On the Post-Quantum Security of Classical Authenticated Encryption Schemes

Nathalie Lang$^{[0000-0002-2768-9878]}$, Stefan Lucks$^{[0000-0003-4906-5131]}$

Bauhaus-Universität Weimar,
✉ nathalie.lang@uni-weimar.de, stefan.lucks@uni-weimar.de

Abstract. We study the post-quantum security of authenticated encryption (AE) schemes, designed with classical security in mind. Under superposition attacks, many CBC-MAC variants have been broken, and AE modes employing those variants, such as EAX and GCM, thus fail at authenticity. As we show, the same modes are IND-qCPA insecure, i.e., they fail to provide privacy under superposition attacks. However, a constrained version of GCM is IND-qCPA secure, and a nonce-based variant of the CBC-MAC is secure under superposition queries. Further, the combination of classical authenticity and classical chosen-plaintext privacy thwarts attacks with superposition chosen-ciphertext and classical chosen-plaintext queries – a security notion that we refer to as IND-qdCCA. And nonce-based key derivation allows generically turning an IND-qdCCA secure scheme into an IND-qCCA secure scheme.

Keywords: authenticated encryption, post-quantum security

1 Introduction

The advent of quantum computers can be a game-changer for cryptography. This is well-known for current public-key cryptosystems [Sho94], thus the ongoing process to find consensus for new public-key cryptosystems [CJL^{+}16]. The impact of quantum computers on symmetric cryptography is less understood. There are two common attack models for adversaries with a quantum computer: Q1 attacks allow only classical communication between the adversary and the victim (or the "challenger" in attack definitions). In Q2 attacks, the adversary can send queries in superposition to the challenger. Accordingly, the challenger's answers can also be in superposition. By a "common wisdom", symmetric cryptosystems can be made quantum-secure "by doubling the key size". This fails in the Q2 setting, where many symmetric cryptosystems have been broken (e.g., [KLLN16,BLNS21]). Even for the Q1 setting, this "common wisdom" has been disproven: Certain Q1 attacks degrade $2.5k$-bit classical security down to k-bit Q1 security [BSS22,US22].

We focus on the Q2 model. To motivate the practical relevance of the Q2 model, we give an example: Consider sensitive classical data (e.g. medical records), stored in an authenticated and encrypted database. The analysis software connects to a security module, which decrypts selected database entries, returning insensitive data (e.g., by anonymizing the records and aggregating data). The module can be a subroutine running on the same computer as the analysis software, though with different privileges. If this computer is a quantum computer, queries can be in superposition. The module could force queries to be classical, e.g., by performing measurements, but this would put the reason of using a quantum computer for data analysis into question. In our example, the analysis software only asks for the decryption of chosen ciphertexts, but never for the encryption of any plaintexts. This inspires a constrained variant of Q2 attacks: Q2d ("Q2 decrypt") attacks can make chosen-ciphertext queries in superposition, while chosen-plaintext queries are always classical. Note that Q2 security trivially implies Q2d security, Q2d security trivially implies Q1 security, and, for chosen-plaintext attacks, Q2d and Q1 are equivalent.

Focus of this paper. We focus on the Q2 and Q2d security of authenticated encryption (AE), motivated by the following question: *"When exposed to superposition queries, which state-of-the-art AE systems maintain a meaningful level of security?"* A "none" answer to this question would imply a difficult transition from legacy to post-quantum cryptosystems, once superposition queries become an issue. Beyond superposition attacks on MACs, which imply some AE systems failing to provide authenticity [KLLN16], this question has, to the best of our knowledge, only been addressed for the OCB mode [MMPR22], and for a sponge-based AE scheme [JS22]. We will consider other modes, but also describe generic conditions for security under superposition attacks.

Related Work. Certain properties, like the no-cloning theorem, require the reconsideration and revision of classical security notions to the quantum scenario [AGM18b,AGM18a]. Many well-established classical message authentication codes have been found insecure in the Q2 model: Quantum period finding breaks many message authentication codes (MACs), including common variants of CBC-MAC [KLLN16]. (Though, there are also the positive results for HMAC and NMAC [SY17,HI21].) Quantum linearization pushes quantum period finding further, for more attacks [BLNS21]. Authenticated encryption (AE) combines privacy with authenticity, and the above attacks also apply to the authentication aspect of AE modes. [ATTU16] did study the privacy of unauthenticated encryption in the Q2 setting, for chosen-plaintext attacks: Some modes (e.g., counter) are secure in the Q2 sense, even when the underlying block cipher is only Q1 secure. Other modes (e.g., CBC encryption) are secure in the Q2 sense when instantiated with a Q2-secure block cipher. The same paper also describes a wicked block cipher, which we will refer to as $\tilde{E}$, which is a 1PRP but not a qPRP. When instantiated with $\tilde{E}$, CBC encryption is insecure in the Q2 model.

In the current paper, we consider the matching IND-qCPA and IND-qCCA notions from [BZ13b] to model the *privacy aspect* of AE modes. These models assume so-called classical "challenge queries" (two chosen messages, the challenger will encrypt one of them), while "learning queries" (one message which the challenger will encrypt or, in the case of IND-qCCA also decrypt) can be in superposition. [CETU20] proposes stronger security notions for quantum chosen-plaintext security, which also allow to eliminate the distinction between challenge and learning queries. But [CETU20] only deals with chosen-plaintext security, while we will consider both chosen-plaintext and chosen-ciphertext attacks.

We model the *authenticity aspect* of AE modes by the “Plus One” (PO) notion [BZ13a]. We refer to PO security in a Q2 setting as qPO, and to PO security in a Q1 setting as 1PO. The stronger “Blind Unforgeability” (BU) notion [AMRS20] seems to be less natural. Though, we can informally argue that the nonce-prefix MAC, which we analyze in Section 5.1, is BU secure since it behaves like a qPRF, which suffices for BU security. An alternative to “classical modes”, i.e., to modes proposed for classical security, could be new “quantum” modes, such as QCB, a mode for tweakable block ciphers, which has been proven IND-qCPA and qPO secure [BBC$^+$21]. Most of our security proofs are straightforward reductions. We do not require Zhandry’s random oracle recording technique [Zha19]. For some proofs, we we build on the O2H lemma [Unr15,AHU19].

Outline and Contribution

Sections 2 and 3 give preliminaries and sum up known ideas and results.

Section 4 studies the privacy in the IND-qCPA sense, of certain AE modes, which are already known to be qPO insecure [KLLN16,BLNS21]:

(A) The generic SIV mode, GCM mode, and EAX mode are IND-qCPA insecure.

(B) A restricted variant of GCM, which, for a block size of n, only allows nonces of size $n - 32$ bit, is IND-qCPA secure.[1]

In Section 5 we identify two techniques, which have been employed for classical security, but which happen to also defend against superposition attacks:

(C) The *nonce-prefix variant of the CBC-MAC* shares the security properties of CBC encryption described by [ATTU16]: When instantiated with a Q2-secure block cipher, the nonce-prefix MAC is qPO secure, but, when instantiated with the wicked block cipher $\tilde{E}$ from [ATTU16], the MAC is insecure.

(D) *Nonce-based re-keying* defends against superposition chosen-plaintext queries: It turns an IND-1CPA secure AE system into an IND-qCPA secure one.

Section 6 considers generic AE schemes:

(E) If an authenticated encryption scheme is both IND-1CPA secure and 1PO secure, then it is also IND-qdCCA secure.

(F) *Nonce-based re-keying* turns an IND-qdCCA and 1PO secure AE system into an IND-qCCA secure one

[1] This seems to be good news for many practical instantiations of GCM, which often employ $n = 128$ and 96-bit nonces. But the attack from [KLLN16] still applies, i.e., even that variant is qPO insecure.

In Section 7 we conclude.

Note that by combining results (E) and (F) we can construct IND-qCCA secure AE from an AE scheme, which is only Q1 secure.

2 Definitions

2.1 Notation

We refer to security against classical adversaries by the prefix "c", to security against Q1 adversaries by "1", to security against Q2 adversaries by the "q", and to security against Q2d adversaries ("Q2 decrypt") by "qd". E.g., we write "IND-qCPA" for Q2 security in a chosen-plaintext setting, "IND-qdCCA" for Q2d security in a chosen-ciphertext setting, "1PO" for authenticity against Q1 adversaries, and "cPRP" for a classically secure PRP. If "P" is a primitive, C[P] denotes the instantiation of a generic construction "C" by P. If S is a set of primitives, Cr[S] denotes all instantiations C[P] for all P $\in S$. We write $s||t$ for the concatenation of bit-strings $s, t \in \{0,1\}^*$.

Much of our methodological approach is based on reductions. Typically, we assume the existence of a Q2-adversary A_2 against some scheme, and we describe a Q1 adversary A_1 against another scheme, or in a different attack setting. A_1 performs some simple operations to transform superposition queries from A_2 into the classical queries A_1 can make, and to transform the classical responses A_1 receives from its challenger into superposition responses for A_2, and to compute its own final output from A_2's final output. I.e., A_1 is about as efficient as A_2 (though Adv(A_1) and Adv(A_2) may be significantly different). We thus propose the following notation:

Definition 1 **($A_1 \leftarrow_{\mathbf{wrap}} A_2$).** *Consider two adversaries A_1 and A_2. A_2 makes q_2 queries $Q^2_1, \ldots, Q^2_{q_2}$ of total length $\sigma_2 = \sum_{1 \le i \le q_2} |Q^2_i|$ and forwards them to A_1 who makes q_1 queries $Q^1_1, \ldots, Q^1_{q_1}$ of total length $\sigma_1 = \sum_{1 \le i \le q_1} |Q^1_i|$. We write $T(A_x)$ for the running time of A_x. Then A_2 is a wrapper for A_1, written as $A_1 \leftarrow_{wrap} A_2$, if ($\sigma_1 \in O(\sigma_2)$ and $T(A_2) \in T(A_1) + O(\sigma_2)$).*

2.2 Symmetric schemes

Definition 2 **(Encryption).** *Let $\mathcal{K}$ be a finite set of secret keys. An encryption scheme $(\mathcal{E}, \mathcal{D})$ is a pair of two efficient algorithms $\mathcal{E}$ and $\mathcal{D}$, in combination with a soundness property. The encryption algorithm $\mathcal{E}$ takes a key $K \in \mathcal{K}$, a nonce N, a header H, and a message $M \neq \perp$ and generates a ciphertext $C = \mathcal{E}_K(N, H, M)$. The decryption algorithm $\mathcal{D}$ takes a key $K \in \mathcal{K}$, a nonce N, a header H, and a ciphertext C and generates a message $M = \mathcal{D}_K(N, H, C)$. The soundness property*

$$\mathcal{D}_K(N, H, \mathcal{E}_K(N, H, M)) = M$$

holds for all $K \in \mathcal{K}$, all nonces N, all headers H and all messages M.
Encryption schemes are either unauthenticated, i.e., for all $K \in \mathcal{K}$, all nonces N, all headers H, and ciphertext C, $M = D_K(N, H, C)$ is a valid message, or authenticated, i.e., some triples (N, H, C) are invalid and cannot be decrypted. Then, we write $\perp = \mathcal{D}_K(N, H, C)$, understanding $\perp$ as an error message.

Remark: All authenticated encryption schemes we consider in the current paper have a constant-size expansion, i.e., there exists a constant τ such that $|C| = |M| + \tau$ for all K, N, H, and M and $C = E_K(N, H, M)$.

Definition 3 (Message authentication codes (MACs)). *Let $\mathcal{K}$ be a finite set of secret keys. A MAC $\mathcal{M}$ is a deterministic function, which can be implemented by an efficient algorithm. If $\mathcal{M}$ is nonce-based, it takes a key $K \in \mathcal{K}$, a nonce N, and a message M and computes an authentication tag $T = \mathcal{M}_K(N, M)$. If $\mathcal{M}$ is deterministic, it takes a key $K \in \mathcal{K}$ and a message M and computes an authentication tag $T = \mathcal{M}'_K(M)$.*

Now we formalize the notion of privacy under chosen-plaintext attacks:

Definition 4 (The generic IND-CPA and IND-CCA games). *Let $(\mathcal{E}, \mathcal{D})$ denote a nonce-based encryption scheme with keyspace $\mathcal{K}_\mathcal{E}$ and $\mathcal{A}$ an adversary, making q queries.* [2]
The Generic IND-CCA game consists of the following three steps:

Initialize: *The challenger randomly chooses $K \xleftarrow{\$} \mathcal{K}_\mathcal{E}$ and $b \xleftarrow{\$} \{0, 1\}$.*
It maintains a set B of "blocked triples", which is initially empty: $B = \{\}$.

Query Phase: *For $i \in \{1, \ldots, q\}$, $\mathcal{A}$ makes either of the following queries:*

Forward Learning query: *$\mathcal{A}$ chooses a nonce/header/message triple (N_i, H_i, M_i) and receives $C_i = \mathcal{E}_K(N_i, H_i, M_i)$.*
The challenger sets $B = B \cup \{(N_i, H_i, M_i)\}$.

