

Tweakable Blockciphers, Revisited

Anonymized for submission

April 21, 2009

Abstract

Tweakable blockciphers, first formalized by Liskov, Rivest, and Wagner [17], are blockciphers with an additional input, the *tweak*, which provides an easy mechanism for obtaining multiple “essentially different” permutations from a single key. Liskov et al. advocate an altered methodology for symmetric cryptography: instead of designing modes of operation using blockciphers directly, first design tweakable blockciphers, and then build modes of operation. Though this method has conceptual advantages, it can introduce an extra layer of analysis in which proof tightness can be lost.

We consider the notion of *security-preserving* mid-level constructions, by which we mean constructions that do not introduce any loss of security. We give tweakable blockciphers that meet this goal in a limited sense (they are security-preserving for certain applications), and show that they can help us create tighter overall proofs of security. We also show the novelty of these constructions by demonstrating that all previously proposed generic tweakable blockciphers are not security preserving even in this limited sense.

1 Introduction

Tweakable blockciphers are designed to internally handle the need for variability arising in the design of blockcipher modes of operation. *Tweakable blockciphers* are blockciphers which take an extra input, the tweak, to provide a source of variation to the blockcipher. The tweak is meant to be public and can be thought to be chosen by the adversary during his queries. A tweakable blockcipher is considered secure if it is indistinguishable from a family of random permutations indexed by the tweak[17].

The tweakable blockcipher methodology. Liskov et al. [17] espouse the theory that the use of tweakable blockciphers represent a better methodology for approaching the design of modes of operation. While the traditional methodology separates the design of the blockcipher from the design and analysis of the mode of operation (which uses the blockcipher as a black box), the newer methodology solves the overall problem differently, breaking the abstraction at a higher level.

Liskov et al. argue that the new methodology is a conceptually cleaner way to approach the problem of designing modes of operation. The abstraction of the blockcipher’s needed variability to a tweakable blockcipher is a natural one, and allows for simpler proofs of security. However, once a mode of operation has been proven secure it does not matter how easy the proof was to find. An unanswered question is whether this methodology may be costly in other ways, especially in terms of the tightness of security reductions.

Are tweakable blockciphers the weakest link in the chain of security? The security of the ultimate application is determined, in the old methodology, by two factors: the security of the blockcipher, and the tightness of the mode of operation. If the tweakable blockcipher is constructed *generically* –

that is, using an underlying blockcipher as a black box, there is a third layer in the new methodology: the “tightness” of the tweakable blockcipher itself.¹ If the analysis leads to a loss of tightness in reductions because of this third layer, this methodology would be inferior despite its conceptual advantages.

This issue is certainly of concern, because existing generic tweakable blockcipher constructions have proofs that at best match the “birthday bound” $\Omega(\frac{q^2}{2^n})$. Any application using such a generic tweakable blockcipher must necessarily have security advantage no tighter than the birthday bound, even if neither the tweakable mode of operation nor the underlying blockcipher are susceptible to such an attack.

Our results. To address this concern, we formalize the notion of *security-preserving* mid-level constructions – i.e. generic constructions (like tweakable blockciphers) that are “as secure as” the underlying primitives (like blockciphers) they are built from and the application (like a tweakable mode of operation) they are used in. In other words, security-preserving constructions are guaranteed to not be a weak link. We discuss three types of security preservation: *absolute security preservation* (for *all* applications), *security preservation* for a specific application, and security preservation in an *attack model*, that is, against adversaries that obey certain restrictions. Attack models drastically alter the normal notion of security, and are only of interest when the reduction proving an application secure happens to be compliant with a given attack model; in such a case, security preservation for the attack model implies security preservation for the application.

We then turn our attention to security-preserving tweakable blockciphers. We give two examples of attack models for which we construct security-preserving tweakable blockciphers, and which imply security preservation for two specific applications. In particular, we consider an attack model compatible with TAE mode for authenticated encryption [17] and give a tweakable blockcipher security-preserving in this model. This results, overall, in a proof of security for an authenticated encryption scheme with security advantage *totally independent* of the adversary’s time or number of queries. In contrast, no authenticated encryption mode has been proven with tightness better than the birthday bound [22, 21, 6, 7], however the security advantage we establish is $\Omega(\frac{c}{2^n})$ where c is a constant.

We also show that *all* previously proposed generic tweakable blockciphers are not security-preserving even in a weak model where the adversary may not query with any tweak more than once. This gives strong evidence to suggest that all previously proposed constructions are not security-preserving for *any* interesting application, since this attack model is so restrictive.

Our conclusions. Our results show that the tweakable blockcipher methodology does not inherently cause security degradation. There is even an apparent *advantage* in using the tweakable blockcipher methodology, if issues of security preservation are considered, as applications built on security-preserving tweakable blockciphers have optimal security bounds.

Prior work. Schroepel’s Hasty Pudding Cipher [23] was the first to include an auxiliary input for variability, the “spice”. Liskov et al. later formalized the notion of tweakable blockciphers [16, 17]. Later work further analyzed and elaborated on the generic constructions of Liskov et al [21, 18, 4, 25], while Goldenberg et al. give direct constructions based on Luby-Rackoff blockciphers [8]. Halevi and Rogaway [11, 12] suggest an application of tweakable blockciphers to disk encryption where

¹Generic constructions are the norm: most literature on tweakable blockciphers deals exclusively with generic constructions, the sole exception being the recent paper of Goldenberg, et al. [8].

the tweak is set to be the memory address of an encrypted block. This leads to constructions of “tweakable enciphering modes”, such as EME, EME*, EMD, CMC [11, 12, 10, 18], and analyses of these constructions [14, 19]. The idea of using tweakable blockciphers for disk encryption is being considered by SISWG for the proposed IEEE disk encryption standard P1619 [24]. Tweakable blockciphers have also been studied in a variety of other contexts [3, 1].

As far as we know, our notion of security preservation is new, but similar research has been done in the area of combiners. Unlike the notion of security preservation, combiners are meant to provide some level of security, given that “enough” of the underlying primitives are secure [20, 13].

Roadmap. In Section 2, we provide some notation and background definitions. In Section 3, we

formally develop the notion of security preservation. In Section 4, we give constructions of tweakable blockciphers that are security-preserving in specific attack models, and show that those models are applicable to modes of operation for symmetric encryption and for authenticated encryption. Section 5 presents attacks on previous tweakable blockcipher constructions and variants. We then conclude the paper by discussing the implications of this work in Section 6.

2 Preliminaries

In this section, we introduce definitions and notation that are used throughout the paper.

Adversaries. We denote $A^{\mathcal{O}}$ as an adversary with oracle access to the oracle \mathcal{O} . We denote by $\langle \mathcal{O}_1, \mathcal{O}_2 \rangle$ an oracle with two sub-oracles \mathcal{O}_1 and \mathcal{O}_2 and we assume that an adversary’s query to $\langle \mathcal{O}_1, \mathcal{O}_2 \rangle$ identifies which oracle the adversary is querying.

Security advantage. The advantage that an adversary has when distinguishing between two oracles \mathcal{O} and \mathcal{O}' with security parameter n is:

$$\text{ADV-DIST}(\mathcal{O}, \mathcal{O}', A, n) = |\Pr[A^{\mathcal{O}}(1^n) = 1] - \Pr[A^{\mathcal{O}'}(1^n) = 1]|.$$

We similarly define

$$\text{ADV-DIST}(\mathcal{O}, \mathcal{O}', q, t, n) = \max_{A \in \mathcal{A}_{q,t}} \text{ADV-DIST}(\mathcal{O}, \mathcal{O}', A, n)$$

where $\mathcal{A}_{q,t}$ consists of all adversaries that make at most q oracle queries and run in at most t time.

