshaw, editors, *CRYPTO 2015, Part II*, volume 9216 of *LNCS*, pages 609–629. Springer, Heidelberg, August 2015.

[BLNS21] Xavier Bonnetain, Gaëtan Leurent, María Naya-Plasencia, and André Schrottenloher. Quantum linearization attacks. In Mehdi Tibouchi and Huaxiong Wang, editors, *ASIACRYPT 2021*, volume 13090 of *LNCS*, pages 422–452. Springer, 2021.

[BN17] Ritam Bhaumik and Mridul Nandi. Improved security for OCB3. In Tsuyoshi Takagi and Thomas Peyrin, editors, *ASIACRYPT 2017, Part II*, volume 10625 of *LNCS*, pages 638–666. Springer, Heidelberg, December 2017.

[BZ13a] Dan Boneh and Mark Zhandry. Quantum-secure message authentication codes. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 592–608. Springer, Heidelberg, May 2013.

[BZ13b] Dan Boneh and Mark Zhandry. Secure signatures and chosen ciphertext security in a quantum computing world. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 361–379. Springer, Heidelberg, August 2013.

[CETU21] Tore Vincent Carstens, Ehsan Ebrahimi, Gelo Noel Tabia, and Dominique Unruh. Relationships between quantum IND-CPA notions. In Kobbi Nissim and Brent Waters, editors, *TCC 2021*, volume 13042 of *LNCS*, pages 240–272. Springer, 2021.

[DDKA21] Mina Doosti, Mahshid Delavar, Elham Kashefi, and Myrto Arapinis. A unified framework for quantum unforgeability. *CoRR*, abs/2103.13994, 2021.

[Deu85] David Deutsch. Quantum theory, the church-turing principle and the universal quantum computer. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 400(1818):97–117, 1985.

[DFNS14] Ivan Damgård, Jakob Funder, Jesper Buus Nielsen, and Louis Salvail. Superposition attacks on cryptographic protocols. In Carles Padró, editor, *ICITS 13*, volume 8317 of *LNCS*, pages 142–161. Springer, Heidelberg, 2014.

[Dwo] Morris Dworkin. Recommendation for block cipher modes of operation: The xts-aes mode for confidentiality on storage devices, national institute of standards and technology. Technical report, Tech. Rep. 800-38E, 2010.[Online].

[fNE13] European Union Agency for Network and Information Security (ENISA). Algorithms, key sizes and parameters report - 2013 recommendations. 2013. https://www.enisa.europa.eu/publications/algorithms-key-sizes-and-parameters-report.

[GHS16] Tommaso Gagliardoni, Andreas Hülsing, and Christian Schaffner. Semantic security and indistinguishability in the quantum world. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part III*, volume 9816 of *LNCS*, pages 60–89. Springer, Heidelberg, August 2016.

[Gro96] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *28th ACM STOC*, pages 212–219. ACM Press, May 1996.

[HS18] Akinori Hosoyamada and Yu Sasaki. Quantum Demiric-Selçuk meet-in-the-middle attacks: Applications to 6-round generic Feistel constructions. In Dario Catalano and Roberto De Prisco, editors, *SCN 18*, volume 11035 of *LNCS*, pages 386–403. Springer, Heidelberg, September 2018.

[IIMP19] Akiko Inoue, Tetsu Iwata, Kazuhiko Minematsu, and Bertram Poettering. Cryptanalysis of OCB2: Attacks on authenticity and confidentiality. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part I*, volume 11692 of *LNCS*, pages 3–31. Springer, Heidelberg, August 2019.

[IM19] Takanori Isobe and Kazuhiko Minematsu. Plaintext recovery attacks against XTS beyond collisions. In Kenneth G. Paterson and Douglas Stebila, editors, *SAC 2019*, volume 11959 of *LNCS*, pages 103–123. Springer, Heidelberg, August 2019.

[KLLN16] Marc Kaplan, Gaëtan Leurent, Anthony Leverrier, and María Naya-Plasencia. Breaking symmetric cryptosystems using quantum period finding. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part II*, volume 9815 of *LNCS*, pages 207–237. Springer, Heidelberg, August 2016.

[KM10] Hidenori Kuwakado and Masakatu Morii. Quantum distinguisher between the 3-round feistel cipher and the random permutation. In *ISIT 2010*, pages 2682–2685. IEEE, 2010.

[KM12] Hidenori Kuwakado and Masakatu Morii. Security on the quantum-type even-mansour cipher. In *ISITA 2012*, pages 312–316. IEEE, 2012.

[KMV17] Louiza Khati, Nicky Mouha, and Damien Vergnaud. Full disk encryption: Bridging theory and practice. In Helena Handschuh, editor, *CT-RSA 2017*, volume 10159 of *LNCS*, pages 241–257. Springer, Heidelberg, February 2017.

[KR11] Ted Krovetz and Phillip Rogaway. The software performance of authenticated-encryption modes. In Antoine Joux, editor, *FSE 2011*, volume 6733 of *LNCS*, pages 306–327. Springer, Heidelberg, February 2011.

[LM08] Moses Liskov and Kazuhiko Minematsu. Comments on XTS-AES. *Comments to NIST*, 2008. https://csrc.nist.gov/CSRC/media/Projects/Block-Cipher-Techniques/documents/BCM/Comments/XTS/XTS_comments-Liskov_Minematsu.pdf.

[MS16] Shahram Mossayebi and Rüdiger Schack. Concrete security against adversaries with quantum superposition access to encryption and decryption oracles. *CoRR*, abs/1609.03780, 2016.

[oEE08] Institute of Electrical and Electronics Engineers. IEEE Std. 1619-2007, IEEE standard for cryptographic protection of data on block-oriented storage devices. 2008.

[RBBK01] Phillip Rogaway, Mihir Bellare, John Black, and Ted Krovetz. OCB: A block-cipher mode of operation for efficient authenticated encryption. In Michael K. Reiter and Pierangela Samarati, editors, *ACM CCS 2001*, pages 196–205. ACM Press, November 2001.

[Reg02] Oded Regev. Quantum computation and lattice problems. In *43rd FOCS*, pages 520–529. IEEE Computer Society Press, November 2002.

[Reg10] Oded Regev. On the complexity of lattice problems with polynomial approximation factors. ISC, pages 475–496. Springer, Heidelberg, 2010.

[Rog04] Phillip Rogaway. Efficient instantiations of tweakable blockciphers and refinements to modes OCB and PMAC. In Pil Joong Lee, editor, *ASIACRYPT 2004*, volume 3329 of *LNCS*, pages 16–31. Springer, Heidelberg, December 2004.

[Rog11] Phillip Rogaway. Evaluation of some blockcipher modes of operation. *Cryptography Research and Evaluation Committees (CRYPTREC) for the Government of Japan*, 2011.

[Sho99] Peter W Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review*, 41(2):303–332, 1999.

[Sim94] Daniel R. Simon. On the power of quantum computation. In *35th FOCS*, pages 116–123. IEEE Computer Society Press, November 1994.

[SS17] Thomas Santoli and Christian Schaffner. Using simon's algorithm to attack symmetric-key cryptographic primitives. *Quantum Inf. Comput.*, 17(1&2):65–78, 2017.

[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, Heidelberg, May 2014.

[Zha12] Mark Zhandry. How to construct quantum random functions. In *53rd FOCS*, pages 679–687. IEEE Computer Society Press, October 2012.