Backward Learning query: *$\mathcal{A}$ chooses a nonce/header/ciphertext triple (N_i, H_i, C_i).*
If $(N_i, H_i, C_i) \in B$, the challenger sends $\perp$ to $\mathcal{A}$.
Else, the challenger sends $\mathcal{D}_K(N_i, H_i, C_i)$ to $\mathcal{A}$.

Challenge query: *$\mathcal{A}$ chooses a nonce N_i, two headers $H_{i,0}$ and $H_{i,1}$ and two messages $M_{i,0}$ and $M_{i,1}$, receives ciphertext $C_i = \mathcal{E}_K(N_i, H_{i,b}, M_{i,b})$.*
The challenger sets $B = B \cup \{(N_i, H_{i,0}, M_{i,0}), (N_i, H_{i,1}, M_{i,1})\}$.

Finalize: *$\mathcal{A}$ outputs a classical bit $b' \in \{0, 1\}$. The event win($\mathcal{A}$) occurs if $b' = b$.*
The ***advantage*** *of $\mathcal{A}$ is*

$$Adv(\mathcal{A}) = |\Pr[b' = 1|b = 1] - \Pr[b' = 1|b = 0]| \qquad (1)$$

[2] $\mathcal{A}$ is constrained to choose unique nonces for forward learning and challenge queries. This will be formalized in Definition 8 below.

The Generic IND-CPA game consists of the same steps, except that $\mathcal{A}$ cannot make any backward learning queries.

In the next step, we formalize the notion of authenticity:

Definition 5 (The generic PO game). *Let $(\mathcal{E}, \mathcal{D})$ denote an encryption scheme (authenticated or not) with key space $\mathcal{K}_{\mathcal{E}}$. In this game, the adversary $\mathcal{A}$ can make q learning queries and no challenge queries.*

Initialize: *The challenger randomly chooses $K \xleftarrow{\$} \mathcal{K}_{\mathcal{E}}$.*

Query Phase: *For $i \in \{1, \dots, q\}$, $\mathcal{A}$ chooses a nonce/header/message triple (N_i, H_i, M_i) as a learning query and receives $C_i = \mathcal{E}_K(N_i, H_i, M_i)$.*

Finalize: *$\mathcal{A}$ outputs $q+1$ distinct classical triples (N'_i, H'_i, C'_i) $(1 \leq i \leq q+1)$ of nonce N'_i, header H'_i and ciphertext C'_i. $\mathcal{A}$ wins if all triples are valid, i.e., if*

$$\forall i \in \{1, \dots, q+1\} : \ \mathcal{D}_K(N'_i, H'_i, C'_i) \neq \perp.$$

*$\mathcal{A}$'s **advantage** is $Adv(\mathcal{A}) = Pr[\mathcal{A} \text{ wins}]$.*

The generic PO game extends naturally to an adversary $\mathcal{A}'$ attacking a deterministic MAC and to an adversary $\mathcal{A}''$ attacking a nonce-based MAC. In the query phase, $\mathcal{A}'$ chooses a message M_i and receives the authentication tag $T_i = MAC_K(M_i)$. Next, $\mathcal{A}''$ chooses a pair (N_i, M_i) of nonce and message and receives $T_i = MAC_K(N_i, M_i)$. Upon finalization, $\mathcal{A}'$ outputs $q+1$ pairs (T'_j, M'_j) and wins if all pairs are valid. $\mathcal{A}''$ outputs $q+1$ triples (T'_j, N'_j, M'_j) and wins if all triples are valid.

Finally, we consider the primitives our modes are built from.

Definition 6 (Generic PRPs/PRFs). *If $\$: \{0,1\}^n \to \{0,1\}^n$ is a permutation over n bit (or $\$: \{0,1\}^n \to \{0,1\}^m$ an n-bit to m-bit function), chosen uniformly at random, and $P : \{0,1\}^n \to \{0,1\}^n$ is another permutation (or $F : \{0,1\}^n \to \{0,1\}^m$ another function), chosen according to some probability distribution, we define the PRP advantage (PRF-advantage) of A by*

$$Adv^{PRP}(A, (P, \$)) = \left| \Pr[A^P = 1] - \Pr[A^\$ = 1] \right|$$

(or $Adv^{PRF}(A(F, \$)) = \left|\Pr[A^F = 1] - \Pr[A^\$ = 1]\right|$).

Note that the definition of the PRP advantage allows A to choose x and query for $P(x)$ or $\$(x)$, respectively, but we do not allow A to query for the inverse permutations of P and \$. I.e., we do not consider "strong PRPs".

2.3 Quantum Attacks and Types of Adversaries

We assume the reader to be familiar with classical and quantum computers, and their distinction.

Definition 7 (Types of queries). *Let f be a function, available by an oracle. In the case of a* classical query*, an adversary chooses x and receives $f(x)$ as the answer. In the case of a* superposition query*, the adversary chooses $|x\rangle |y\rangle$ and receives $|x\rangle |y \oplus f(x)\rangle$ as the answer.*

Definition 8 (Types of adversaries). *A* classical adversary *is running a classical computer and its queries are classical.*

A Q1 adversary *is running a quantum computer but only makes classical queries.*

A Q2 adversary *is running a quantum computer and can make superposition learning queries. More precisely, for a forward learning query (N_i, H_i, M_i), the header H_i and the message M_i can be in superposition, and the nonce N_i is classical.*[3] *For a backward learning query, (N_i, H_i, C_i), the header H_i, the ciphertext C_i and also the nonce N_i can be in superposition.*

Challenge queries are completely classical.

A Q2d adversary *is a Q2 adversary, restricted to classical forward learning queries; only its backward learning queries can be in superposition.*[4]

All adversaries use unique nonces for their challenge and forward learning queries. That is, if for $i \neq j$ neither the i-th nor the j-th query are backward learning queries, and the nonces for either query are N_i and N_j, then $N_i \neq N_j$. [5]

2.4 A Wicked PRP

[ATTU16] proposes a family of permutations over $\{0,1\}^n$, which they prove to be a secure 1PRP, but which they show to be vulnerable under superposition attacks, i.e., it fails to be a secure qPRP. In the current paper, we will refer to this as the "wicked PRP $\tilde{E}$":

Definition 9 (Wicked PRP $\tilde{E}$). *$\tilde{E} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ is a family of permutations $\tilde{E}_k(\cdot)$ over $\{0,1\}^n$. There exist efficiently computable functions $f' : \{0,1\}^n \to \{0,1\}^n, f'' : \{0,1\}^n \to \{0,1\}^n, E' : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^{n-1}$,*

[3] Prohibiting superposition nonces is the established approach in the related work since the nonce, even though we *model* it as chosen by the adversary, is a counter, a timestamp, or a random value generated by the sender's communication machinery.

[4] Similarly, a Q2e ("Q2 encrypt") adversary can make superposition forward learning queries, but only classical backward learning queries. Though, we do not need Q2e adversaries in our context.

[5] $N_i \neq N_j$ is well-defined, even in the Q2 model with forward learning queries in superposition, since the nonces N_i and N_j are always classical.

and $E'' : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$, such that for every key k, $\tilde{E}_k(x)$ can be written as

$$\tilde{E}_k(x) = E'_{f'(k)}(x)||E''_{f''(k)}(x).$$

Furthermore, for all x and k, E' satisfies

$$E'_{f'(k)}(x) = E'_{f'(k)}(x \oplus k).$$

See [ATTU16] for a concrete construction of the wicked PRP, based on an $n-1$-bit PRP and some random oracles. [ATTU16] prove $\tilde{E}$ to be a secure 1PRP. Even though $\tilde{E}$ is a 1PRP, is not a qPRP: The adversary can use Simon's algorithm to find k, and then it is trivial to distinguish $\tilde{E}_k$ from random. Thus, $\tilde{E}$ can be seen as a 1PRP, i.e., secure against Q1-adversaries, but with a built-in backdoor for Q2-adversaries.

3 Known Ideas and Results

We summarize some known ideas and results, which we draw on in subsequent sections: Firstly, we recall Simon's algorithm. Secondly, we consider the security of the Counter- and the CBC-Mode under superposition chosen-plaintext queries [ATTU16]. Thirdly, we consider quantum period finding attacks [KLLN16], using chosen-message queries in superposition to break numerous message authentication codes, such as most variants of the CBC-MAC and MACs based on polynomial hashing. Fourth, we recall quantum linearization attacks [BLNS21], an extension of quantum period finding to break (among other things) beyond-birthday message authentication codes.

3.1 Simon's problem, -subprogram, and -algorithm

Given oracle access to a function $f : \{0,1\}^m \rightarrow \{0,1\}^n$ with $f(x) = f(y) \Leftrightarrow x \oplus y \in \{0, s\}$ for a hidden nonzero "period" $s \in \{0,1\}^m$. Simon's algorithm [Sim97] allows to generically recover s in polynomial time. A classical generic algorithm would require time $\Omega(2^{m/2})$. Here, "generic" means without exploiting any specific property of f, except for the existence of the hidden period s.

The algorithm can be described as running Simon's subprogram $O(m)$ times, and then solving a system of linear equations. Simon's subprogram performs the following steps:

1. Initialize a $2n$-qubit register to a superposition of 2^n values: $2^{-n/2} \sum_x |x\rangle |0\rangle$. (One can do so by initializing the first n qubits to $|0\rangle$ and applying the Hadamard transform to get $2^{-n/2} \sum_x |x\rangle = H^{\oplus n} |0\rangle$.)
2. Call the f-oracle for $2^{-n/2} \sum_x |x\rangle |f(x)\rangle$.
3. Measure the second register. Let v be the result of the measurement. The state in the $2n$-qubit register collapses to a superposition of only two values: $2^{-1/3}(|y\rangle + |y \oplus s\rangle) |v\rangle$.

4. Apply the Hadamard-transform to the first n qubits: $H^{\oplus n}(2^{-1}(|y\rangle + |y+s\rangle))\,|y\rangle$.
5. Measure the first n qubits. This yields a random z with $z \cdot s = 0$. Return z.

Each evaluation of Simon's subprogram implies one superposition query (cf. step 2). Simon's algorithm runs Simon's subprogram until m linearly independent equations $z_i \cdot s = 0$ have been collected. In the algorithm's final step, one computes s by solving this system. Note that with overwhelming probability, it suffices to call Simon's subprogram $O(m)$ times. Below, we will write $T(\text{Simon}(m))$ for the run time required to recover an m-bit secret this way.

[KLLN16] made two observations, which are helpful for the application of Simon's algorithm for cryptanalytic purposes:

1. Assume many independent functions f_1, f_2, all with the same period s (i.e., $f_i(x) = f_i(y) \Leftrightarrow x \oplus y \in \{0, s\}$). Even if each oracle call provides access to another function f_i, one can apply Simon's algorithm to recover s.
2. One can even apply Simon's algorithm with a relaxed version of Simon's problem. Assume a period s, such that $f(x) = f(y) \Leftarrow x \oplus y \in \{0, s\}$ but allow for certain cases of $f(x) = f(y)$ even if $x \oplus y \notin \{0, s\}$. As long as $\max_{t \in \{0,1\}^m/\{0,s\}} \Pr_x[f(x) = f(x \oplus t)]$ is negligible, Simon's algorithm will provide the period s in $O(m)$ queries with overwhelming probability. E.g., this is the case if, apart from the constraining $f(x) = f(x \oplus s)$, f is chosen randomly.

3.2 Counter- and CBC-Mode Under Superposition Queries [ATTU16]

Definition 10 (Stream cipher). *Assume a pseudorandom function F_K, such that for every input N, $F_K(N) \in \{0,1\}^*$ is an infinite random string of bits. We write $F_K^m(N) \in \{0,1\}^m$ for the first m bits of $F_K(N)$. An F-*based stream cipher *takes nonces N_i, a messages M_i and computes ciphertext C_i of length $|C_i| = |M_i|$ as $C_i = F_K^{|M_i|}(N_i) \oplus M_i$.*

Theorem 1 (Similar to Lemma 5 of [ATTU16]). *Assume a PRF-based stream cipher and a Q2-adversary A_2 against the stream cipher. Then a Q1-adversary A_1 and a Q2-adversary A_2 against the same stream cipher exists with $A_1 \leftarrow_{wrap} A_2$ and $Adv(A_1) = Adv(A_2)$.*

The proof for Theorem 1 is essentially the same as the proof given in [ATTU16]. But since [ATTU16] only claim polynomial-time equivalence of A_1 and A_2 and assume a random nonce, we provide the proof in Appendix A .