Blockciphers and tweakable blockciphers. We review the standard definitions of a blockcipher [9] and a tweakable blockcipher [17] respectively. A *blockcipher* family is a family of efficiently computable, functions $E_n : \mathcal{K}_n \times \{0, 1\}^n \rightarrow \{0, 1\}^n$ where \mathcal{K}_n is a non-empty set, and for all $K \in \mathcal{K}_n$, for each $n \geq 1$, $E_n(K, \cdot)$ is an efficiently invertible permutation. Define

$$\text{ADV-BC}^\alpha(E_n, q, t, n) = \text{ADV-DIST}(\mathcal{O}_{E_n}^\alpha, \Pi^\alpha, q, t, n)$$

where α is either $+$ or \pm and \mathcal{O}_{E_n} is the oracle which selects $K \in \mathcal{K}_n$ at random, and then returns $E(K, \cdot)$. For $\alpha = +$, $\mathcal{O}_{E_n}^\alpha = \mathcal{O}_{E_n}$, and $\Pi^\alpha = \Pi$ where Π is a random permutation. For $\alpha = \pm$, $\mathcal{O}_{E_n}^\alpha = \langle \mathcal{O}_{E_n}, \mathcal{O}_{E_n^{-1}} \rangle$ and $\Pi^\alpha = \langle \Pi, \Pi^{-1} \rangle$.² $\text{ADV-BC}^+(E_n, q, t, n)$ is the chosen-plaintext security of E , and $\text{ADV-BC}^\pm(E_n, q, t, n)$ is the chosen-ciphertext security of E .

²Here, $\mathcal{O}_{E_n^{-1}}$ is defined in the natural way: it shares K with \mathcal{O}_{E_n} and computes $E^{-1}(K, \cdot)$.

A *tweakable blockcipher* family is an efficiently computable, efficiently invertible function family $\tilde{E}_n : \mathcal{K}_n \times \mathcal{T}_n \times \{0, 1\}^n \rightarrow \{0, 1\}^n$ where \mathcal{K}_n and n are as before, \mathcal{T}_n is a non-empty set, and for every n , $K \in \mathcal{K}_n$, $T \in \mathcal{T}_n$ $\tilde{E}_n(K, T, \cdot)$ is an efficiently invertible permutation. Define

$$\text{ADV-TBC}^\alpha(\tilde{E}_n, q, t, n) = \text{ADV-DIST}(\mathcal{O}_{\tilde{E}_n}^\alpha, \tilde{\Pi}^\alpha, q, t, n)$$

where α is either $+$ or \pm and $\mathcal{O}_{\tilde{E}_n}$ computes $\tilde{E}_n(K, \cdot, \cdot)$ for a random K , and where $\tilde{\Pi}$ is a family of independent random permutations indexed by the tweak. We define $\mathcal{O}_{\tilde{E}_n}^\alpha$ and $\tilde{\Pi}^\alpha$ similarly to $\mathcal{O}_{E_n}^\alpha$ and Π^α .

For tweakable blockciphers (and where applicable to blockciphers) we define n as the blocksize, \mathcal{T} as the tweak space, and \mathcal{K} as the key space. When n is clear, we refer to a blockcipher and a tweakable blockcipher as E and \tilde{E} , respectively.

Attack models. Informally, an attack model is specified by an oracle \mathcal{M} that acts like a filter, restricting the adversary to making certain allowable queries. We denote by $\mathcal{O}^\mathcal{M}$, the oracle \mathcal{O} in the attack model defined by \mathcal{M} . When an adversary queries $\mathcal{O}^\mathcal{M}$, the query is first given to \mathcal{M} , which either returns \perp , or passes the query onto \mathcal{O} . We say an adversary is *model-compliant* if $A^{\mathcal{O}^\mathcal{M}}$ does not ever cause \mathcal{M} to return \perp .

We denote security advantages in an attack model with the notation $\text{ADV}^\mathcal{M}$. For instance, the advantage that an adversary has in attacking a tweakable blockcipher under a specific attack model is:

$$\text{ADV-TBC}^{\alpha, \mathcal{M}}(\tilde{E}_n, q, t, n) = \text{ADV-DIST}(\mathcal{O}_{\tilde{E}_n}^{\alpha, \mathcal{M}}, \mathcal{O}_{\tilde{\Pi}}^{\alpha, \mathcal{M}}, q, t, n).$$

3 Security preservation

In this section we define and explore the notion of security preservation. We imagine a scenario where a top-level application Φ is implemented using a mid-level construction C which in turn is implemented with a low-level primitive Δ . C 's security is defined by its indistinguishability from an ideal version Ψ , and similarly, Δ 's security is defined by an ideal version Θ .

We are concerned with the damage the use of a specific choice for the mid-level construction might cause to the tightness of the overall proof of the application.

We give some general notational conventions. Let Δ and Θ be cryptographic primitives, Θ being an ideal primitive, where Δ is considered secure if it is indistinguishable from Θ . Let r be a constant, and let C be a cryptographic construction which uses r primitives, where C is secure if $C_{\Delta_1, \Delta_2, \dots, \Delta_r}$ is indistinguishable from Ψ , an ideal construction. We denote by C_Θ the construction which utilizes only ideal primitives, and C_Δ the construction utilizing only the real primitives. C refers to the construction with the underlying primitives treated as black boxes.

Definition 3.1 *The absolute security preservation of C is $\max_A \text{ADV-DIST}(C_\Theta, \Psi, A, n)$.*

The tightness of the overall application will generally be based on three parts, two which are independent of the construction C . One part is the security of the underlying primitives: $\text{ADV-DIST}(\Delta_i, \Theta_i)$, while the other is the tightness of the reduction between Φ and the ideal Ψ that C aims to implement. The remaining piece is the distinguishing advantage between C_Θ and Ψ . The absolute security preservation of C represents a bound on this one relevant advantage.

Note that we take the maximum over *all* adversaries here: if, for instance $\text{ADV-DIST}(C_\Theta, \Psi, q, t, n) = \frac{q^2}{2^n}$, then we would not be able to prove the application secure for adversaries that make $q \geq 2^{n/2}$

queries, even if the other two factors would not rule out security in such a circumstance: in other words, the mid-level construction has created a vulnerability against (possibly) avoidable attacks.

Whether we consider a construction absolutely security preserving is a judgment about whether the absolute security preservation is “small enough”, so we try to avoid the term. However, certainly a construction is not absolutely security-preserving if its security preservation is inverse polynomial or greater than a non-zero constant.

Using Definition 3.1, we show that the security advantage of the resulting construction can be bounded by the advantages of the underlying primitives and the absolute security preservation of the construction, provided we account for the overhead required to run the reduction in our bounds for q and t . $\rho(C, i, n)$ and $\tau(C, i, n, q)$ represent this overhead; see Appendix B for a precise definition.

Theorem 3.2 *If ASP_C is the absolute security preservation of C , then*

$$\text{ADV-DIST}(C_\Delta, \Psi, q, t, n) \leq \text{ASP}_C(n) + \sum_{i=1}^r \text{ADV-DIST}(\Delta_i, \Theta_i, q \cdot \rho(C, i, n), t + \tau(C, i, n, q), n).$$

Proof. The proof is a straightforward hybrid argument. See Appendix B for details. \square

Security preservation for applications. Absolute security preservation may be a stronger notion than what is necessary for tight reductions. In this section we define the notion of *security preservation for an application*, which captures the necessary properties to make the proof of a specific application as tight as possible.

First, we need some additional notation. Let C be an r -primitive construction, and let Φ_C be an application which utilizes C as a black box. We assume the existence of a black box reduction R that transforms an adversary A that attempts to attack Φ , into an adversary R^A that attempts to distinguish C from Ψ .

Definition 3.3 *The security preservation for Φ under R of C is $\max_A \text{ADV-DIST}(C_\Theta, \Psi, R^A, n)$.*

We now prove an application-oriented version of Theorem 3.2. It is more challenging to describe the specific overheads in this construction. Ultimately, the overheads involved will be just as for Theorem 3.2, but based on R^A rather than on A .

We assume that the security of Φ is defined via an advantage function $\text{ADV}_{ap}(\Phi, A, n)$.

Definition 3.4 *The tightness of the reduction R is the function $f_R(l, q, t)$ defined by $f_R(l, q, t) = \max_{A \in \mathcal{A}_{l, q, t}} \text{ADV}_{ap}(\Phi_{C_\Delta}, A, n) - \text{ADV-DIST}(C_\Delta, \Psi, R^A, n)$, where $\mathcal{A}_{l, q, t}$ refers to the class of adversaries that take no more than t total time and use no more than l total query length and q total queries.*

Theorem 3.5 *If $\text{SP}_{C, \Phi, R}$ is the security preservation for Φ under R of C , then for all l, q , and t ,*

$$\max_{A \in \mathcal{A}_{l, q, t}} \text{ADV}_{ap}(\Phi_{C_\Delta}, A, n) \leq \text{SP}_{C, \Phi, R}(n) + f_R(l, q, t) + \sum_{i=1}^r \text{ADV-DIST}(\Delta_i, \Theta_i, q'_i, t'_i, n),$$

where $q'_i(l, q, t) = q_R(l, q, t) \cdot \rho(C, i, n)$ and $t'_i = t_R(l, q, t) + \tau(C, i, n, q_R(l, q, t))$, where $q_R(l, q, t)$ and $t_R(l, q, t)$ are defined as the maximal number of queries and time R^A makes, respectively, given an A that makes queries of total length at most l , at most q queries, and uses time at most t .

Proof. This proof is a fairly straightforward extension of the proof of Theorem 3.2. See Appendix B for details. \square

Security preservation in an attack model. We now introduce the idea of a construction being security preserving in an attack model as a way of showing that the construction is security preserving for an application.

Definition 3.6 *The security preservation in model \mathcal{M} of C is $\max_A \text{ADV-DIST}(C_{\Theta}^{\mathcal{M}}, \Psi^{\mathcal{M}}, A, n)$.*

Before we can use this notion to relate security preservation in \mathcal{M} to security preservation for an application, we must have a reduction that is “compliant” with \mathcal{M} in the following sense:

Definition 3.7 *Let R be a reduction. R is an “ \mathcal{M} -compliant reduction” if for all A , the probability that $(R^A)^{\mathcal{M}}$ makes a query such that \mathcal{M} returns \perp is 0.*

Using Definition 3.6 and 3.3, we can show that for a construction to be security-preserving for an application, it suffices to show that the construction is security-preserving in the \mathcal{M} model and that the reduction is also \mathcal{M} -compliant.

Theorem 3.8 *For any C , if Φ is an application with an \mathcal{M} -compliant reduction R , then the security preservation for Φ under R of C is at most the security preservation in \mathcal{M} of Φ .*

Proof. This is straightforward: $\{R^A\} \subset \{A \mid A \text{ complies with } \mathcal{M}\}$, because R is \mathcal{M} -compliant. Since the security preservation for Φ under R is a maximum over the former set, and the security preservation in \mathcal{M} is a maximum over the latter, we have our result. \square

This, combined with Theorem 3.5, establishes that if a construction is security-preserving in the \mathcal{M} model, and Φ is an application with an \mathcal{M} -compliant reduction, then the security of Φ_C is bounded by the security of the underlying primitives plus the security preservation in \mathcal{M} of C .

4 Security-preserving constructions

We now give examples of tweakable blockciphers that illustrate the practicality of the idea of security preservation. We show the existence of tweakable blockciphers security preserving in attack models relevant to two modes of operation: TIE mode [16] for symmetric encryption, and TAE mode [17] for authenticated encryption.

Notation. We use similar notation to that of the previous section. We use E to denote a blockcipher instantiated with a randomly chosen key and Π to denote the idealized version of E , a random permutation. We similarly use F to denote a pseudorandom function with a randomly chosen seed and R to denote a random function.³ In order to define a construction without specifying the primitives we write the construction foregoing any subscripts; for example, we define FEF' to be $F'(T) \oplus E(M \oplus F(T))$. We then write $\text{FEF}'_{F,E,F'}$ to refer to FEF' with real primitives, and $\text{FEF}'_{R,\Pi,R'}$ to refer to FEF' with ideal primitives.

³Note that this standard definition inherently rules out constructions in which the tweak affects the key used by the blockcipher, so we restrict our analysis to generic constructions of this type, which dominate the literature. Key-affecting constructions additionally raise the question of security against related-key attacks [2], which can only be resolved by relying on a strong assumption like the ideal cipher model, which we avoid in this work.

4.1 Security preservation in the SQT model

In this section, we consider the single query per tweak attack model:

Definition 4.1 (SQT attack model) *The SQT attack model oracle, \mathcal{SQT} , returns \perp if the adversary ever makes a query (M, T) such that there exists a previous query (M', T') where $T' = T$.*

We give a generic construction with security preservation of 0 in the SQT model and we prove that tweak incrementation mode, or TIE mode [16], has an SQT-compatible reduction. Thus, we exhibit a tweakable blockcipher construction security preserving for TIE mode. Security preservation for TIE mode is exceedingly simple, and we discuss it only briefly as a proof of concept.

TIE mode. TIE mode is a stateful tweakable mode of operation for symmetric encryption. TIE mode was first proposed by Schroeppel, as an intended use of the Hasty Pudding cipher [23] and later formalized by Liskov [16].

To encrypt a message with m message blocks, $M_1||M_2||\dots||M_m$, TIE mode proceeds as follows: set $T = 0$ if it is undefined, and set $IV = T$. For each $1 \leq i \leq m$, compute $\tilde{E}(M_i, T) = C_i$, (or $C_i = \perp$ if $T \geq |T|^4$), and then increment T . The output of TIE mode is $(C_1||C_2||\dots||C_m, IV)$, and the value T is preserved as state. To decrypt, we compute $M_i = \tilde{E}^{-1}(C_i, IV + i - 1)$ for each i .

We now show that the reduction used to prove that TIE mode is secure is SQT-compliant.

Theorem 4.2 *There is a reduction from an attack on TIE mode to an attack on the tweakable blockcipher that is SQT-compatible.*

Proof. To prove this theorem, we prove that the reduction R used to prove that TIE mode is secure is SQT-compliant. The reduction R is simple: if an adversary A can distinguish the output of TIE mode encryptions from a random function with appropriate-sized outputs, then R^A distinguishes the underlying tweakable blockcipher from a random permutation family, by using its oracle to implement TIE mode in the attack A performs, and mimic A when A outputs. Since this reduction queries only what A indirectly queries, and since no matter how many queries A makes, each tweak is used only once, the reduction is SQT-compliant. \square

Since the reduction for TIE mode is SQT-compliant, any constructions security-preserving in the single query per tweak model are also security-preserving for TIE mode.

We prove that $F = M \oplus F(T)$ is security-preserving in the single tweak attack model, and is thus security-preserving for TIE mode.

Theorem 4.3 *The security preservation of F in the SQT model is 0.*

Proof. Note that $\tilde{\Pi}$ queried in the single-tweak model will give independent random outputs for every query. It is obvious that F does the same in the single tweak model, as the output of each query is the input XOR'ed with a fresh random value (regardless of whether the input is a query to F or its inverse). \square

Therefore F is security-preserving for TIE mode. This is not surprising: with F , TIE mode is simply an ordinary stream cipher.

⁴Note that our presentation here slightly modifies TIE mode so that it fails after encrypting $|T|$ message blocks. This is reasonable, as encrypting so many message blocks would lead to a distinguishing attack in any case, and this property does not meaningfully affect our ability to decrypt messages.

4.2 Security preservation in the NSQT_l model

In this section we define an attack model where the adversary is allowed to repeat l tweak queries twice. We call this attack model the “near-single query per tweak” or NSQT_l model and give a tweakable blockcipher we prove is security-preserving in this model. We also prove that the reduction used to prove TAE mode (which is itself very tight) [17] is NSQT_l -compatible. Thus, TAE mode can be proven secure with a very tight bound, when the blockcipher is security preserving in the NSQT_l model.

Definition 4.4 (NSQT_l attack model) *The NSQT_l attack model oracle allows a query (M, T) as long as: there exists no previous query (M', T') where $T = T'$ **or** the total number of repeated tweaks is $< l$ and this query is the second use of T . Otherwise, NSQT_l returns \perp .*

The NSQT_l tweak attack model allows for up to l tweaks to be repeated twice, while all other tweaks can be used only once.

Authenticated encryption. TAE mode [17] is an *authenticated encryption* tweakable mode of operation in the tradition of prior modes by Jutla [15] and Rogaway et al. [22]. First, we must define the notion.

An authenticated encryption scheme, \mathcal{E} , is a set of algorithms (E, D) for encryption and decryption, respectively. E takes input a key K , a message M , a nonce N , and a parameter τ and produces a ciphertext C and a tag Tag . D takes input a key K , a ciphertext C , a nonce N and a tag Tag , and outputs either \perp or a message M , where \perp indicates that the tag was invalid.

The security properties required are

1. *Pseudorandomness:* Defined as $\text{ADV-DIST}(\mathcal{O}_E^{\mathcal{NR}}, R^{*\mathcal{NR}}, A, n)$, where R^* is a random function oracle taking (M, N, τ) to random strings C of length $|M|$ and Tag of length τ .
2. *Unforgeability.* Define the forgery advantage of an adversary A as:

$$\begin{aligned} \text{ADV-F}(E, A, n) = & \Pr[K \leftarrow \mathcal{K}; (C, Tag, N) \leftarrow A^{\mathcal{O}_{EK}^{\mathcal{NR}}}; M \leftarrow D_K(C, N, Tag) : \\ & M \neq \perp \wedge (M, N, |Tag|) \text{ was not queried to } E_K]. \end{aligned}$$

Adversaries are expected to be “nonce-respecting”, meaning that they work in this attack model. We impose this restriction via \mathcal{NR} , an attack model oracle that returns \perp whenever it receives a query involving a nonce N for the second time. Note that the adversary may attempt a forgery involving a nonce that was used previously.