Definition 11 (Counter mode and CBC). *Assume an n-bit block cipher E and a key K. Given a nonce $N < 2^n$, the* counter mode key stream *is an infinite string $Cnt_K(N) = E_K(N)||E_K(N+1)||E_K(N+2)||\ldots$, where the addition $N+i$ is modulo 2^n.* Counter mode encryption *is the stream cipher based on Cnt_K.*

```
CBC-MAC$_K(M_1, \ldots, M_m)$
$C_0 \leftarrow 0$
for $i \in \{1, \ldots, m\}$ do
    $C_i = E_K(C_{i-1} \oplus M_i)$
return $(C_m)$
```

```
CMAC$_K(M)$
$(M_1, \ldots, M_{m-1}, M_m) \leftarrow$parse$(M)$
$L \leftarrow E_K(0)$
if $|M_m| = n$ then
    $M_m^* \leftarrow 2L \oplus M_m$
else
    $M_m^* \leftarrow 4L \oplus (M_m||1||0^{n-|M_m|-1})$
$T \leftarrow$ CBC-MAC$_K(M_1, \ldots, M_{m-1}, M_m^*)$
return $(T)$
```

Fig. 1: Pseudocode of CBC-MAC[6] and CMAC[7]. Both algorithms receive a key K and a message $(M_1, \ldots, M_m)$. For CBC-MAC it holds that $\forall i$: $M_i \in \{0,1\}^n$.

Given an m-block messsage $M = (M_1, \ldots M_m) \in (\{0,1\}^n)^m$, CBC encryption *consists of two steps: first randomly choose $C_0 \in \{0,1\}^n$, then compute $C_i = E_K(M_i \oplus C_{i-1})$. The ciphertext from CBC encryption is the $(m+1)$-block string $(C_0, C_1, \ldots, C_m)$.*

Counter[E] and CBC[E] are the instantiations of the Counter- and the CBC-mode by E.

Definition 12. *Two nonce-message pairs (N, M) and (N', M') with nonempty messages M and M' are* counter-overlapping, *if $((N \leq N')$ and $(N' - N < |M|/n))$ or $((N \geq N')$ and $(N - N' < |M'|/n))$. Else, they are* counter-overlap free.

According to Theorem 1, *superposition queries fail to provide any benefit at all over classical queries* – for the counter mode:

Theorem 2 (Similar to Theorem 3 of [ATTU16]). *If A_2 is a Q2-adversary on the counter mode, a Q1-adversary $A_1 \leftarrow_{wrap} A_2$ on the counter mode exists with $Adv(A_1) = Adv(A_2)$.*

CBC encryption. For the CBC-mode, *the adversary can benefit greatly from superposition queries*, except when the underlying block cipher is secure against such queries:

Theorem 3 ([ATTU16]). *CBC[$\tilde{E}$] is IND-1PRP secure. CBC[$\tilde{E}$] is IND-qCPA insecure. CBC[qPRP] is IND-qCPA secure.*

Recall that we write $\tilde{E}$ for the wicked PRP from Definition 9.

[6] CBC-MAC by itself is classically insecure when applied to messages where one message is allowed to be a prefix of a longer message. On the other hand, CMAC has been proven secure, assuming the block cipher E to be secure (i.e., a good PRF).

[7] For CMAC, $2L$ and $4L$ are defined as products over GF(2^n), and the "parse" operation splits any nonempty M into n-bit blocks $M_1, \ldots, M_{m-1}$ and one block M_m

```
$\mathrm{EAX}_K(N, H, M)$
$N' \leftarrow \mathrm{CMAC}_K(\langle 0\rangle \mid\mid N)$
$H' \leftarrow \mathrm{CMAC}_K(\langle 1\rangle \mid\mid H)$
$C \leftarrow \mathrm{Ctr}_K(N', M)$
$C' \leftarrow \mathrm{CMAC}_K(\langle 2\rangle \mid\mid C)$
$T \leftarrow N' \oplus H' \oplus C'$
return (T,C)
```

Fig. 2: The EAX mode. It takes a key K, nonce N, header H, and message M as input. By $\langle i\rangle \in \{0,1\}^n$, we denote an initial block encoding the number $i \in \{0,1,2\}$.

3.3 Quantum Period Finding Attacks [KLLN16]

Quantum period finding [KLLN16] applies Simon's algorithm to create forgeries for a variety of messsage authentication codes (MACs). This also breaks the authenticity of AE schemes employing these MACs. We will outline the attacks on the CMAC and GMAC, due to their relevance for the rest of this paper.

The CBC-MAC and its variants. Given the learning queries as the interface, the core idea is to define a function f maintaining Simon's promise, such that finding the period s is useful for the adversary. Kaplan et al [KLLN16] used these to attack the two-key variant of CBC-MAC, but, as they pointed out, the attack applies to many other CBC-MAC variants. Here, we consider the attack on the one-key variant, also dubbed CMAC.

Attacking CMAC [KLLN16]. CMAC is a variant of CBC-MAC see Figure 1. For the attack we assume a string constant $\sigma \in (\{0,1\}^n)^*$, two constants $\beta_0 \neq \beta_1$ in $\{0,1\}^n$, and the function $f : \{0,1\} \times \{0,1\}^n \to \{0,1\}^n$ by

$$f(b,x) = \mathrm{CMAC}_K(\sigma || \beta_b || x) = E_K(x \oplus 2L \oplus E_K(\overbrace{\beta_b \oplus \text{CBC-MAC}_K(\sigma)}^{\alpha_b})).$$

The secret period s is $s = (1 || E_K(\alpha_0) \oplus E_K(\alpha_1))$, since

$$f(0,x) = f(1,y) \Leftrightarrow x \oplus y = E_K(\alpha_0) \oplus E_K(\alpha_1).$$

Assume finding s did require $q' \in \Omega(n)$ learning queries. The adversary now makes $q'+1$ queries for $T_i = \mathrm{CMAC}_K(\sigma || \beta_0 || X_i)$, with distinct X_i. Thanks to the secret period $s = 1 || (E_K(\alpha_0) \oplus E_K(\alpha_1))$ we know $T_i = \mathrm{CMAC}_K(\sigma || \beta_1 || X_i)$. In total, we made $q = 2q'+1$ learning queries and got $q+1 = 2q'+2$ pairs (M_i, T_i), thus winning the qPO game.

Attacking EAX. Consider the EAX mode for authenticated encryption, as depicted in Figure 2. Note that EAX makes three calls to the CMAC. If the message is empty, EAX is essentially a nonce-dependent MAC for the header, and the CMAC attack applies to EAX just as well. Thus, EAX is not qPO secure.

with $|M_m| \in \{1, \ldots, n\}$. If M is the empty string, then parse$(M) = M_1$ with M_1 being an empty block: $|M_1| = 0$.

```
$\mathrm{GHASH}_L(M_1, \ldots, M_m)$
$H \leftarrow \bigoplus_{1 \le i \le m+1} M_i L^{|M|-i+1}$
return $(H)$

$\mathrm{Ctr}_K(J_0, M)$
$(M_1, \ldots, M_m) \leftarrow \mathrm{pad}_0(M)$
for $i \in \{1, \ldots, m\}$ do
    $C_i \leftarrow E_K(J_0 + i) \oplus M_i$
$C \leftarrow \mathrm{trunc}_{|M|}(C_1, \ldots, C_m)$
return $(C)$
```

```
$\mathrm{GCM}_K(N, H, M)$
$L \leftarrow E_K(0)$
if $/N/=n\text{-}32$ then
    $J_0 \leftarrow N||0^{31}||1 \in \{0,1\}^n$
else
    $J_0 \leftarrow$
      $\mathrm{GHash}_L(\mathrm{pad}_0(N)||\langle|N|\rangle)$
$C \leftarrow \mathrm{Ctr}_K(J_0, M)$
$R \leftarrow \mathrm{pad}_0(H)||\mathrm{pad}_0(C)$
$S \leftarrow \mathrm{GHash}_L(R||\langle 2^{n/2}|H| + |M|\rangle)$
$T \leftarrow E_K(J_0) \oplus S$
return $(T, C)$
```

Fig. 3: GHash under key L[8] of a message $(M_1, \ldots, M_m)$ of n-bit blocks, Ctr-mode to encrypt a message M with a start counter $J_0 \in \{0,1\}^n$ and GCM encryption of a message M[9], depending on a key K, nonce N, and a header H. For $S \in \{0,1\}^*$, the operation $\mathrm{pad}_0(S)$ denotes appending the minimum number of 0-bits, such that the length of the result is a multiple of the block size n. We will refer to $E_K(N), E_K(N+1), \ldots$ as the *key stream.*

Attacking GMAC/GCM [KLLN16] GCM and its related algorithms are presented in Figure 3. Unlike EAX, GCM does not employ any variant of CBC-MAC, but rather employs a polynomial hash. When calling GCM with an empty message ϵ, GCM is de facto used as a nonce-based MAC for the header. We will refer to this as $\mathrm{GMAC}_K(N, X) = \mathrm{GCM}_K(N, X, M)$. As it turns out, GMAC, and, by implication, GCM, are not qPO secure. Furthermore, the attack is essentially the same as for CMAC: define $\sigma \in /\{0,1\}^n)^*$, $\beta_0 \neq \beta_1$ in $\{0,1\}^n$ and the $f : \{0,1\} \times \{0,1\}^n \rightarrow \{0,1\}^n$ by

$$f_N(b, x) = \mathrm{CMAC}_K(N, \sigma||\beta_b||x). \tag{2}$$

The secret period $s \in \{0,1\}^{n+1}$ is

$$s \;=\; 1 \;||\; (L\beta_0 \oplus L\beta_1) \;=\; 1 \;||\; (L(\beta_0 \oplus \beta_1)). \tag{3}$$

Since s does not depend on the nonce N, one can apply Simon's algorithm exactly as in the case of CMAC: "As for the CBC-MAC, repeating these two steps leads to an existential forgery attack" [KLLN16].

3.4 Quantum Linearization for Beyond-Birthday MACs [BLNS21]

We continue by giving examples for quantum linearization attacks. A deeper understanding of such attacks is not necessary for the reader. However, for further

[8] L^i denotes exponentiation in $\mathrm{GF}(2^n)$ and $M_i L^i$ is a product in $\mathrm{GF}(2^n)$.

[9] The length of messages is restricted to less than 2^{32} blocks.

```
$\mathrm{GenericSIV}_{K,L}(N,H,M)$
$V \leftarrow \mathrm{MAC}_L(I(N,H,M))$    $C \leftarrow \mathrm{Enc}_K(V,M)$,    **return** $(V,C)$
```

Fig. 4: Pseudocode of the generic SIV encryption, combining a MAC and a nonce-based encryption operation Enc, using two independent keys K and L. $I(N,H,M)$ is an input encoding of a nonce N, header H, and message M, which satisfies $I(N,H,M) \neq I(N',H',M')$ if $(N,H,M) \neq (N',H',M')$. Given (V,C), authenticated SIV decryption first computes $M = \mathrm{Enc}_K^{-1}(V,C)$, and then $V' = \mathrm{MAC}_L(I(N,H,M))$. If $V = V'$, authenticated SIV decryption returns M, else it rejects (V,C).

reading we refer to [BLNS21]. Typical Variants of CBC-MAC (e.g. CMAC), and typical MACs based on polynomial hashing (e.g. GMAC), only provide classical security up to the birthday bound of $2^{n/2}$, where n is the block size. Coincidently, many beyond-birthday MACs (against classical adversaries) seem to be save from a straightforward application of period finding. [BLNS21] extended period finding to linearization. At the cost of increasing the size of each query from $\Theta(1)$ blocks to $\Theta(n)$ blocks, quantum linearization allows breaking several beyond-birthday MACs (and also many other schemes).

Attacking the GCM-SIV2 and the GCM-SIV2-MAC. SIV is a mode of operation to perform "deterministic authenticated encryption", i.e., authenticated encryption with minimal damage when nonces are reused [RS06]. Generic SIV is depicted in Figure 4. Many instantiations of generic SIV have been proposed by different authors. GCM-SIV2 is a beyond-birthday-secure instantiation, combining a beyond-birthday secure MAC with a beyond-birthday secure variant of the counter mode [IM16]. The MAC, which we refer to as GCM-SIV2-MAC, takes two n-bit keys L_1, L_2, four block cipher keys K'_1, K'_2, K'_3, and K'_4, a nonce N, a header $H = (H_1, \ldots, H_j)$, and a message $M(M_1, \ldots, M_m)$ and computes a $2n$-bit authentication tag

$$(T_1, T_2) = \text{GCM-SIV2-MAC}_{L_1,L_2,K'_1,K'_2,K'_3,K'_4}(N,H,M).$$

After parsing the joint input H and M as a sequence $(X_1, \ldots, X_{a+m+1})$ of $a+m+1$ n-bit blocks, with the last block $(\langle|H|\rangle||\langle|M|\rangle)$ holding encodings of the lengths of H and M, GCM-SIV2-MAC computes intermediate values

$$V_1 = N \oplus \bigoplus_{1 \le i \le a+m+1} L_1^i X_i \quad \text{and} \quad V_2 = N \oplus \bigoplus_{1 \le i \le a+m+1} L_2^i X_i$$

and then returns

$$T_1 = E_{K'_1}(V_1) \oplus E_{K'_3}(V_2) \quad \text{and} \quad T_2 = E_{K'_2}(V_1) \oplus E_{K'_4}(V_2)$$

Plain quantum period finding fails for the GCM-SIV2-MAC, but quantum linearization succeeds [BLNS21]. To attack the MAC, one generates pairs $(N,H,M) \neq (N,H',M')$ with the same authentication tag

$$T = \text{GCM-SIV2-MAC}_K(N,H,M) = \text{GCM-SIV2-MAC}_K(N,H',M').$$

This not only allows to break the qPO security of the MAC but also of GCM-SIV2 itself: For any such pair, ask for $(C, T) = \text{GCM-SIV2}_K(N, H, M)$. Then decrypting (N, C', H', T) with $C' = C \oplus M \oplus M'$ will return M'.