Note that all previous authenticated encryption modes have proofs that require q to be much smaller than $2^{\frac{n}{2}}$, regardless of the security of the underlying blockcipher. Furthermore, OCB mode has a known attack meeting this bound [6].

TAE mode. To encrypt a message $M_1||M_2||\dots||M_m$ where $|M_i| = n$, (with the exception of the M_m), using a tweakable blockcipher \tilde{E} which takes n bit message to n bit ciphertexts, and nonce N , TAE mode encryption E operates as follows:

- $Z_0 = N||\langle b \rangle||1$ where $\langle b \rangle$ is an a -bit representation of the bit length of the full message or \perp if b can not be represented in a bits. (a may be considered a parameter of the scheme.)
- $Z_i, i = 1\dots m$ are defined as $N||\langle i \rangle||0$, where $\langle i \rangle$ is defined as an a -bit representation of i .

- $C_i, i = 1 \dots m - 1$ is defined as $\tilde{E}(M_i, Z_i)$.
- $C_m = \tilde{E}(l, Z_m)|_l \oplus M_m$, where $l = \text{len}(M_m)$ is the n -bit representation of the length of the last message block, and $A|_n$ denotes A truncated to its first n bits.
- $\text{Checksum} = M_1 \oplus M_2 \oplus \dots \oplus (M_m 0^{n-\text{len}(M_m)})$.
- $\text{Tag} = \tilde{E}(\text{Checksum}, Z_0)|_\tau$, or \perp if $Z_0 = \perp$.
- The ciphertext is defined as $N, C_1, \dots, C_m, \text{Tag}$.

A ciphertext, C_1, \dots, C_m , is valid if the message produces a checksum which is valid with its corresponding Tag . Specifically, we compute $M_i = D(C_i, Z_i)$ for $i < m$, $M_m = \tilde{E}(\text{len}(C_m), Z_m)|_{\text{len}(C_m)} \oplus C_m$, $\text{Checksum} = M_1 \oplus M_2 \oplus \dots \oplus M_m 0^{n-\text{len}(M_m)}$, and then check that Tag is the first τ bits of $\tilde{E}(\text{Checksum}, Z_0)$ and if $Z_0 \neq \perp$.⁵

Since TAE mode needs to have both properties of pseudorandomness and unforgeability, the proof of security consists of two reductions. The reduction for pseudorandomness is simple: use the given oracle in place of the tweakable blockcipher and allow the adversary to launch its attack indirectly; mimic the output of the adversary. This reduction is obviously SQT-compliant.

However, the reduction for unforgeability is not SQT-compliant, so we define an attack model appropriate for that reduction. However, it is NSQT_{2^a}-compliant:

Theorem 4.5 *TAE mode has a NSQT_{2^a}-compliant reduction, with respect to its unforgeability property, with tightness $\frac{3}{2^{n+1}-2}$.*

Proof. We exhibit a reduction R such that for any forging adversary A , R^A performs a distinguishing attack on the underlying tweakable blockcipher. R has access to an oracle; it will treat that oracle like a tweakable blockcipher and run A in its forgery attack against TAE mode as built with R 's oracle. When A produces an attempted forgery, R decipheres the message blocks $M_1 \dots M_m$, computes the checksum, and then checks if the adversary's tag is correct. If it is, R outputs 1, otherwise, R outputs 0.

Liskov et al. prove that this reduction has tightness $\frac{3}{2^{n+1}-2}$ [17].

Furthermore, note that if the adversary is nonce-respecting then no tweak will ever be repeated twice in the adversary's queries. However, the adversary may produce a forgery that reuses an old nonce. Note that R^A need not decipher the adversary's message if it knows ahead of time that $Z_0 = \perp$. Therefore, the maximum number of tweaks that R uses twice is the maximal length of an attempted forgery where $Z_0 \neq \perp$; this length is at most $2^a + 1$. \square

As we show in section 5, all previously proposed tweakable blockciphers are not security-preserving in the SQT model, and are thus not security-preserving in the NSQT_l model. We consider the tweakable blockcipher $\text{FEF}' = F'(T) \oplus E(M \oplus F(T))$. We prove that FEF' is security-preserving in the NSQT_l model.

Theorem 4.6 *The security preservation of FEF' in the NSQT_l model is at most $\frac{2(l^2-3l)}{2^n}$.*

⁵Our presentation here slightly differs from TAE mode as presented in Liskov et al. In particular, we specify a , the maximum length of $\langle b \rangle$, as a parameter whereas in Liskov et al. $a = n/2$ was specified. Second, we explicitly make the verification process fail for extremely long messages (longer than 2^a bits), whereas this boundary condition was not addressed in the original.

Proof sketch. Due to space constraints, we give only an outline of the proof here; for details, see Appendix B. Any query made by the adversary with a tweak that is used only once yields a random result for either $\text{FEF}'_{R,\Pi,R'}$ or $\tilde{\Pi}$. Using the principle of deferred decisions, it will be indistinguishable to the adversary if we do not assign inputs and outputs for Π except for queries that use a tweak a second time. Among those queries, if no collision on input or output values for Π occurs, the behavior is identical. Furthermore, with only l such queries, the probability of such a collision depends only on l , not on the total number of queries or the computation time. \square

Since FEF' is security-preserving (to some degree) for TAE mode so long as $2^a + 1$ is small compared to $2^{n/2}$, TAE mode is “as secure as” the underlying blockcipher for such settings of the parameter a . Note that $a = n/2$ which is originally suggested for TAE mode does not work but for instance $a = n/4$ works, or indeed any $a = \frac{n}{2}(1 - \epsilon)$ for a fixed $\epsilon > 0$.

This is an improvement in security compared with OCB or TAE mode implemented with known blockciphers, whose proofs of security break down for $q \approx 2^{n/2}$ queries. TAE mode with FEF' and with $a = n/4$ can be broken with probability at most $\frac{3}{2^{n+1}-2} + \frac{1}{2^{\frac{n}{2}-1}}$ which is greater than the probability of breaking the blockcipher, regardless of the adversary’s power. Thus we find that not only does the tweakable blockcipher methodology not necessarily loosen security proofs, it can even be used to improve them! Tweakable blockciphers are powerful primitives and can lead (as was the case for TAE mode) to tight proofs for applications. If such a proof is combined with a security-preserving tweakable blockcipher, the result will be a very tight overall proof.

5 Attacks

Now that we have demonstrated the existence of security-preserving tweakable blockcipher constructions in the SQT and NSQT_l models, it remains to answer whether any known tweakable blockcipher constructions are *absolutely* security-preserving, or, if not, whether existing tweakable blockcipher constructions could be security-preserving for any interesting application.

We answer this in the negative. In this section, we show that all previously proposed generic tweakable blockciphers are not security-preserving, even in the SQT model⁶. It is hard to imagine a more restrictive attack model for which an interesting mode of operation could have a compliant reduction. Thus, attacks in the SQT model suggest that these constructions are not security-preserving for *any* interesting mode of operation.

In most cases, we demonstrate that these constructions are not security-preserving by giving a lower bound on ADV-TBC that increases as q increases. Since security preservation is independent of q , any such construction would have security preservation very close to 1.

The first tweakable blockcipher construction proposed by Liskov et al., $\text{ETE} = E(E(M) \oplus T)$, is not security-preserving even in the SQT model, and is distinguishable from a random permutation family in $O(2^{n/2})$ queries in the SQT model. Let $\text{ETE}' = E'(E(M) \oplus T)$; we show that even this generalized version is not security-preserving.

Theorem 5.1 *ETE' is not security-preserving in the SQT model. Specifically,*
 $\text{ADV-TBC}^{+\text{SQT}}(\text{ETE}'_{\Pi,\Pi'}, q, O(q), n) \geq \frac{q^2 - q}{2^{n+1}}$.