4 Privacy under Q2 Attacks (or Lack Theorof)

As we argued in Section 3, AE modes, such as EAX, GCM, and GCM-SIV2, directly inherit qPO insecurity from the MAC schemes they are based on. Thus authenticity is lost – but could any of these AE modes still preserve privacy, at least in the IND-qCPA sense?

4.1 GCM-SIV2

Recall the generic SIV mode from Figure 4. It uses the result of the MAC operation as the nonce for the encryption operation Enc. Thus, when instantiated with any of the MACs studied in Section 3, the adversary can force a nonce-reuse for Enc. One such instantiation is GCM-SIV2, which furthermore instantiates Enc by the counter mode. Using the counter mode with the same nonce is insecure. Thus, GCM-SIV2 is IND-qCPA insecure.

4.2 GCM

For GCM, it is easy to recover the internal secret $L = E_K(0)$ from Equation 3, since β_0, β_1, and i are known to the adversary, and the computations are in $\text{GF}(2^n)$). *But can we exploit knowing L for a qCPA attack?* The answer depends on the nonce length. Observe that GCM treats nonces N of size $|N| = n - 32$ differently from different-sized nonces, cf. lines 2–5 in GCM (Figure 3). The case $\text{GCM}_{|N|=n-32}$ will be considered below. But, *GCM without restrictions on the nonce-space is not IND-qCPA secure – not even if the block cipher is a qPRP*. If $|N| \neq n - 32$, the initial counter J_0 is derived by calling GHash: $J_0 = \text{GHash}_L(\text{pad}_0(N)||\langle|N|\rangle)$. Knowing L allows us to create overlapping counters efficiently. We present two approaches.

Our first approach employs two nonces N and N', with $|N'| = n - 32 \neq |N|$. We do not need unplausibly long nonces; as a reasonable choice, we set $|N| = n$. Fix N' and $J_0 = N'||0^{31}||1$. Now the adversary just has to compute N with

$$\text{GHash}_L(N||\langle n\rangle) = L^2 N \oplus L\langle n\rangle = J_0,$$

i.e., compute $N = (J_0 \oplus L\langle n\rangle) * L^{-2}$ in $\text{GF}(2^n)$. A nonce-respecting adversary uses either N or N' in a learning query, and the other one in the challenge query. Both nonces generate the same J_0 and thus the same key stream.

Our second approach even works for nonces $N \neq N'$ of equal size $|N| = |N'| = n$. We choose N and N' such that $J'_0 = J_0 + 1$. This creates a counter overlap.

Namely, we fix N', compute J'_0 and then solve the equation below for N:

$$\overbrace{L^2N \oplus L\langle n\rangle}^{J_0} = \overbrace{(L^2N' \oplus L\langle n\rangle)}^{J'_0} - 1.$$

Due to $J'_0 = J_0 + 1$, the keystreams overlap, except for the first output block from the J_0 key stream.

GCM$_{|N|=n-32}$*, a Restricted Variant of GCM.* If we restrict all nonces N to $|N| = n-32$, the initial counter is set to $J_0 = N||0^{31}||1$, without invoking GHash. As with all variants of GCM, GCM$_{|N|=n-32}$ fails at authenticity, i.e., is qPO insecure. But the restriction $|N| = n-32$ preserves privacy: Due to the message length restriction, all nonce-respecting queries are overlap-free. According to Theorem 1, GCM$_{|N|=n-32}$ is IND-qCPA secure if the block cipher is a 1PRP, even when the adversary knows L.

4.3 EAX

Reconsider CMAC. We aim to recover the key-dependent secret $L = E_K(0)$. Recall lines 3–6 of Algorithm 2. Consider an arbitrary n_1-bit string X and set $X_1 = X||1 \in n\{0,1\}^n$. If $M_m = X$, then $M^*_m = X_1 \oplus 4L$. If $M_m = X_1$, then $M^*_m = X_1 \oplus 2L$. For the attack, we thus assume learning queries with a superposition of two identical messages, except for M_m being either X or X_1.[10] We continue with the attack to recover $6L$ (and by implication: L). Note that this attack is specific to CMAC, and would not apply to most other CBC-MAC variants:

1. Guess $\beta = \mathrm{LSB}(6L)$.
2. Define the function $f_\beta : \{0,1\} \times \{0,1\}^{n-1} \to \{0,1\}^n$ by
$$f_\beta(b,x) = \begin{cases} \mathrm{CMAC}_K(x) & \text{if } b{=}0 \\ \mathrm{CMAC}_K(x||\beta) & \text{else} \end{cases}.$$
3. Note that if $\beta = \mathrm{LSB}(6L)$, then f is periodic: $f_\beta(0||x) = f_\beta(1||x \oplus \mathrm{MSB}_{n-1}(6L)$. Apply Simon's algorithm to recover the period $s = \mathrm{MSB}_{n-1}(6L)$.
4. If Simon's algorithm fails, replace β by $1-\beta$ and go to step 2. Else return $6L = (s||\beta)$

For MAC forgeries, there seems to be no benefit over the attack from Section 3.3. But knowing L allows us to mount the following simple IND-qCPA attack:

[10] The design of a quantum interface for the superposition of messages of different lengths may not be obvious. For concreteness, assume a maximum message length μ, and a message of length $m = |M| \leq \mu$ is encoded as a $(\mu + \log_2(\mu))$-qubit string $|M\rangle\,|0^{\mu-m}\rangle\,|m\rangle$ of message, padding and message length.

1. Choose an n-bit nonce $N_0 = L \oplus 2L$, a $2n$-bit nonce $N_1 = (L||N_0)$, a header H and two messages $M_0 \neq M_1$ with $|M_0| = |M_1|$.
2. Make $\begin{cases} \text{a learning query for } (T_1, C_1) \ = \text{EAX}_K(N_0, H, M_0), \text{ and} \\ \text{a challenge query for } (T_2, C_2) = \text{EAX}_K(N_1, H, M_b) \text{ for unknown } b. \end{cases}$
3. If $C_1 = C_2$ then return 0; else return 1.

As it turns out, this attack succeeds with advantage 1. Namely, we generate the same keystream in either query and thus, if $b = 0$ then $C_1 = C_2$. Also, if $b = 1$ then $C_1 = C_2 \oplus M_1 \oplus M_2 \neq C_1$. To verify this claim, observe

$$\begin{aligned} \text{CMAC}_K(\langle 0\rangle||N_0) &= \text{CMAC}_K(\langle 0\rangle||(L \oplus 2L)) \\ &= E_K(E_K(0) \oplus L \oplus 2L \oplus 2L) \qquad = E_K(0) = L \end{aligned}$$

and

$$\begin{aligned} \text{CMAC}_K(\langle 0\rangle||L||N_0) &= \text{CMAC}_K(\langle 0\rangle||L||(L \oplus 2L)) \\ &= E_K(E_K(E_K(0) \oplus L) \oplus L \oplus 2L \oplus 2L) \\ &= E_K(E_K(0) \oplus L \oplus 2L \oplus 2L) \qquad = E_K(0) = L. \end{aligned}$$

Note that this attack does not work for nonces of equal sizes.

5 Accidential Protection from Q2 Attacks

All AE modes we studied in the context of the current research have been designed and proposed with classical security in the mind. Thus, one must not blame the modes' authors for insecurities under quantum attacks. But we found some design aspects in some of the modes we studied, which happen to protect against Q2 attacks.

5.1 The Nonce-Prefix MAC from the CCM Mode

CCM [WHF03] is an early AE mode. Though criticized for a variety of practical undesirabilities [RW03], CCM did inspire the evolution of improved AE modes ("improved" from a classical point of view), such as EAX. We focus on CCM-MAC, which applies CBC-MAC to $(N, |M|, M)$, where N is the nonce, $|M|$ is the message length, and M is the message itself. The original CCM-MAC did allow to squeeze the nonce N of length $|N| \ll n$ and the encoding of $|M|$ into the first input block for the CBC-MAC, while we propose NP-MAC, where the nonce $N \in \{0,1\}^n$ fits exactly into the first block, see Figure 5. We tried to find possible attacks on NP-MAC by using quantum period finding and quantum linearization. Not only did we not find any feasible attacks, but instead we could even prove that those two techniques would not work in attacking NP-MAC. As we will show below, it is secure when instantiated with a qPRP and used with random nonces, but insecure when instantiated with the wicked 1PRP $\tilde{E}$ (cf. Definition 9), just like CBC encryption.

```
NP-MAC_K(M)
choose nonce $N \in \{0,1\}^n$ at random
$T \leftarrow$ CBC-MAC$_K(N, |M|, \mathrm{pad}_0(M))$
return $(T)$
```

Fig. 5: The NP-MAC with a full-block nonce $N \in \{0,1\}^n$; for simplicity, we assume the encoding of the message length $|M| < 2^n$ to be a full-block value $|M| \in \{0,1\}^n$. We write NP-MAC[E] for the instantiation of NP-MAC with a block cipher E.

Theorem 4. *For NP-MAC[$\tilde{E}$] a Q2-Adversary A exists, which recovers the secret key. The accumulated length of all queries from A is $O(n^2)$, and $T(A) \in O(T(Simon(n)))$.*

Proof. Assume a single-block-message $M = \sum_{x\in\{0,1\}^n} |x\rangle$ as the superposition $M = M_1 = \sum_{x\in\{0,1\}^n} |x\rangle$. Note that $|M| = n$ and $\mathrm{pad}_0(M) = M$. Every learning query will be of the form (N_i, M) with a unique nonce N_i and M in superposition[11]. The challenger responds $T = \text{CBC-MAC}_K(N_i, |M|, M)$, also in superposition. The first two blocks N_i and $|M|$ are classical values. Thus, $\gamma = \text{CBC-MAC}_k(N_i, |M|)$ is a classical constant. The final block is $M = \sum_{x\in\{0,1\}^n} |x\rangle$. Thus, the authentication tag is

$$T = \sum_{x\in\{0,1\}^n} |x\rangle\, |f_{k,N_i}(M)\rangle = \sum_{x\in\{0,1\}^n} |x\rangle\, |\tilde{E}_k(x \oplus \overbrace{\text{CBC-MAC}_k(N_i, |M|)}^{\gamma}\,)\rangle \qquad (4)$$

$$= \sum_{x\in\{0,1\}^n} |x\rangle\, |\tilde{E}_k(x)\rangle\,. \qquad (5)$$

Recall the definition of $\tilde{E}_k$ in Definition 9: The first $n-1$ bits of $\tilde{E}$ form a subfunction $E'_{f(k)}$, with period k, i.e., $E'_{f(k)}(x) = E'_{f(k)}(y)$ if and only if $x = y$ or $x = y \oplus k$. Thus, the first $n-1$ bits of f_{k,N_i} also form a function with the same period. This allows a Q2 adversary to recover the secret key k by running Simon's algorithm. The adversary makes $O(n)$ queries, each of length n, so the total query length is $O(n^2)$.[12]

Corollary 1. *NP-MAC[$\tilde{E}$] is qPO insecure.*

Theorem 5. *Let E be a qPRF. For every adversary A_2, distinguishing NP-MAC[E] from random, an adversary $A_1 \leftarrow_{wrap} A_2$ exists, which distinguishes CBC[E_K] from random with $Adv(A_1) = Adv(A_2)$.*

[11] We would like to point out that there is no need for N_i to be in superposition since the attack already works for classical nonces.

[12] Here, the query length is counted in bits.

Proof. Let N_i be the random nonce for i-th chosen message $|M_i\rangle$ of length m_i[13] from A_2. For each such $|M_i\rangle$, A_1 creates its own chosen message $M_i' = (0^n \,||\, m_i \,||\, |M_i\rangle)$ and requests the CBC encryption of M_i' from the challenger, using the nonce N_i. A_1 ignores the answer to its query, except for the final block, C_{i,m_i}, which it returns to A_2. Observe $C_{i,m_i} = \text{CBC-MAC}_K(N_i, |M_i\rangle)$. Thus, A_2 receives exactly the answer it expects. $A_1 \leftarrow_{\text{wrap}} A_2$, since A_1 extends each query from A_2 by prepending just two blocks, and otherwise, A_1 performs no additional work beyond invoking A_2 and the challenger. Similarly, A_1 succeeds if and only if A_2 succeeds, hence $\text{Adv}(A_1) = \text{Adv}(A_2)$.