Proof. The distinguishing attack is simple; make q queries using the same message but different and distinct tweaks. If a collision in the ciphertexts occurs, output 0, otherwise output 1. With

⁶This implies that these constructions are not absolutely security-preserving. In addition, F is trivial to attack, and we show in Theorem B.1 that FEF' is not absolutely security-preserving.

regards to the construction, there should never be a collision, however one should occur after approx $2^{\frac{n}{2}}$ queries to a random permutation family. For sufficiently large q , the probability that such a collision occurs becomes close to 1. \square

Liskov et al.’s second construction, HEH is defined by $\text{HEH} = h(T) \oplus E(M \oplus h(T))$, where h is an $\epsilon - AXU_2$ hash function. As a first step towards analyzing this, we consider the construction $\text{FEF} = F(T) \oplus E(M \oplus F(T))$ where F is a pseudorandom function, and show that it is not security-preserving in the SQT attack model.

Theorem 5.2 *FEF is not security-preserving in the SQT model. Specifically, there exists a $q = O(2^{\frac{n}{2}})$ such that $\text{ADV-TBC}^{+\text{SQT}}(\text{FEF}_{R,\Pi,R}, q, O(q), n) \geq (1 - \frac{1.2}{2^{n/2}})/4$.*

Proof. The idea of the attack is that a collision can only happen at random with a random permutation family, but with the construction there are two different random ways to get collisions. Fix a message M and query M with q randomly chosen unique tweaks. If any two queries lead to the same result, output 1, otherwise output 0.

Let p be the probability that in picking q uniform n -bit values, some value is picked twice. Then, the probability of this attack returning 1 given $\tilde{\Pi}$, the ideal tweakable blockcipher, is p .

Similarly, p is the probability that, given $\text{FEF}_{R,\Pi,R}$, we query two tweaks such that $R(T) = R(T')$, which will always lead to a collision. If this does not occur (with probability $1 - p$), then the values $M \oplus R(T)$, $M \oplus R(T')$ are distinct. In this case, with probability at least p , a collision of the type $\Pi(M \oplus R(T)) \oplus \Pi(M \oplus R(T')) = R(T) \oplus R(T')$ occurs. Therefore, the advantage of this attack is $p(1 - p)$.

It has been proven that there exists a $q = O(2^{n/2})$ such that $p(1 - p) \geq (1 - \frac{1.2}{2^{n/2}})/4$ [8]. \square

This shows that HEH is not *in general* security-preserving in the SQT model, as using a pseudo-random function family for H is one type of $\epsilon - AXU_2$ hash function. We will next show how our attacks on ETE and FEF can be generalized to allow for attacks on all known generic constructions of tweakable blockciphers, including specific implementations of HEH.

5.1 Extending the attack on FEF

The tweakable blockcipher constructions we are aware of include: the Hasty Pudding Cipher [23] and the Mercy Cipher [5] (two direct constructions); the generic ETE and HEH constructions of Liskov et al. [17]; Rogaway’s XEX mode [21], an instantiation of HEH; constructions by Minematsu [18] and Chakraborty and Sarkar [4], which are variants similar to XEX mode; Goldenberg et al. [8] direct constructions of tweakable blockciphers from pseudorandom functions; and EME mode and its variants [11, 12, 10] which are generic but are not tweakable blockciphers, but rather tweakable symmetric encryption modes of operation.

We show that our general attacks apply to XEX, and also that these attacks apply to the conceptual tweakable blockcipher embedded in OCB mode.⁷ We do not address the Hasty Pudding Cipher, the Mercy Cipher, or the constructions of Goldenberg et al., as these are not generic constructions. Likewise, EME mode and its variants are not directly relevant here, as they are designed to encrypt arbitrary-length messages rather than single blocks. Nonetheless, in Appendix A we show a generic SQT model attack against EME and its variants with $O(2^{n/2})$ queries.

There are two key points in the analysis of Theorem 5.2. One: that for random T, T' , $R(T) = R(T')$ with probability at least $\Omega(2^{-n})$, and second: that $R(T) \oplus R(T') = X$ over randomly chosen

⁷The comments regarding XEX easily extend to [18] and [4], so we do not address those constructions explicitly.

$X \neq 0$ and random, distinct T, T' with probability $\Omega(2^{-n})$. If both of these properties apply to a function f , we say that f has the “collision property”. As long as f has the collision property, then the analysis of Theorem 5.2 will hold, which leads to the following corollary:

Corollary 5.3 *If f has the collision property, there exists a $q = O(2^{\frac{n}{2}})$ such that $\text{ADV-TBC}(\text{FEF}_{f,\Pi,f}, q, O(q), n) \geq (1 - \frac{1.2}{2^{n/2}})/4$.*

Note that most, but not all, f would have this property. Specifically, if f were a permutation, it would not have this property.

We first examine XEX mode of Rogaway [21]. This mode is similar to $\text{FEF}_{f,\Pi,f}$, where the function f is the “masking function” $\Delta(T, i_1, \dots, i_k) = \alpha_1^{i_1} \alpha_2^{i_2} \dots \alpha_k^{i_k} T$ where $\alpha_j \in F_{2^n}$ such that $\alpha_1^{i_1} \dots \alpha_k^{i_k} = \alpha_1^{i'_1} \dots \alpha_k^{i'_k}$ only when $i_j = i'_j$ for all j . Here, the tweak is the entire input (T, i_1, \dots, i_k) .

Theorem 5.4 *The XEX mode Δ function has the collision property.*

Proof. $\Delta(T, i_1, \dots, i_k)$ is a permutation for each fixed tuple (i_1, \dots, i_k) . If we let X be any arbitrary value (including 0), and we choose $(T, i_1, \dots, i_k) \neq (T', i'_1, \dots, i'_k)$ at random, then the probability that $\Delta(T, i_1, \dots, i_k) \oplus \Delta(T', i'_1, \dots, i'_k) = X$ is 2^{-n} whenever $(i_1, \dots, i_k) \neq (i'_1, \dots, i'_k)$, is $\frac{1}{2^n - 1}$ whenever $(i_1, \dots, i_k) = (i'_1, \dots, i'_k)$ and $X = 0$, and is 0 otherwise. However, the overall probability of a full collision on Δ is still $\Omega(2^{-n})$, and the probability of an X -collision for $X \neq 0$ is always 2^{-n} , so Δ has the collision property. \square

OCB mode’s f function also has the collision property; the analysis is similar. See Appendix A.1 for details.

6 Discussion

The tweakable blockcipher methodology approaches the design of cryptographic applications via three steps:(1) Construct secure cryptographic primitives. (2) Construct a tweakable blockcipher from secure primitives.(3) Construct an application from a tweakable blockcipher.

Compared to the traditional methodology, which constructs applications directly from primitives, this introduces an extra step. This extra step has a corresponding effect on the tightness of the ultimate proof. In some cases, the tightness of the overall proof may be weakened by the use of this methodology.

Here, we are able to show some positive results for tweakable blockciphers: specifically, we show tweakable blockciphers that are security preserving in the SQT and NSQT_l models. These models are in turn relevant to the TIE symmetric encryption mode and the TAE authenticated encryption mode, respectively. Thus, security preservation for relevant applications can be established for the right tweakable blockciphers. This leads us to the conclusion that the tweakable blockcipher methodology can be followed without introducing unneeded weaknesses.

In addition, our overall security bound for TAE mode, when implemented with FEF' , has a security advantage of $\Omega(\frac{c}{2^n})$, that is, a constant, negligible security advantage. This is a tighter bound than has been shown for an authenticated encryption mode previously. Its independent interest aside, this shows that the tweakable blockcipher methodology, when security preservation is considered, can be fruitful in improving the overall proof tightness for interesting applications.