Corollary 2. *NP-MAC[qPRP] is qPO secure, and a quantum-secure MAC.*

We do not claim NP-MAC[qPRP] to *be* a qPRF. NP-MAC is not even a function, as its output depends on the random nonce N chosen in the first step as presented in Figure 5. But once N has been set, NP-MAC is a function, and if N is chosen at random, the output is indistinguishable from random. In that sense, NP-MAC[qPRP] can be regarded as a "weak qPRF". Here, the next input block, after the nonce, does not actually need to be an encoding of the message length $|M|$. For qPO security, we just need classical security (and, of course, the random nonce as the first block). For the classical security of NP-MAC, any prefix-free encoding of M suffices [Jon02], and $|M| \,||\, M$ is such an encoding.

5.2 Key Derivation, as in AES-GCM-SIV

AES-GCM-SIV is another instantiation of the generic SIV principle (cf. Figure 4). It follows the approach of nonce-based key derivation, see Figure 6. The goal is to improve the concrete security of GCM-SIV against classical attacks, while maintaining much of the original performance of GCM-SIV. (This is in contrast to GCM-SIV2, which improves the security of GCM-SIV at the performance costs of running GCM-SIV twice.)

```
KD-Enc_K(N, H, M)
  $K_N \leftarrow \text{KD}_K(N)$
  $(T, C) \leftarrow \text{AEnc}_{K_N}(N, H, M)$
  return $(T, C)$
```

```
KD-Dec_K(N, H, T, C)
  $K_N \leftarrow \text{KD}_K(N)$
  $M \leftarrow \text{AEnc}_{K_N}(N, H, T, C)$
  return $(M)$
```

Fig. 6: Nonce-based key derivation scheme KD-Enc taking as input a nonce N, header H, and message M. It is based on an authenticated encryption scheme AEnc and a random function KD to derive the temporary key K_N from N and K. We write KD-Enc[F^{KD}, E^{AE}] for the instantiation of the KD-Enc scheme with a key derivation scheme F^{KD} and an authenticated encryption scheme E^{AE}.

[13] The chosen messages $|M_i\rangle$ can be in superposition, but all messages $|M_i\rangle$ in superposition are of the same length m_i

Chosen-plaintext queries in the IND-qCPA attack games must use different classical nonces. Nonce-based key derivation thus forces a fresh temporary key for every chosen-plaintext query. This has two implications. The first implication is that we get Q2 security w.r.t. chosen-plaintext attacks, from Q1 security:

Theorem 6. *Let E be a block cipher and assume a Q2-adversary A_2 attacking the IND-qCPA security of KD-Enc[E, AEnc]. Then a Q1-adversary A_{1PRF} attacking the 1PRF security of KD and a Q1-adversary $A_{IND\text{-}1CPA}$ attacking the IND-1CPA security of AEnc exist, such that*

$$\begin{aligned} &A_{1PRF} \leftarrow_{wrap} A_2, \ \textit{and} \ A_{IND\text{-}1CPA} \leftarrow_{wrap} A_2, \ \textit{and} \\ &Adv(A_2) \leq 2Adv(A_{1PRF}) + Adv(A_{IND\text{-}1CPA}). \end{aligned}$$

Proof. Let A_2 be given. Define games G_{A_2,A_2}, $G_{\mathrm{KD,KD}}$, $G_{\$,\mathrm{KD}}$, and $G_{\$,A_2}$. Game G_{A_2,A_2} is plainly running A_2: $\mathrm{Adv}(A_2) = |\Pr[A_2 = 1|b = 1] - \Pr[A_2 = 1|b = 0]|$. The other games are defined in Figure 7. From an adversarial point of view, the games G_{A_2,A_2} and $G_{\mathrm{KD,KD}}$ are identical, just that G_{A_2,A_2} employs the challenger's secret key while $G_{\mathrm{KD,KD}}$ uses its on key K', thus

$$\begin{aligned} &|\Pr[\mathrm{Exp}(G_{A_2,A_2}, 0) = 1] - \Pr[\mathrm{Exp}(G_{\mathrm{KD,KD}}, 0) = 1]| \\ &\quad = |\Pr[\mathrm{Exp}(G_{A_2,A_2}, 1) = 1] - \Pr[\mathrm{Exp}(G_{\mathrm{KD,KD}}, 1) = 1]| = 0. \end{aligned} \tag{6}$$

Similarly to the argument for Equation 6, the games $G_{\$,\mathrm{KD}}$ and $G_{\$,A_2}$ are identical, and thus:

$$\begin{aligned} &|\Pr[\mathrm{Exp}(G_{\$,\mathrm{KD}}, 0) = 1] - \Pr[\mathrm{Exp}(G_{\$,A_2}, 0) = 1]| \\ &\quad = |\Pr[\mathrm{Exp}(G_{\$,\mathrm{KD}}, 1) = 1] - \Pr[\mathrm{Exp}(G_{\$,A_2}, 1) = 1]| = 0. \end{aligned}$$

Next, we define the adversaries A_{1PRF} and $\mathcal{A}_{\mathrm{IND\text{-}1CPA}}$

- A_{1PRF} randomly chooses $b \in \{0, 1\}$ and runs either $\mathrm{Exp}(G_{\mathrm{KD,KD}}, b)$ or $\mathrm{Exp}(G_{\$,\mathrm{KD}}, b)$. The advantage of A_{1PRF} satisfies

$$\begin{aligned} 2 \cdot \mathrm{Adv}(A_{\mathrm{1PRF}}) \leq \ &\left|\Pr[\mathrm{Exp}(G_{\$,\mathrm{KD}}, 0) = 1] - \Pr[\mathrm{Exp}(G_{\mathrm{KD,KD}}, 0) = 1]\right| \\ &+ \left|\Pr[\mathrm{Exp}(G_{\$,\mathrm{KD}}, 1) = 1] - \Pr[\mathrm{Exp}(G_{\mathrm{KD,KD}}, 1) = 1]\right|. \end{aligned}$$

- $\mathcal{A}_{\mathrm{IND\text{-}1CPA}}$ runs $\mathrm{Exp}(G_{\$,A_2}, b)$ for a b chosen by the challenger and unknown to $\mathcal{A}_{\mathrm{IND\text{-}1CPA}}$. The resulting advantage is

$$\mathrm{Adv}(A_{\mathrm{IND\text{-}1CPA}}) = \left|\Pr[\mathrm{Exp}(G_{\$,A_2}, 0) = 1] - \Pr[\mathrm{Exp}(G_{\$,A_2}, 1) = 1]\right|.$$

Only learning queries can be in superposition, and all of the games $G_{\mathrm{KD,KD}}$, $G_{\$,\mathrm{KD}}$ and $G_{\$,A_2}$ compute their response to learning queries on their own, without invoking an oracle. Thus, both A_{1PRF} and $\mathcal{A}_{\mathrm{IND\text{-}1CPA}}$ are Q1 adversaries. Also, all of these games are wrappers around A_2, thus $A_{\mathrm{1PRF}} \leftarrow_{\mathrm{wrap}} A_2$, and $A_{\mathrm{IND\text{-}1CPA}} \leftarrow_{\mathrm{wrap}} A_2$. Finally, for the claimed bound on the advantage of A:

```
Exp($G, b$)
if $G = G_{KD,KD}$ or $G = G_{\$,KD}$ then
    $K' \xleftarrow{\$} \{0,1\}^k$
for $i \in 1 \ldots p(n)$ do
    Perform either Learn(G) or Challenge(G,b)
Receive guess $b'$ from $A_2$
return $b'$
```

```
Learn($G$)
Receive $(N_i, H_i, M_i)$ from $A_2$
if $G = G_{KD,KD}$ then
    $K_i \leftarrow \mathrm{KD}_{K'}(N_i)$
    $C_i \leftarrow \mathrm{AEnc}_{K_i}(N_i, H_i, M_i)$
else
    $R_i \xleftarrow{\$} \{0,1\}^k$
    $C_i \leftarrow \mathrm{AEnc}_{R_i}(N_i, H_i, M_i)$
Send $C_i$ to $A_2$
```

```
Challenge($G, b$)
Receive $(N_i, H_{i,0}, H_{i,1}, M_{i,0}, M_{i,1})$ from $A_2$
if $G = G_{\$,A_2}$ then
    Forward query to challenger
    Receive $C_{i,b}$ from challenger
else
    $K_i \leftarrow \mathrm{KD}_{K'}(N_i)$
    $C_{i,b} \leftarrow \mathrm{AEnc}_{K_i}(N_i, H_{i,b}, M_{i,b})$
Send $C_{i,b}$ to $A_2$
```

Fig. 7: Experiment run by the adversary. G can be $G_{\mathrm{KD,KD}}$, $G_{\$,\mathrm{KD}}$, or $G_{\$,A_2}$. Note that here we do not consider $G = G_{A_2,A_2}$.

$$
\begin{aligned}
\mathrm{Adv}(A_2) &= |\Pr[\mathrm{Exp}(G_{A_2,A_2}, 0) = 1] - \Pr[\mathrm{Exp}(G_{A_2,A_2}, 1) = 1]| \\
&\leq \quad |\Pr[\mathrm{Exp}(G_{A_2,A_2}, 0) = 1] - \Pr[\mathrm{Exp}(G_{\mathrm{KD,KD}}, 0) = 1]| \\
&\quad + \big|\Pr[\mathrm{Exp}(G_{\mathrm{KD,KD}}, 0) = 1] - \Pr[\mathrm{Exp}(G_{\$,\mathrm{KD}}, 0) = 1]\big| \\
&\quad + \big|\Pr[\mathrm{Exp}(G_{\$,\mathrm{KD}}, 0) = 1] - \Pr[\mathrm{Exp}(G_{\$,A_2}, 0) = 1]\big| \\
&\quad + \big|\Pr[\mathrm{Exp}(G_{\$,A_2}, 0) = 1] - \Pr[\mathrm{Exp}(G_{\$,A_2}, 1) = 1]\big| \qquad (7) \\
&\quad + \big|\Pr[\mathrm{Exp}(G_{\$,A_2}, 1) = 1] - \Pr[\mathrm{Exp}(G_{\$,\mathrm{KD}}, 1) = 1]\big| \\
&\quad + \big|\Pr[\mathrm{Exp}(G_{\$,\mathrm{KD}}, 1) = 1] - \Pr[\mathrm{Exp}(G_{\mathrm{KD,KD}}, 1) = 1]\big| \\
&\quad + |\Pr[\mathrm{Exp}(G_{\mathrm{KD,KD}}, 1) = 1] - \Pr[\mathrm{Exp}(G_{A_2,A_2}, 1) = 1]| \\
&\leq 4 \cdot 0 + 2 \cdot \mathrm{Adv}(A_{\mathrm{1PRF}}) + \mathrm{Adv}(A_{\mathrm{IND\text{-}1CPA}})
\end{aligned}
$$

Corollary 3. *If AEnc is IND-1CPA secure, KD-Enc[1PRP, AEnc] is IND-qCPA secure.*

6 Generic Approaches for Q2d and Q2 Security

We describe how to turn Q1-secure AE schemes into Q2d-secure and even proper Q2-secure AE schemes. The definition of the IND-qdCCA security stems from the generic IND-CCA game (cf. Definition 4) and that of a Q2d adversary (cf. Definition 8). Below, Subsection 6.1 provides an intuitive and informal overview over core ideas. Subsections 6.2 to 6.4 are more formal and technical. Subsection 6.5 briefly discusses the tightness of the concrete results.

6.1 Intuition

Consider a Q1-secure AE scheme AEnc. Assume that AEnc provides both chosen-plaintext privacy (1CPA) and authenticity (1PO) when all queries are classical. We claim that in such case, AEnc is also Q2d-secure. For instance, it provides privacy even when decryption queries are in superposition. To prove this claim, assume an adversary making decryption queries (N_i, H_i, C_i) in superposition. Imagine to measure one such query. The measurement result (N_i', H_i', C_i') can be old (i.e., C_i' stems from a matching encryption query (N_i', H_i', M_i') for some M_i') or invalid (decryption returns $\perp$). If there is a non-negligible probability for (N_i, H_i, C_i) in superposition to be found neither old nor invalid after a potential measurement, then we refer to (N_i, H_i, C_i) as *pivotal*. Based on this notion, we distinguish two cases: *(1) The adversary makes no pivotal decryption queries.* Then, the ability to choose decryption queries in superposition, or to choose decryption queries at all, does not provide any significant advantage over making only classical encryption queries. A successful attack in this scenario would thus violate 1CPA security. *(2) The adversary makes at least one pivotal decryption query.* By guessing the index of a pivotal query and actually measuring it, one can generate a forgery. A successful attack in this scenario would thus violate 1PO security. The proof of Theorem 7 centers around this idea.

We also claim that key derivation with a 1PRF secure key derivation function can turn a Q2d chosen-ciphertext secure and 1PO secure scheme AEnc into a fully Q2 chosen-ciphertext secure scheme. Thus, if AEnc is secure against adversaries making decryption queries in superposition, then the derived scheme is also secure against adversaries making all queries in superposition. This is the intuitution for Theorem 8. For the proof, recall the proof of Theorem 6: The main key is only used to generate nonce-dependent derived keys, and each chosen plaintext is encrypted under another derived key. In this case, it turns out that the benefit from choosing plaintexts in superposition is negligible.

By combining both results, one can create a proper Q2-secure AE scheme (encryption and decryption queries in superposition) from a (seemingly much weaker) Q1-secure AE scheme and a 1PRF.

6.2 The O2H (“One-Way to Hiding”) Lemma

To analyze the approach to gather Q2d and Q2 security from Q1 security, we first explain the semi-classical one-way to hiding (O2H) lemma from [AHU19], an improved version of the classical O2H lemma from [Unr15]. At its core are so-called punctured oracles. Consider a subset S from the set of inputs to the oracle H. $H \backslash S$ (H punctured by S) takes a value x as input and computes whether $x \in S$ or not. The event “Find” denotes the case this measurement returns that $x \in S$ is true. When Find does not occur, the outcome of $A^{H/S}$ is independent of $H(x)$ for $x \in S$.

Lemma 1 (Semi-Classical O2H [AHU19]). *Let $S \subset X$ be random. Let $G, H : X \to Y$ be random functions satisfying $\forall x \notin S$. $G(x) = H(x)$. Let z be a* *random bitstring. (S, G, H, z may have arbitrary joint distribution.) Let A be an oracle algorithm of query depth d (not necessary unitary). Let*

$$P_{\mathit{left}} := \Pr[b = 1 : b \leftarrow A^H(z)],\ \ P_{\mathit{right}} := \Pr[b = 1 : b \leftarrow A^G(z)]$$

$$P_{\mathit{find}} := \Pr[\mathit{Find} : A^{G \setminus S}(z)] = \Pr[\mathit{Find} : A^{H \setminus S}(z)]$$

Then

$$|P_{\mathit{left}} - P_{\mathit{right}}| \leq 2\sqrt{(d+1) \cdot P_{\mathit{find}}}, \tag{8}$$

$$|\sqrt{P_{\mathit{left}}} - \sqrt{P_{\mathit{right}}}| \leq 2\sqrt{(d+1) \cdot P_{\mathit{find}}}. \tag{9}$$

This lemma needs to be contextualized. Firstly, the notion "depth d" considers an adversary to perform multiple queries in parallel. In our context, it suffices to point out that $d \leq q$ holds for every q-query adversary.

Secondly, Relationship 8 results in better bounds for our purpose. We deal with Relationship 9 in Appendix B . Thus, Relationship 8 can be rewritten as

$$P_{\text{left}} \leq 2\sqrt{(d+1) \cdot P_{\text{find}}} + P_{\text{right}}. \tag{10}$$

Thirdly, in the context of that paper, $\Pr[b = 1 : b \leftarrow A^H(z)]$ in the notation of [AHU19] is the same as $\text{win}(A) = \Pr[b = b']$ in our notation, where b' is the output from A and b the challenger's internal choice (cf. Definition 4 and Equation 1). As we define the advantage of an adversary by $\text{Adv}(A) = |\Pr[b' = 1|b = 1] - \Pr[b' = 1|b = 0]|$, we have

$$\text{win}(A) = (\text{Adv}(A) + 1)/2.$$

6.3 From Q1 Security to Q2d Security

Theorem 7. *Let AEnc be an AE scheme producing an n-bit output and assume a Q2d-adversary A_{qdCCA} attacking the IND-qdCCA security of AEnc. Then a Q1-adversary A_{1PO} attacking the 1PO security and a Q1-adversary A_{1CPA} attacking the IND-1CPA security of AEnc exists, such that*

$$A_{1PO} \leftarrow_{wrap} A_{qdCCA} \quad \text{and} \quad A_{1CPA} \leftarrow_{wrap} A_{qdCCA} \quad \text{and}$$

$$Adv(A_{qdCCA}) \leq 4\sqrt{(q+1)}\sqrt{Adv(A_{1PO})} + Adv(A_{1CPA}).$$

Proof. Consider an IND-qdCCA adversary $\mathcal{A}$, and define the H oracle: H is identical to the Q2d challenger, responding to all queries from $\mathcal{A}$, including backward learning queries in superposition. I.e., A_{qdCCA}, when connected to its challenger, can be written as $\mathcal{A}^H$.

Define S as the set of those backward learning queries, which return a valid message: $S = (N_i, H_i, C_i) : \text{ADec}_K(N_i, H_i, C_i) \neq \perp$. Recall that the set B of "blocked" triples from Definition 4 includes the triples (N_i, H_i, C_i) known from forward learning queries, and, any backward query trying to decrypt a "blocked" triple returns $\bot$. Thus, whenever the event Find occurs, there is a new triple (N_i, H_i, C_i), which has not been known from any other queries. This allows an adversary to win the PO game by measuring $(N_i, H_i, C_i) \notin B$ but $(N_i, H_i, C_i) \in S$. A PO forgery consists of the new triple (N_i, H_i, C_i) and all the known triples. This defines $A_{1\mathrm{PO}}$. Obviously, $A_{1\mathrm{PO}} \leftarrow_{\mathrm{wrap}} A_{\mathrm{qdCCA}}$ and $\mathrm{Adv}(A_{1\mathrm{PO}}) = P_{\mathrm{Find}}$.

Let G be an oracle, which relays challenge and forward learning queries to a Q1 challenger, and which responds to all backward learning queries by returning $\bot$. Clearly, $H\backslash S$ and $G\backslash S$ behave identically. We write $A_{1\mathrm{CPA}}$ as $\mathcal{A}^G$. Obviously $A_{1\mathrm{CPA}} \leftarrow_{\mathrm{wrap}} A_{\mathrm{qdCCA}}$.

We still have to prove the bound

$$\mathrm{Adv}(A_{\mathrm{qdCCA}}) \leq 2\sqrt{(q+1)}\sqrt{\mathrm{Adv}(A_{1\mathrm{PO}})} + \mathrm{Adv}(A_{1\mathrm{CPA}}).$$

If $\mathrm{Adv}(A_{1\mathrm{CPA}}) \geq \mathrm{Adv}(A_{\mathrm{qdCCA}})$, this is trivial. For the rest of this proof, we assume $\mathrm{Adv}(A_{1\mathrm{CPA}}) < \mathrm{Adv}(A_{\mathrm{qdCCA}})$. In our context, $P_{\mathrm{left}} = \mathrm{win}(A_{\mathrm{qdCCA}})$, $P_{\mathrm{right}} = \mathrm{win}(A_{1\mathrm{CPA}})$, and $P_{\mathrm{find}} = A_{1\mathrm{PO}}$.

Equation 8 (or rather, the derived Equation 10) implies

$$\mathrm{win}(A_{\mathrm{qdCCA}}) \leq 2\sqrt{(q+1)}\sqrt{\mathrm{Adv}(A_{1\mathrm{PO}})} + \mathrm{win}(A_{1\mathrm{CPA}}).$$

We apply $\mathrm{win}(A) = (\mathrm{Adv}(A) + 1)/2$ and simplify the expression to

$$\mathrm{Adv}(A_{\mathrm{qdCCA}}) \leq 4\sqrt{(q+1)}\sqrt{\mathrm{Adv}(A_{1\mathrm{PO}})} + \mathrm{Adv}(A_{1\mathrm{CPA}}).$$

Corollary 4. *If AEnc is both IND-1CPA secure and 1PO secure, then AEnc is also IND-qdCCA secure.*

6.4 Transitioning Q2d Security into Q2 Security

Theorem 8. *Let E be a block cipher and assume a q-query Q2-adversary A_2 attacking the IND-qCCA security of KD-Enc[E, AEnc]. Then a Q1-adversary A_{1PRF} attacking the 1PRF security of KD, a Q2d-adversary $A_{Ind\text{-}qdCCA}$ attacking the IND-qdCCA security of AEnc and a Q1-adversary A_{1PO} exists, attacking the 1PO security of AEnc, such that*

$$A_{1PRF} \leftarrow_{wrap} A_2, \text{ and } A_{IND\text{-}qdCCA} \leftarrow_{wrap} A_2, \text{ and } A_{1PO} \leftarrow_{wrap} A_2, \text{ and}$$
$$Adv(A_2) \leq 2\sqrt{(q+1)\cdot(2\cdot Adv(A_{1PRF}) + Adv(A_{qdCCA}) + 4\cdot Adv(A_{1PO}))}.$$

Proof. The proof resembles the proof of Thm. 6. First, we need to redefine the games introduced in Figure 7 by introducing a modified Experiment Exp′ which also includes backwards learning queries. We handle those queries in a cheating manner such that for all games, the adversary runs $\mathcal{A}_2^{\text{cheat}}$ but answers all backwards learning queries with $\perp$. G_{A_2,A_2} now becomes $G_{A_2,\perp,A_2}$ with

$$\text{Adv}(A_2^{\text{cheat}}) = |Pr[\text{Exp}'(G_{A_2,\perp,A_2},0)=1] - \Pr[\text{Exp}'(G_{A_2,\perp,A_2},1)=1]|.$$

Furthermore, we add the event *Find* previously defined in Lemma 1. Figure 8 at page 32 shows the pseudocode of the new Experiment Exp′.

Similar to before, the following equalities hold:

$$\begin{aligned} &|\Pr[\text{Exp}'(G_{A_2,\perp,A_2},0)=1] - \Pr[\text{Exp}'(G_{\text{KD},\perp,\text{KD}},0)=1]| \\ &\quad = |\Pr[\text{Exp}'(G_{A_2,\perp,A_2},1)=1] - \Pr[\text{Exp}'(G_{\text{KD},\perp,\text{KD}},1)=1]| = 0. \end{aligned} \tag{11}$$

$$\begin{aligned} &|\Pr[\text{Exp}'(G_{\$,\perp,A_2},0)=1] - \Pr[\text{Exp}'(G_{\$,\perp,\text{KD}},0)=1]| \\ &\quad = |\Pr[\text{Exp}'(G_{\$,\perp,A_2},1)=1] - \Pr[\text{Exp}'(G_{\$,\perp,\text{KD}},1)=1]| = 0. \end{aligned} \tag{12}$$

Furthermore, for the adversary, the advantages when running the games $G_{A_2,\perp,A_2}$ and $G_{A_2,\perp,\text{Find}}$ and the games $G_{\text{KD},\perp,\text{KD}}$ and $G_{\text{KD},\perp,\text{Find}}$ are identical since we assume that the hardness of distinguishing the ciphertexts and triggering the event *Find* is equivalent. Thus,

$$\begin{aligned} &|\Pr[\text{Exp}'(G_{A_2,\perp,A_2},0)=1] - \Pr[\text{Exp}'(G_{A_2,\perp,\text{Find}},0)=1]| \\ &\quad = |\Pr[\text{Exp}'(G_{A_2,\perp,A_2},1)=1] - \Pr[\text{Exp}'(G_{A_2,\perp,\text{Find}},1)=1]| = 0. \end{aligned} \tag{13}$$

$$\begin{aligned} &|\Pr[\text{Exp}'(G_{\text{KD},\perp,\text{KD}},0)=1] - \Pr[\text{Exp}'(G_{\text{KD},\perp,\text{Find}},0)=1]| \\ &\quad = |\Pr[\text{Exp}'(G_{\text{KD},\perp,\text{KD}},1)=1] - \Pr[\text{Exp}'(G_{\text{KD},\perp,\text{Find}},1)=1]| = 0. \end{aligned} \tag{14}$$

We continue by defining the adversaries $A_{1\text{PRF}}$, A_{qdCCA} and $A_{1\text{PO}}$.