References

- [1] M. Bellare and T. Kohno. A Theoretical Treatment of Related-Key Attacks: PKA-PRPs, RKA-PRFs, and Applications. In E. Biham, editor, *Advances in Cryptology – EUROCRYPT ’03*, volume 2656 of LNCS, pages 491–506, 2003.
- [2] Eli Biham. New types of cryptanalytic attacks using related keys. *Journal of Cryptology*, 7(4):229–246, Fall 1994. Also available at: citeseer.nj.nec.com/biham94new.html.
- [3] J. Black, M. Cochran, and T. Shrimpton. On The Impossibility of Highly-Efficient Blockcipher-Based Hash Functions. In *Advances in Cryptology – Eurocrypt 2005*, volume 3494 of LNCS, pages 526–541. Springer Verlag, May 2005.
- [4] D. Chakraborty and P. Sarkar. A General Construction of Tweakable Block Ciphers and Different Modes of Operation. In *Information Security and Cryptology*, volume 4318, pages 88–102. Springer-Verlag Berlin Heidelberg, 2006.
- [5] P. Crowley. Mercy: A Fast Large Block Cipher for Disk Sector Encryption. In Bruce Schneier, editor, *FSE - Fast Software Encryption 2000*, volume 1978 of *Lecture Notes in Computer Science*, pages 49–63. Springer, 2000.
- [6] N. Ferguson. Collision attacks on OCB. Unpublished manuscript, 2002. <http://www.cs.ucdavis.edu/~rogaway/ocb/fe02.pdf>.
- [7] V. Gligor and P. Donescu. Fast encryption and authentication: Xcbc encryption and xecb authentication modes, 2000.
- [8] D. Goldenberg, S. Hohenberger, M. Liskov, E. Crump Schwartz, and H. Seyalioglu. On Tweaking Luby-Rackoff Blockciphers. In *Advances in Cryptology – ASIACRYPT 2007*. to appear. Full version available on ePrint, report 2007/350.
- [9] O. Goldreich. *Foundations of Cryptography*. Cambridge University Press, Cambridge, the United Kingdom, 2001.
- [10] S. Halevi. EME*: extending EME to handle arbitrary-length messages with associated data. In *INDOCRYPT*, pages 315–327, 2004.
- [11] S. Halevi and P. Rogaway. A Tweakable Enciphering Mode. In Dan Boneh, editor, *Advances in Cryptology – CRYPTO ’03*, volume 2729 of LNCS, pages 482–499, 2003.
- [12] S. Halevi and P. Rogaway. A Parallelizable Enciphering Mode. In Tasuaki Okamoto, editor, *Topics in Cryptology – CT-RSA ’04*, volume 2964 of LNCS, pages 292–304, 2004.
- [13] D. Harnik, J. Killian, M. Naor, O. Reingold, and A. Rosen. On Robust Combiners for Oblivious Transfer and other Primitives. In *EUROCRYPT*, 2005.
- [14] A. Joux. Cryptanalysis of the EMD Mode of Operation. In E. Biham, editor, *Advances in Cryptology – EUROCRYPT ’03*, volume 2656 of LNCS, pages 1–16, 2003.
- [15] C. Jutla. Encryption modes with almost free message integrity. In Birgit Pfitzmann, editor, *Advances in Cryptology—EUROCRYPT 2001*, volume 2045 of *Lecture Notes in Computer Science*, pages 529–544. Springer-Verlag, 6–10 May 2001.
- [16] M. Liskov. *New Tools in Cryptography: Mutually Independent Commitments, Tweakable Block Ciphers, and Plaintext Awareness via Key Registration*. PhD thesis. PhD. Thesis, MIT, 2004.
- [17] M. Liskov, R. Rivest, and D. Wagner. Tweakable Block Ciphers. In Moti Yung, editor, *Advances in Cryptology – CRYPTO ’02*, volume 2442 of LNCS, pages 31–46, 2002.
- [18] K. Minematsu. Improved Security Analysis of XEX and LRW Modes. In *Proceedings of 13th annual workshop on Selected Areas in Cryptography (SAC) 2006*, pages 92–109, August 2006 (preprint).
- [19] R. Chung-Wei Phan and Bok-Min Goi. On the security bounds of CMC, EME, EME+ and EME* modes of operation. In Sihang Qing, Wenbo Mao, Javier Lopez, and Guilin Wang, editors, *ICICS*, volume 3783 of *Lecture Notes in Computer Science*, pages 136–146. Springer, 2005.
- [20] K. Pietrzak. Non-trivial black-box combiners for collision-resistant hash-functions don’t exist. Cryptology ePrint Archive, Report 2006/348, 2006. <http://eprint.iacr.org/>.
- [21] P. Rogaway. Efficient Instantiations of Tweakable Blockciphers and Refinements to Mode OCB and PMAC. In Pil Joong Lee, editor, *Advances in Cryptology – ASIACRYPT ’04*, volume 3329 of LNCS, pages 16–31, 2004.

- [22] P. Rogaway, M. Bellare, J. Black, and T. Krovetz. A Block-Cipher Mode of Operation for Efficient Authenticated Encryption. In *Eighth ACM Conference on Computer and Communications Security (CCS-8)*, pages 196–205. ACM Press, Aug 16 2001. See <http://www.cs.ucdavis.edu/~rogaway/ocb/ocb-doc.htm>.
- [23] R. Schroepfel. The Hasty Pudding Cipher. NIST AES proposal, available <http://www.cs.arizona.edu/rcs/hpc>, 1998.
- [24] Standard Architecture for Encrypted Shared Storage Media, IEEE Project 1619 (P1619). Available at <http://ieee-p1619.wetpaint.com/>.
- [25] P. Wang, D. Feng, and W. Wu. On the security of tweakable modes of operation: TBC and TAE. In Jianying Zhou, Javier Lopez, Robert H. Deng, and Feng Bao, editors, *ISC*, volume 3650 of *Lecture Notes in Computer Science*, pages 274–287. Springer, 2005.

A Attacks on Specific Constructions Cont.

This section is a continuation of the discussion from Section 5.

A.1 OCB

Here we examine OCB mode [22] in order to show that, viewed as a tweakable blockcipher used in a mode of operation, OCB is not security-preserving. OCB is just one direct mode of operation for authenticated encryption from blockciphers, but is the most efficient one known before the Liskov et al. result. OCB similarly uses masking functions.

For OCB the masking function is $f(i, T) = \gamma(i) \oplus \Pi(0) \oplus \Pi(T \oplus \Pi(0))$, where $\gamma(i)$ is a bijection.

Theorem A.1 *For all $i \neq j$ and random X and random but distinct T and T' , the probability that $f(i, T) \oplus f(j, T') = X$ is $\Omega(2^{-n})$.*

Proof. If the equation holds true, we have that $\gamma(i) \oplus \gamma(j) \oplus X = \Pi(T \oplus \Pi(0)) \oplus \Pi(T' \oplus \Pi(0))$. If $X = \gamma_i \oplus \gamma_j$ then this equation cannot be true, however this only occurs with probability 2^{-n} . Otherwise we have the probability of this equation occurring as negligibly close to 2^{-n} as we are giving two distinct inputs to the random permutation. Thus with probability $\Omega(2^{-n})$ the equation will hold. \square

Thus, f likewise has the collision property, and so the OCB tweakable blockcipher is not security-preserving in the SQT model.

We note that both OCB1 mode (with XEX as the tweakable blockcipher) and OCB mode permit indirect attacks along these lines. That is, distinguishing attacks on the tweakable blockcipher can be made indirectly through attacks on the authenticated encryption mode, which gives a lower bound on the tightness that can be shown for those modes of operation. Unfortunately, due to space constraints, we are unable to elaborate on this point.

The generalizations by Minematsu [18] and Chakraborty and Sarkar [4] are similarly based on “masking functions” $f(i, T)$ where for any fixed i , $f(i, \cdot)$ is a permutation, but where for distinct i, j are distinct enough that such collisions can occur.

A.2 EME and its variants

Next we turn our attention to EME mode, developed by Halevi and Rogaway [12] and give a distinguishing attack on this construction. EME does not follow a “mask encrypt mask” paradigm, instead it follows an “encrypt mask encrypt” paradigm for all message blocks except for the first. Here we find that the “mask” is a function of the message and the tweak, namely $M = MP \oplus MC$ where $MP = \Pi(M_1 \oplus L_1) \oplus \dots \oplus \Pi(M_k \oplus L_k) \oplus T$ and MC is $\Pi(MP)$ where L_1 is a value based off of the key and the number of the current message block. Now we explain the attack on EME mode.

Theorem A.2 *EME mode using an ideal blockcipher is not pseudorandom for $O(2^{\frac{n}{2}})$ queries.*

Proof. The distinguishing attack is to query $O(2^{\frac{n}{2}})$ messages, each containing some constant number of message blocks c where the tweaks are distinct, but all messages are the same. If we get a collision where for two messages, all ciphertext blocks after the first are the same, output 1 else output 0. Since we are querying the same message repeatedly, we can say that the probability of a collision $M_i = M_j$ for two different messages is the probability that $\Pi(T \oplus MP) \oplus T = \Pi(T' \oplus MP) \oplus T'$ which occurs with probability 2^{-n} and for q queries will occur with probability $\frac{q^2}{2^n}$ which is non-negligible for $q = O(2^{\frac{n}{2}})$. If a collision on M occurs, the ciphertexts block beyond the first in the two queries should also collide. For a random permutation family, the probability of this occurring is negligibly close to $\frac{q^2}{2^{nc}}$ and so the advantage gained by this attack is $\frac{q^2}{2^n} - \frac{q^2}{2^{nc}}$ which is non-negligible. \square

B Proofs

Here we give detailed proofs for certain theorems in the paper.

B.1 Proof of Theorem 3.2

$\rho(C, i, n)$ represents the number of times the i th primitive in C is invoked when C is queried with security parameter n , and $\tau(C, i, n, q)$ represents the time required to compute q queries to C given an oracle for the i th primitive, while the last $r - i$ primitives must be computed directly, and the first $i - 1$ primitives are replaced by idealized versions that are computed directly.

Theorem 3.2 *For any r , if C is a security-preserving r -primitive construction, and for each $1 \leq i \leq r$, $\text{ADV-DIST}(\Delta_i, \Theta_i, q \cdot \rho(C, i, n), t + \tau(C, i, n, q), n)$ is negligible in n , then $\text{ADV-DIST}(C_\Delta, \Psi, q, t, n)$ is negligible in n .*

Proof. The proof is a simple hybrid argument. Let A be any adversary. $\text{ADV-DIST}(C_\Delta, \Psi, A, n)$ can be bounded by $\text{ADV-DIST}(C_\Theta, \Psi, A, n) + \text{ADV-DIST}(C_\Delta, C_\Theta, A, n)$. The latter of these can be bounded by the triangle inequality in the following way:

$$\text{ADV-DIST}(C_{\Delta_1}, C_\Theta, A, n) \leq \text{ADV}_1(A) + \dots + \text{ADV}_{r-1}(A),$$

where $\text{ADV}_i(A) = \text{ADV-DIST}(C_{\Theta_1, \dots, \Theta_{i-1}, \Delta_i, \dots, \Delta_r}, C_{\Theta_1, \dots, \Theta_i, \Delta_{i+1}, \dots, \Delta_r}, A, n)$. Let R_i^A be the adversary which attempts to distinguish Δ_i from Θ_i . We can define R_i^A in the natural way: it runs A in its attack, and whenever A queries its oracle, R_i^A computes the answer by using Θ_j for $j < i$, Δ_j for $j > i$, and R_i^A 's own oracle as the i th primitive. The overall number of queries R_i^A makes is at most $q \cdot \rho(C, i, n)$, since it makes $\rho(C, i, n)$ queries to its oracle for each C -query A makes. The overall time required to run R_i^A , including running A , is the time that A runs in plus $\tau(C, i, n, q)$. Therefore, $\text{ADV}_i(A) \leq \text{ADV-DIST}(\Delta_i, \Theta_i, q \cdot \rho(C, i, n), t + \tau(C, i, n, q), n)$, which by assumption is negligible in n .

Since $\text{ADV}_i(A)$ is negligible in n for each i and since r is constant, we know that $\text{ADV-DIST}(C_\Delta, C_\Theta, A, n)$ is negligible in n . Since C is security preserving, $\text{ADV-DIST}(C_\Theta, \Psi, A, n)$ is also negligible in n which means that $\text{ADV-DIST}(C_\Delta, \Psi, q, t, n)$ is negligible, which finishes the proof. \square

Note that this proof assumes that the rate $\rho(C, i, n)$ can be given as a well-defined number, which may not be true in all cases. For instance, if C is a hash function that may take arbitrary-sized inputs, the number of times it queries an underlying primitive (a compression function, for instance) may be unbounded. However, in all cases in this paper, the construction will invoke each underlying primitive a constant number of times on every query.

B.2 Proof of Theorem 3.5

Theorem 3.5 *If an r -primitive construction C is security-preserving for an application Φ under a good reduction R , and where for each $1 \leq i \leq r$, $\text{ADV-DIST}(\Delta_i, \Theta_i, q'_i(l, t), t'_i(l, t), n)$ is negligible in n , then $\max_{A \in \mathcal{A}_{l,t}^\Phi} \text{ADV}(\Phi_{C_\Delta}, A, n)$ is negligible in n , where $\mathcal{A}_{l,t}^\Phi = \{A \mid A \text{ uses at most } l \text{ total query length and at most } t \text{ time, and } \text{ADV}(\Phi_\Psi, A, n) \text{ is negligible in } n\}$, for q'_i and t'_i defined as $q'_i(l, t) = q_R(l, t) \cdot \rho(C, i, n)$ and $t'_i(l, t) = t_R(l, t) + \tau(C, i, n, q)$ where $q_R(l, t)$ and $t_R(l, t)$ are defined as the maximum queries and time R^A makes, respectively, given that A makes queries of total length at most l and uses time at most t .*

Proof. Suppose $A \in \mathcal{A}_{l,t}^\Psi$ is an adversary such that $\text{ADV}(\Phi_{C_\Delta}, A, n)$ is not negligible in n , and suppose that A uses a total of at most t time and makes total queries of length at most l , and $\text{ADV}(\Phi_\Psi, A, n)$ is negligible in n .

Since R is a good reduction, we know that $\text{ADV-DIST}(C_\Delta, \Psi, R^A, n) \geq \text{poly}(\text{ADV}(\Phi_{C_\Delta}, A, n))$, so $\text{ADV-DIST}(C_\Delta, \Psi, R^A, n)$ is not negligible in n . By the triangle inequality, $\text{ADV-DIST}(C_\Delta, \Psi, R^A, n) \leq \text{ADV-DIST}(C_\Theta, \Psi, R^A, n) + \text{ADV-DIST}(C_\Delta, C_\Theta, R^A, n)$. Furthermore, since C is security-preserving for Φ with respect to R , and $\text{ADV}(\Phi_\Psi, A, n)$ is negligible in n , we know that $\text{ADV-DIST}(C_\Theta, \Psi, R^A, n)$ is negligible in n .

Therefore, $\text{ADV-DIST}(C_\Delta, C_\Theta, R^A, n)$ is not negligible in n . We can write this advantage as bounded by the sum of advantages $\text{ADV-DIST}(\Delta_i, \Theta_i, R_i^{R^A}, n)$ where $R_i^{R^A}$ is the reduction defined in our proof of Theorem 3.2.

So long as the number of queries made by $R_i^{R^A}$ is at most $q'_i(l, t)$ and the amount of time taken by $R_i^{R^A}$ is at most $t'_i(l, t)$, each of those advantages is negligible by one of our assumptions, which leads to a contradiction, and therefore $\text{ADV}(\Phi_{C_\Delta}, A, n)$ is negligible in n . Since R^A makes at most $q_R(l, t)$ queries and takes at most $t_R(l, t)$ time, we know that $R_i^{R^A}$ makes at most $q'_i(l, t)$ queries and takes at most $t'_i(l, t)$ time, as we argued in the proof of Theorem 3.2. \square

B.3 Proof of Theorem 4.6

Theorem 4.6 *The security preservation of FEF' in the NSQT_l model is at most $\frac{2(l^2-3l)}{2^n}$.*

Proof. We consider a series of hybrid oracles between $\tilde{\Pi}$ and $\text{FEF}'_{R, \Pi, R'}$. First we describe an algorithm for answering $\text{FEF}'_{R, \Pi, R'}$ queries. We assume that R, R', Π , and Π^{-1} are calculated via lazy sampling: on a new input, an output is chosen randomly, on an old input, the old output is returned. Let $\text{FEF}'(1)$ be the following algorithm.

On forward input (M, T) :

1. Calculate $R(T)$, then $\Pi(M \oplus R(T))$, then $R'(T)$.
2. Output $R'(T) \oplus \Pi(M \oplus R(T))$.

On backwards input (C, T) :

1. Calculate $R'(T)$, then $\Pi^{-1}(C \oplus R'(T))$, then $R(T)$.
2. Output $R(T) \oplus \Pi^{-1}(C \oplus R'(T))$.

Note that whenever T is involved in a query (forward or backwards) for the first time, the output is random. Therefore, an equivalent method for answering $\text{FEF}'_{R, \Pi, R'}$ queries is as follows. Let $\text{FEF}'(2)$ be the following algorithm.

On forward input (M, T) :

1. If T has never been queried before, pick and output a random n -bit value C .
2. Otherwise, find (M_0, T, C_0) , the input and result of the prior query involving T .
3. Calculate $R(T)$, $\Pi(M_0 \oplus R(T))$, and let $R'(T) = C_0 \oplus \Pi(M_0 \oplus R(T))$.
4. Calculate $\Pi(M \oplus R(T))$ and output $R'(T) \oplus \Pi(M \oplus R(T))$.