- $A_{1\text{PRF}}$ randomly chooses $b \in \{0,1\}$ and runs either $\text{Exp}'(G_{\text{KD},\perp,\text{KD}},b)$ or $\text{Exp}'(G_{\$,\perp,\text{KD}},b)$. The advantage of $A_{1\text{PRF}}$ can be bounded by

$$\begin{aligned} 2\cdot\text{Adv}(A_{1\text{PRF}}) \leq \; & \left|\Pr[\text{Exp}'(G_{\$,\perp,\text{KD}},0)=1] - \Pr[\text{Exp}'(G_{\text{KD},\perp,\text{KD}},0)=1]\right| \\ & + \left|\Pr[\text{Exp}'(G_{\$,\perp,\text{KD}},1)=1] - \Pr[\text{Exp}'(G_{\text{KD},\perp,\text{KD}},1)=1]\right|. \end{aligned}$$

- $A_{1\text{PO}}$ randomly chooses $b \in \{0,1\}$ and runs either $\text{Exp}'(G_{A_2,\perp,\text{Find}},b)$, $\text{Exp}'(G_{\$,\perp,\text{Find}},b)$ or $\text{Exp}'(G_{\text{KD},\perp,\text{Find}},b)$. Note that b is not explicitly needed in those games. It is mainly used to improve the notation. The resulting advantage is

$$\begin{aligned} 2\cdot\text{Adv}(A_{1\text{PO}}) \leq \; & \left|\Pr[\text{Exp}'(G_{A_2,\perp,\text{Find}},b)=1] - \Pr[\text{Exp}'(G_{\$,\perp,\text{Find}},b)=1]\right| \\ & + \left|\Pr[\text{Exp}'(G_{\$,\perp,\text{Find}},b)=1] - \Pr[\text{Exp}'(G_{\text{KD},\perp,\text{Find}},b)=1]\right|. \end{aligned}$$

- A_{qdCCA} runs $\mathrm{Exp}'(G_{\$,\bot,A_2}, b)$ for a b chosen by the challenger and unknown to A_{qdCCA}. This results in the following advantage:

$$\mathrm{Adv}(A_{\mathrm{qdCCA}}) = \left|\Pr[\mathrm{Exp}'(G_{\$,\bot,A_2}, 0) = 1] - \Pr[\mathrm{Exp}'(G_{\$,\bot,A_2}, 1) = 1]\right| .$$

Like before, only learning queries can be in superposition and all of the games are wrappers around A_2^{cheat}, such that $A_{\mathrm{1PRF}} \leftarrow_{\mathrm{wrap}} A_2^{\mathrm{cheat}}$, $A_{\mathrm{qdCCA}} \leftarrow_{\mathrm{wrap}} A_2^{\mathrm{cheat}}$ and $A_{\mathrm{1PO}} \leftarrow_{\mathrm{wrap}} A_2^{\mathrm{cheat}}$. Now we are able to bound the advantage of A_2^{cheat}:

$$\begin{aligned}
\mathrm{Adv}(A_2^{\mathrm{cheat}}) = &\left|\Pr[\mathrm{Exp}'(G_{A_2,\bot,A_2}, 0) = 1] - \Pr[\mathrm{Exp}'(G_{A_2,\bot,A_2}, 1) = 1]\right| \\
\leq &\quad \left|\Pr[\mathrm{Exp}'(G_{A_2,\bot,A_2}, 0) = 1] - \Pr[\mathrm{Exp}'(G_{A_2,\bot,\mathrm{Find}}, 0) = 1]\right| \\
&+ \left|\Pr[\mathrm{Exp}'(G_{A_2,\bot,\mathrm{Find}}, 0) = 1] - \Pr[\mathrm{Exp}'(G_{\$,\bot,\mathrm{Find}}, 0) = 1]\right| \\
&+ \left|\Pr[\mathrm{Exp}'(G_{\$,\bot,\mathrm{Find}}, 0) = 1] - \Pr[\mathrm{Exp}'(G_{\mathrm{KD},\bot,\mathrm{Find}}, 0) = 1]\right| \\
&+ \left|\Pr[\mathrm{Exp}'(G_{\mathrm{KD},\bot,\mathrm{Find}}, 0) = 1] - \Pr[\mathrm{Exp}'(G_{\mathrm{KD},\bot,\mathrm{KD}}, 0) = 1]\right| \\
&+ \left|\Pr[\mathrm{Exp}'(G_{\mathrm{KD},\bot,\mathrm{KD}}, 0) = 1] - \Pr[\mathrm{Exp}'(G_{\$,\bot,\mathrm{KD}}, 0) = 1]\right| \\
&+ \left|\Pr[\mathrm{Exp}'(G_{\$,\bot,\mathrm{KD}}, 0) = 1] - \Pr[\mathrm{Exp}'(G_{\$,\bot,A_2}, 0) = 1]\right| \\
&+ \left|\Pr[\mathrm{Exp}'(G_{\$,\bot,A_2}, 0) = 1] - \Pr[\mathrm{Exp}'(G_{\$,\bot,A_2}, 1) = 1]\right| \\
&+ \left|\Pr[\mathrm{Exp}'(G_{\$,\bot,A_2}, 1) = 1] - \Pr[\mathrm{Exp}'(G_{\$,\bot,\mathrm{KD}}, 1) = 1]\right| \\
&+ \left|\Pr[\mathrm{Exp}'(G_{\$,\bot,\mathrm{KD}}, 1) = 1] - \Pr[\mathrm{Exp}'(G_{\mathrm{KD},\bot,\mathrm{KD}}, 1) = 1]\right| \\
&+ \left|\Pr[\mathrm{Exp}'(G_{\mathrm{KD},\bot,\mathrm{KD}}, 1) = 1] - \Pr[\mathrm{Exp}'(G_{\mathrm{KD},\bot,\mathrm{Find}}, 1) = 1]\right| \\
&+ \left|\Pr[\mathrm{Exp}'(G_{\mathrm{KD},\bot,\mathrm{Find}}, 1) = 1] - \Pr[\mathrm{Exp}'(G_{\$,\bot,\mathrm{Find}}, 1) = 1]\right| \\
&+ \left|\Pr[\mathrm{Exp}'(G_{\$,\bot,\mathrm{Find}}, 1) = 1] - \Pr[\mathrm{Exp}'(G_{A_2,\bot,\mathrm{Find}}, 1) = 1]\right| \\
&+ \left|\Pr[\mathrm{Exp}'(G_{A_2,\bot,\mathrm{Find}}, 1) = 1] - \Pr[\mathrm{Exp}'(G_{A_2,\bot,A_2}, 1) = 1]\right| \\
\leq &\ 6 \cdot 0 + 2 \cdot \mathrm{Adv}(A_{\mathrm{1PRF}}) + \mathrm{Adv}(A_{\mathrm{qdCCA}}) + 4 \cdot \mathrm{Adv}(A_{\mathrm{1PO}})
\end{aligned} \tag{15}$$

We can apply the semi-classical O2H Lemma (see Lemma 1) for bounding the actual advantage of A_2. It is easy to see that A_2^{cheat} corresponds to P_{find} from the Lemma. Thus, we conclude that for q queries

$$\begin{aligned}
\mathrm{Adv}(A_2) &\leq 2\sqrt{(q+1) \cdot \mathrm{Adv}(A_2^{\mathrm{cheat}})} \\
&\leq 2\sqrt{(q+1) \cdot (2 \cdot \mathrm{Adv}(A_{\mathrm{1PRF}}) + \mathrm{Adv}(A_{\mathrm{qdCCA}}) + 4 \cdot \mathrm{Adv}(A_{\mathrm{1PO}}))}.
\end{aligned}$$

Corollary 5. *If AEnc is IND-qdCCA and 1PO secure, then KD-Enc[1PRP, AEnc] is IND-qCCA secure.*

6.5 On the Tightness of the Reductions

As abstract results, Theorems 7 and 8 and Corollaries 4 and 5 are *very encouraging:* Classical security concerning quantum adversaries (i.e. 1PO and IND-1CPA security) suffices to achieve security even when chosen-ciphertext queries can be in superposition (i.e., IND-qdCCA security).

But, while the other reductions in the current paper are tight, the connection between the required 1PO and IND-1CPA security level and the IND-q(d)CCA security level granted by Theorems 7 and 8 is not. We provide a numerical example for IND-qdCCA (Theorem 7). Consider an AE scheme with a key size of 256 bit, and adversaries restricted to iterate Grover's algorithm about 2^{80} times. The probability to recover the secret key would thus be about $2^{2*80-256} = 2^{-96}$. Assume that the best 1PO or IND-1CPA attack is equivalent to key recovery, i.e., $\mathrm{Adv}(A_{\mathrm{1PO}}) = \mathrm{Adv}(A_{\mathrm{1CPA}}) = 2^{-96}$. Under these assumptions, and for $q < 2^{20}$, the bound from Theorem 7 is

$$\begin{aligned}\mathrm{Adv}(A_{\mathrm{qdCCA}}) &\leq 4 \cdot \sqrt{(q+1)}\sqrt{\mathrm{Adv}(A_{\mathrm{1PO}})} + \mathrm{Adv}(A_{\mathrm{1CPA}}) \\ &\leq 4 \cdot 2^{10} 2^{-48} + 2^{-96} && \approx 2^{-36}.\end{aligned}$$

If we increase the limit on the number of queries to, say, $q < 2^{50}$, the same calculation gives the bound $\mathrm{Adv}(A_{\mathrm{qdCCA}}) \leq 4 \cdot 2^{25} 2^{-48} + 2^{-96} \approx 2^{-21}$. This should still be fine for most practical purposes. Nevertheless, given the ultra-strong bounds of 2^{-96} each for the 1PO and the IND-1CPA advantage, the bounds on $\mathrm{Adv}(A_{\mathrm{qdCCA}})$ may be surprising.

7 Final Remarks

Recall our motivating question: *"When exposed to superposition queries, which state-of-the-art AE systems maintain a meaningful level of security?"* The first answers are negative: EAX, GCM, and variants of SIV do not only fail at authenticity (as known before), but they also fail at privacy under chosen-plaintext queries. We conclude that all of those modes fail to provide any meaningful level of security under superposition queries. Other answers are positive: A restricted variant of GCM avoids the vulnerability to superposition chosen-plaintext attacks. The nonce-prefix MAC is secure under superposition attacks, and can thus be used as a building block for superposition-resistant AE systems. Theorem 7 provides a path from chosen-plaintext privacy and authenticity in the Q1 model (IND-1CPA and 1PO security) to chosen-ciphertext security in the Q2d model (IND-qdCCA). If we also consider Theorem 8 (i.e., if we apply nonce-based key-derivation), the path leads to chosen-ciphertext privacy in the unrestricted Q2 model (IND-qCCA). To some degree, Theorems 7 and 8 resemble results from [BN00], which provide a generic path from classical chosen-plaintext privacy and classical authenticity to classical chosen-ciphertext privacy.

Based on our findings, new questions arise: (1) By definition, the nonce-prefix MAC requires random nonces. *Could a variant of the nonce-prefix MAC, with nonces chosen by a nonce-respecting adversary, also be qPO secure?* (2) The concrete bounds for IND-q(d)CCA security from Theorems 7 and 8 are a bit unsatisfactory. *Is there a matching attack? Or can one improve the concrete* *bounds?* An answer to either question might require different methods than ours, or, perhaps, a stronger variant of the O2H Lemma.

References

AGM18a. Gorjan Alagic, Tommaso Gagliardoni, and Christian Majenz. Can you sign a quantum state? *CoRR*, abs/1811.11858, 2018.

AGM18b. Gorjan Alagic, Tommaso Gagliardoni, and Christian Majenz. Unforgeable quantum encryption. In Jesper Buus Nielsen and Vincent Rijmen, editors, *Advances in Cryptology - EUROCRYPT 2018 - 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tel Aviv, Israel, April 29 - May 3, 2018 Proceedings, Part III*, volume 10822 of *Lecture Notes in Computer Science*, pages 489–519. Springer, 2018.

AHU19. Andris Ambainis, Mike Hamburg, and Dominique Unruh. Quantum security proofs using semi-classical oracles. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part II*, volume 11693 of *Lecture Notes in Computer Science*, pages 269–295. Springer, 2019.

AMRS20. Gorjan Alagic, Christian Majenz, Alexander Russell, and Fang Song. Quantum-access-secure message authentication via blind-unforgeability. In Anne Canteaut and Yuval Ishai, editors, *Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part III*, volume 12107 of *Lecture Notes in Computer Science*, pages 788–817. Springer, 2020.

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, Fukuoka, Japan, February 24-26, 2016, Proceedings*, volume 9606 of *Lecture Notes in Computer Science*, pages 44–63. Springer, 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, *Advances in Cryptology - ASIACRYPT 2021 - 27th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 6-10, 2021, Proceedings, Part I*, volume 13090 of *Lecture Notes in Computer Science*, pages 668–698. Springer, 2021.

BLNS21. Xavier Bonnetain, Gaëtan Leurent, María Naya-Plasencia, and André Schrottenloher. Quantum linearization attacks. In Mehdi Tibouchi and Huaxiong Wang, editors, *Advances in Cryptology - ASIACRYPT 2021 - 27th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 6-10, 2021, Proceedings, Part I*, volume 13090 of *Lecture Notes in Computer Science*, pages 422–452. Springer, 2021.

BN00. Mihir Bellare and Chanathip Namprempre. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In Tatsuaki Okamoto, editor, *Advances in Cryptology - ASIACRYPT 2000,* *6th International Conference on the Theory and Application of Cryptology and Information Security, Kyoto, Japan, December 3-7, 2000, Proceedings*, volume 1976 of *Lecture Notes in Computer Science*, pages 531–545. Springer, 2000.

BSS22. Xavier Bonnetain, André Schrottenloher, and Ferdinand Sibleyras. Beyond quadratic speedups in quantum attacks on symmetric schemes. In Orr Dunkelman and Stefan Dziembowski, editors, *Advances in Cryptology - EUROCRYPT 2022 - 41st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Trondheim, Norway, May 30 - June 3, 2022, Proceedings, Part III*, volume 13277 of *Lecture Notes in Computer Science*, pages 315–344. Springer, 2022.

BZ13a. Dan Boneh and Mark Zhandry. Quantum-secure message authentication codes. In Thomas Johansson and Phong Q. Nguyen, editors, *Advances in Cryptology - EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings*, volume 7881 of *Lecture Notes in Computer Science*, pages 592–608. Springer, 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, *Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part II*, volume 8043 of *Lecture Notes in Computer Science*, pages 361–379. Springer, 2013.

CETU20. Tore Vincent Carstens, Ehsan Ebrahimi, Gelo Noel Tabia, and Dominique Unruh. On quantum indistinguishability under chosen plaintext attack. *IACR Cryptol. ePrint Arch.*, page 596, 2020.

CJL$^+$16. Lily Chen, Stephen Jordan, Yi-Kai Liu, Dustin Moody, Rene Peralta, Ray Perlner, and Daniel Smith-Tone. Breaking the quadratic barrier: Quantum cryptanalysis of milenage, telecommunications' cryptographic backbone, 2016.

HI21. Akinori Hosoyamada and Tetsu Iwata. On tight quantum security of HMAC and NMAC in the quantum random oracle model. In Tal Malkin and Chris Peikert, editors, *Advances in Cryptology - CRYPTO 2021 - 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16-20, 2021, Proceedings, Part I*, volume 12825 of *Lecture Notes in Computer Science*, pages 585–615. Springer, 2021.

IM16. Tetsu Iwata and Kazuhiko Minematsu. Stronger security variants of GCM-SIV. *IACR Trans. Symmetric Cryptol.*, 2016(1):134–157, 2016.

Jon02. Jakob Jonsson. On the security of CTR + CBC-MAC. In Kaisa Nyberg and Howard M. Heys, editors, *Selected Areas in Cryptography, 9th Annual International Workshop, SAC 2002, St. John's, Newfoundland, Canada, August 15-16, 2002. Revised Papers*, volume 2595 of *Lecture Notes in Computer Science*, pages 76–93. Springer, 2002.

JS22. Christian Janson and Patrick Struck. Sponge-based authenticated encryption: Security against quantum attackers. *IACR Cryptol. ePrint Arch.*, page 139, 2022.

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, *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference,* *Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II*, volume 9815 of *Lecture Notes in Computer Science*, pages 207–237. Springer, 2016.

MMPR22. Varun Maram, Daniel Masny, Sikhar Patranabis, and Srinivasan Raghuraman. On the quantum security of OCB. *IACR Cryptol. ePrint Arch.*, page 699, 2022.

RS06. Phillip Rogaway and Thomas Shrimpton. A provable-security treatment of the key-wrap problem. In Serge Vaudenay, editor, *Advances in Cryptology - EUROCRYPT 2006, 25th Annual International Conference on the Theory and Applications of Cryptographic Techniques, St. Petersburg, Russia, May 28 - June 1, 2006, Proceedings*, volume 4004 of *Lecture Notes in Computer Science*, pages 373–390. Springer, 2006.

RW03. Phillip Rogaway and David A. Wagner. A critique of CCM. *IACR Cryptol. ePrint Arch.*, page 70, 2003.

Sho94. Peter W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *35th Annual Symposium on Foundations of Computer Science, Santa Fe, New Mexico, USA, 20-22 November 1994*, pages 124–134. IEEE Computer Society, 1994.

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

SY17. Fang Song and Aaram Yun. Quantum security of NMAC and related constructions - PRF domain extension against quantum attacks. In Jonathan Katz and Hovav Shacham, editors, *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part II*, volume 10402 of *Lecture Notes in Computer Science*, pages 283–309. Springer, 2017.

Unr15. Dominique Unruh. Revocable quantum timed-release encryption. *J. ACM*, 62(6):49:1–49:76, 2015.

US22. Vincent Ulitzsch and Jean-Pierre Seifert. iarr eprint 2022/733, 2022.

WHF03. Doug Whiting, Russell Housley, and Niels Ferguson. Counter with CBC-MAC (CCM). *RFC*, 3610:1–26, 2003.

Zha19. Mark Zhandry. How to record quantum queries, and applications to quantum indifferentiability. In Alexandra Boldyreva and Daniele Micciancio, editors, *Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part II*, volume 11693 of *Lecture Notes in Computer Science*, pages 239–268. Springer, 2019.

A Proof of Theorem 1

To simplify reading this paper, we first restate the theorem:

Theorem 1 (Similar to Lemma 5 of [ATTU16]) Assume a PRF-based stream cipher and a Q2-adversary A_2 against the stream cipher. Then a Q1-adversary A_1 and a Q2-adversary A_2 against the same stream cipher exists with $A_1 \leftarrow_{\text{wrap}} A_2$ and $\text{Adv}(A_1) = \text{Adv}(A_2)$.

Proof (Proof (Thm. 1).). Assume a Q1-challenger having secretly chosen a key K and a bit b. Also, assume the existence of A_2. A Q2 learning query from A_2

consists of a classical nonce N_1 and a $2|M_i|$-qubit quantum state $|M_i\rangle\,|C_i\rangle$. Since the Q1-challenger cannot deal with a Q2 query, A_1 performs the following steps for each learning query from A_2:

1. request $S_i \leftarrow \mathrm{PRF}_K^{|M_i|}(N_i) \oplus 0^{|M_i|}$ as the encryption of $0^{|M_i|}$ under K,
2. update the quantum register from $|M_i\rangle\,|C_i\rangle$ to $|M_i\rangle\,|C_i \oplus S_i \oplus M_i\rangle$,
3. and return the quantum register to A_2.

Clearly, A_1 and A_2 are tightly equivalent, and when A_2 outputs a bit b', A_1 also outputs b', thus $\mathrm{Adv}(A_1) = \mathrm{Adv}(A_2)$.

B The Bound for Relationship 9

In Section 6 we focused on Relationship 8. We continue by presenting the same proof concerning Relationship 9.

If we assume $P_{\text{left}} > P_{\text{right}}$ Relationship 9 can be rewriten as

$$P_{\text{left}} \leq 4 \cdot (d+1) \cdot P_{\text{find}} + 4\sqrt{(d+1) \cdot P_{\text{find}}}\sqrt{P_{\text{right}}} + P_{\text{right}}. \tag{16}$$

Theorem 9. *Let AEnc be an AE scheme producing an n-bit output and assume a Q2d-adversary A_{qdCCA} attacking the IND-qdCCA security of AEnc. Then a Q1-adversary A_{1PO} attacking the 1PO security and a Q1-adversary A_{1CPA} attacking the IND-1CPA security of AEnc exists, such that*

$$A_{1PO} \leftarrow_{wrap} A_{qdCCA} \quad \text{and} \quad A_{1CPA} \leftarrow_{wrap} A_{qdCCA} \quad \text{and}$$

$$\begin{aligned} Adv(A_{qdCCA}) \leq{}& 8((q+1) \cdot Adv(A_{1PO})) + 8\sqrt{(q+1) \cdot Adv(A_{1PO})}\sqrt{2}\sqrt{Adv(A_{1CPA}) + 1} \\ &+ Adv(A_{1CPA}). \end{aligned}$$

Proof. Consider an IND-qdCCA adversary $\mathcal{A}$, and define the H oracle: H is identical to the Q2d challenger, responding to all queries from $\mathcal{A}$, including backward learning queries in superposition. I.e., A_{qdCCA}, when connected to its challenger can be written as $\mathcal{A}^H$.

Define S as the set of those backward learning queries, which return a valid message: $S = (N_i, H_i, C_i) : \mathrm{ADec}_K(N_i, H_i, C_i) \neq \perp$. Recall that the set B of "blocked" triples from Definition 4 includes the triples (N_i, H_i, C_i) known from forward learning queries, and, any backward query trying to decrypt a "blocked" triple returns $\perp$. Thus, whenever the event Find occurs, there is a new triple (N_i, H_i, C_i), which has not been known from any other queries. This allows an adversary to win the PO game by measuring $(N_i, H_i, C_i) \notin B$ but $(N_i, H_i, C_i) \in S$. A PO forgery consists of the new triple (N_i, H_i, C_i) and all the known triples. This defines $A_{1\text{PO}}$. Obviously $A_{1\text{PO}} \leftarrow_{\text{wrap}} A_{\text{qdCCA}}$ and $\mathrm{Adv}(A_{1\text{PO}}) = P_{\text{Find}}$.

Let G be an oracle, which relays challenge and forward learning queries to a Q1 challenger, and which responds to all backward learning queries by returning $\perp$. Clearly, $H \backslash S$ and $G \backslash S$ behave identically. We write A_{1CPA} as $\mathcal{A}^G$. Obviously $A_{\text{1CPA}} \leftarrow_{\text{wrap}} A_{\text{qdCCA}}$.

So we still have to prove the bound

$$\begin{aligned}\text{Adv}(A_{\text{qdCCA}}) \leq{} & 4((q+1) \cdot \text{Adv}(A_{\text{1PO}})) + 4\sqrt{(q+1) \cdot \text{Adv}(A_{\text{1PO}})}\sqrt{\text{Adv}(A_{\text{1CPA}})} \\ & + \text{Adv}(A_{\text{1CPA}}).\end{aligned}$$

If $\text{Adv}(A_{\text{1CPA}}) \geq \text{Adv}(A_{\text{qdCCA}})$, this is trivial. For the rest of this proof, we assume $\text{Adv}(A_{\text{1CPA}}) < \text{Adv}(A_{\text{qdCCA}})$.

In our context, $P_{\text{left}} = \text{win}(A_{\text{qdCCA}})$, $P_{\text{right}} = \text{win}(A_{\text{1CPA}})$, and $P_{find} = A_{\text{1PO}}$.

Equations 9 (or rather, the derived Equation 16) implies

$$\begin{aligned}\text{win}(A_{\text{qdCCA}}) \leq{} & 4((q+1) \cdot \text{Adv}(A_{\text{1PO}})) + 4\sqrt{(q+1) \cdot \text{Adv}(A_{\text{1PO}})}\sqrt{\text{win}(A_{\text{1CPA}})} \\ & + \text{win}(A_{\text{1CPA}}).\end{aligned}$$

We apply $\text{win}(A) = (\text{Adv}(A) + 1)/2$ and simplify the expression to

$$\begin{aligned}\text{Adv}(A_{\text{qdCCA}}) \leq{} & 8((q+1) \cdot \text{Adv}(A_{\text{1PO}})) + 8\sqrt{(q+1) \cdot \text{Adv}(A_{\text{1PO}})}\sqrt{2}\sqrt{\text{Adv}(A_{\text{1CPA}}) + 1} \\ & + \text{Adv}(A_{\text{1CPA}}).\end{aligned}$$

as claimed.

LearnForward(G)

```
Receive $(N_i, H_i, M_i)$ from $A_2^{\text{cheat}}$
if $G = G_{KD,\perp,KD}$ or
  $G = G_{KD,\perp,Find}$ then
    $K_i \leftarrow \text{KD}_{K'}(N_i)$
    $C_i \leftarrow \text{AEnc}_{K_i}(N_i, H_i, M_i)$
else
    if $G = G_{\$,\perp,A_2}$ or
      $G = G_{\$,\perp,KD}$ or
      $G = G_{\$,\perp,Find}$ then
        $R_i \xleftarrow{\$} \{0,1\}^k$
        $C_i \leftarrow \text{AEnc}_{K_i}(N_i, H_i, M_i)$
    else
        Forward $(N_i, H_i, M_i)$ to
        challenger
        Receive $C_i$ from
        challenger
Send $C_i$ to $A_2$
```

LearnBackward()

```
Receive $(N_i, H_i, C_i)$ from $A_2^{\text{cheat}}$
Send $\perp$ to $A_2^{\text{cheat}}$
```

Challenge(G, b)

```
Receive $(N_i, H_{i,0}, H_{i,1}, M_{i,0}, M_{i,1})$
  from $A_2^{\text{cheat}}$
if $G = G_{\$,\perp,A_2}$ or $G = G_{A_2,\perp,A_2}$
  then
    Forward query to challenger
    Receive $C_{i,b}$ from challenger
else
    if $G = G_{\$,\perp,KD}$ or
      $G = G_{KD,\perp,KD}$ then
        $K_i \leftarrow \text{KD}_{K'}(N_i)$
        $C_{i,b} \leftarrow$
          $\text{AEnc}_{K_i}(N_i, H_{i,b}, M_{i,b})$
    else
        Choose new
        $(N_{i+1}, H_{i+1}, C_{i+1})$
Send $C_{i,b}$ to $A_2^{\text{cheat}}$
```

Exp′(G, b)

```
if $G = G_{KD,\perp,KD}$ or $G = G_{\$,\perp,KD}$ or
  $G = G_{KD,\perp,Find}$ then
    $K' \xleftarrow{\$} \{0,1\}^k$
for $i \in 1 \dots p(n)$ do
    Perform either LearnForward(G),
      LearnBackward() or Challenge(G,b)
if $G = G_{A_2,\perp,Find}$ or $G = G_{\$,\perp,Find}$ or
  $G = G_{KD,\perp,Find}$ then
    Receive answer $X$ from challenger
    if $X = \perp$ then
        return 0
    return 1
Receive guess $b'$ from $A_2^{\text{cheat}}$
return $b'$
```

Fig. 8: Experiment run by the adversary. G can be $G_{A_2,\perp,A_2}$, $G_{\$,\perp,A_2}$, $G_{A_2,\perp,\text{Find}}$, $G_{\text{KD},\perp,\text{KD}}$, $G_{\$,\perp,\text{KD}}$, $G_{\text{KD},\perp,\text{Find}}$, or $G_{\$,\perp,\text{Find}}$.