On backwards input (C, T) :

1. If T has never been queried before, pick and output a random n -bit value M .
2. Otherwise, find (M_0, T, C_0) , the input and result of the prior query involving T .
3. Calculate $R(T)$, $\Pi(M_0 \oplus R(T))$, and let $R'(T) = C_0 \oplus \Pi(M_0 \oplus R(T))$.
4. Calculate $\Pi^{-1}(C \oplus R'(T))$ and output $R(T) \oplus \Pi^{-1}(C \oplus R'(T))$.

Note that in this method, $R'(T)$ is still a random value, as it chosen to be $C_0 \oplus \Pi(M_0 \oplus R(T))$, where C_0 was chosen at random.

A third, wholly equivalent method for answering $\text{FEF}'_{R, \Pi, R'}$ queries involves always picking a value at random to output, and then using that value only if the choice does not conflict with prior values of R , Π , or R' . Let $\text{FEF}'(3)$ be the following algorithm:

On forward input (M, T) :

1. If T has never been queried before, pick and output a random n -bit value C .
2. Otherwise, find (M_0, T, C_0) , the input and result of the prior query involving T .
3. Pick C to be a random n -bit value other than C_0 .
4. Calculate $R(T)$, $\Pi(M_0 \oplus R(T))$, and let $R'(T) = C_0 \oplus \Pi(M_0 \oplus R(T))$.
5. If $\Pi(M \oplus R(T))$ and $\Pi^{-1}(C \oplus R'(T))$ are undefined, set $\Pi(M \oplus R(T)) = C \oplus R'(T)$.
Output C .
6. Otherwise, output $R'(T) \oplus \Pi(M \oplus R(T))$.

On backwards input (C, T) :

1. If T has never been queried before, pick and output a random n -bit value M .
2. Otherwise, find (M_0, T, C_0) , the input and result of the prior query involving T .
3. Pick M to be a random n -bit value other than M_0 .
4. Calculate $R'(T)$, $\Pi^{-1}(C_0 \oplus R'(T))$, and let $R(T) = M_0 \oplus \Pi^{-1}(C_0 \oplus R'(T))$.
5. If $\Pi(M \oplus R(T))$ and $\Pi^{-1}(C \oplus R'(T))$ are undefined, set $\Pi(M \oplus R(T)) = C \oplus R'(T)$.
Output M .
6. Otherwise, output $R(T) \oplus \Pi^{-1}(C \oplus R'(T))$.

Recall that if Π , in lazy sampling, picked an output for some input that was already the output of another input, it would simply discard that output and pick a new random one. The above works exactly the same way: it picks a random value $C \oplus R'(T)$ to be the output of $\Pi(M \oplus R(T))$ (if it is not already defined), and if that random value is already the output from some other input, it discards it and tries again. It is merely the order in which these values are chosen that differs.

Another approach is to use this third algorithm, except that in step 6 (for either query direction), the algorithm simply outputs C or M , respectively. Compare this to the algorithm for $\tilde{\Pi}$:

On forward input (M, T) :

1. If T has never been queried before, pick and output a random n -bit value C .
2. Otherwise, output a random n -bit value C other than the C value previously associated

with T .

On backwards input (C, T) :

1. If T has never been queried before, pick and output a random n -bit value C .
2. Otherwise, output a random n -bit value C other than the C value previously associated with T .

It is then made clear that $\tilde{\Pi}$ and $\text{FEF}'(3)$ differ only when $\text{FEF}'(3)$ reaches step 6. Thus, if we can bound the probability that $\text{FEF}'(3)$ reaches step 6, that bound will serve as the security preservation.

Algorithm $\text{FEF}'(3)$ reaches step 6 (on a query in either direction) only when (1) the tweak in the query is being used for the second time, and (2) either $\Pi(M \oplus R(T))$ or $\Pi^{-1}(C \oplus R'(T))$ is already defined. Note that if there have been a total of i tweaks queried twice before the current query, then there are at most $2i$ input/output pairs determined for Π .

Suppose a query is given to $\text{FEF}'(3)$ that uses a tweak for the second time, and is a forward query. Then $R(T)$ is chosen at random, so $M \oplus R(T)$ is a random value, and similarly, C is chosen at random (such that $C \neq C_0$), so $C \oplus R'(T)$ is a random value other than $C_0 \oplus R'(T)$. Thus, the probability that one of these values is part of a defined behavior for Π is at most $\frac{4i-1}{2^n}$. Similarly, if the query is a backwards query, then $R'(T)$ is random, so $C \oplus R'(T)$ is random, and M is random $\neq M_0$ so $M \oplus R(T)$ is a random value $\neq M_0 \oplus R(T)$, and once again, the probability of entering step 6 is $\frac{4i-1}{2^n}$.

In the adversary's entire attack at most l tweaks are queried twice. Thus the probability that $\text{FEF}'(3)$ ever enters step 6 is at most $\sum_{i=1}^l \frac{4i-1}{2^n} = \frac{2(l^2-3l)}{2^n}$. Therefore, the security preservation of FEF' in the NSQT_l model is at most $\frac{2(l^2-3l)}{2^n}$. \square

Note that while FEF' is security preserving for some attack models, it is not absolutely security-preserving. However, it is a secure tweakable blockcipher in the normal sense (unlike, for instance, F from Section 4.1).

Theorem B.1 *ADV-TBC($\text{FEF}'_{R,\Pi,R'}, q, t, n$) is $\Theta(\frac{q^2}{2^n})$ and thus FEF' is not absolutely security-preserving, but is a secure tweakable blockcipher.*

Proof. First, we give the attack that proves the lower bound.

Choose $\frac{q}{8}$ messages and two tweaks T and T' all at random. For each message, compute the ciphertexts $\text{FEF}'_{R,\Pi,R'}(M_i = M \oplus i, T)$ and $\text{FEF}'_{R,\Pi,R'}(M'_i, T')$ for $i = 0, 1, 2, 3$. Now for each message M_i , we see if there is a message M'_j where $\text{FEF}'_{R,\Pi,R'}(M_{i \oplus k}, T) \oplus \text{FEF}'_{R,\Pi,R'}(M_i, T) = \text{FEF}'_{R,\Pi,R'}(M'_{j \oplus k}, T') \oplus \text{FEF}'_{R,\Pi,R'}(M'_j, T')$, for $k = 0, 1, 2, 3$. If so, we output 1.

If the oracle is a random permutation family, the probability that we find such a pair of messages is negligibly close to $\frac{q^2}{2^{4n}} = \frac{q}{2^n}$. In the case of the construction, we only need to find two messages M'_j and M_i such that $M'_j \oplus R(T') = M_i \oplus R(T)$. For such a pair, we have that $\text{FEF}'_{R,\Pi,R'}(M_{i \oplus k}, T) \oplus \text{FEF}'_{R,\Pi,R'}(M_i, T) = \Pi(R(T) \oplus M_i) \oplus \Pi(R(T) \oplus M_i \oplus k) = \Pi(R(T') \oplus M'_j) \oplus \Pi(R(T') \oplus M'_j \oplus k) = \text{FEF}'_{R,\Pi,R'}(M'_j, T') \oplus \text{FEF}'_{R,\Pi,R'}(M'_{j \oplus k}, T')$ for all k . Since we select the messages independently at random, the probability that this occurs is $\frac{q^2}{2^n}$. Thus with probability $\frac{q^2}{2^n} - \frac{q}{2^n}$ an adversary can successfully distinguish between FEF' and a family of random permutation.

We now prove the upper bound. We know from Liskov et al. that FEF is a secure tweakable blockcipher (as a full PRF is certainly ϵ - AXU_2). We can think of $\text{FEF}'(M, T)$ as $\text{FEF}(M, T) \oplus F''(T)$ where $F''(T) = F(T) \oplus F'(T)$, which is itself clearly a pseudorandom function. Therefore, given an

A that attempts to distinguish FEF' from $\tilde{\Pi}$, we can construct A' as follows: pick a pseudorandom function F'' from a family, and run A in its attacks, except applying $\oplus F''(T)$ to the output of every query (or the input of an inverse query). A is thus interacting with something indistinguishable from FEF' , and thus A' has the same advantage as A . But A' cannot have advantage more than $O(q^2/2^n)$ [17]. \square