On Tweaking Luby-Rackoff Blockciphers

David Goldenberg[1], Susan Hohenberger[2⋆], Moses Liskov[1],
Elizabeth Crump Schwartz[1], and Hakan Seyalioglu[1⋆⋆]

[1] The College of William and Mary,
{dcgold,mliskov,eacrum}@cs.wm.edu, hakan.seyalioglu@gmail.com
[2] The Johns Hopkins University, susan@cs.jhu.edu

Abstract. Tweakable blockciphers, first formalized by Liskov, Rivest, and Wagner [13], are blockciphers with an additional input, the *tweak*, which allows for variability. An open problem proposed by Liskov et al. is how to construct tweakable blockciphers without using a pre-existing blockcipher. This problem has yet to receive any significant study. There are many natural questions in this area: is it significantly more efficient to incorporate a tweak directly? How do direct constructions compare to existing techniques? Are these direct constructions *optimal* and for what levels of security? How large of a tweak can be securely added? In this work, we address these questions for Luby-Rackoff blockciphers. We show that tweakable blockciphers can be created directly from Feistel ciphers, and in some cases show that direct constructions of tweakable blockciphers are more efficient than previously known constructions.

1 Introduction

A *blockcipher*, also known as a *pseudorandom permutation*, is a pair of algorithms E and D. The encryption algorithm E takes two inputs – a key K and a message block M, and produces a ciphertext block C of the same length as M, while the decryption algorithm D reverses this process. A blockcipher is considered secure if, for a random secret key K, the cipher is indistinguishable from a random permutation.

A *tweakable blockcipher* takes an extra input, the *tweak* ,(T), that is used only to provide variation and is not kept secret. Unlike changing the key, changing the tweak should involve minimal extra cost. A tweakable blockcipher is considered secure if it is indistinguishable from a family of random permutations indexed by the tweak. The Hasty Pudding Cipher by Schroeppel [23] was the first to introduce an auxiliary blockcipher input called a "spice" and Liskov, Rivest, and Wagner [13] later formalized the notion of tweakable blockciphers. Liskov et al. describe two levels of security: a secure (CPA) tweakable blockcipher is one that is indistinguishable from a random permutation family to any adversary that may make chosen plaintext queries, while a strongly secure (CCA) tweakable blockcipher is pseudorandom even to an adversary that may also make chosen ciphertext queries.

Tweakable blockciphers have many practical applications. Liskov et al. describe how they can be used to implement secure symmetric encryption and authenticated encryption. Halevi and Rogaway [10, 11] suggest an immediate application to private storage where the tweak is set to be the memory address of an enciphered block; and thus, the encryptions of two blocks with the same plaintext are not likely to look the same and yet decryption remains straightforward. Tweakable blockciphers have also been studied in a variety of other contexts [1, 12, 22, 2, 16].

⋆ Supported by an NDSEG Fellowship and NSF grant CT-0716142.
⋆⋆ Partially supported by a Monroe Grant and a Cummings Grant.

Feistel Blockciphers. Feistel blockciphers [7] have been an actively studied class of constructions, since Horst Feistel invented them in 1973. In particular, Luby and Rackoff showed how to construct a pseudorandom permutation from a pseudorandom function by composing three (or four in the case of CCA security) Feistel permutations [14]. We call this construction the Luby-Rackoff blockcipher. In 1996, Lucks [15] described an optimization for the secure 3-round version by replacing the first round with a universal hash function. Shortly afterwards, Naor and Reingold [17] provided the analogous optimization for the strongly secure 4-round cipher, replacing both the first and last rounds with a more general type of function. In 2001, Ramzan [20] formally studied many variations on the Luby-Rackoff cipher. Patarin gave proofs of security for certain constructions against unbounded adversaries with access to exponentially many queries, albeit assuming the individual round functions are random functions rather than pseudorandom. Specifically, Patarin proved security for 7 rounds against $q \ll 2^k$ queries, where the blockcipher input is of size $2k$ [18], and later improved this to show that 5 rounds is sufficient, both for chosen-plaintext and chosen-ciphertext attacks [19], which remains the best proven security level for Feistel ciphers. Dodis and Puniya recently provided a combinatorial understanding of Feistel networks when the round functions are *unpredictable* rather than pseudorandom [6].

Our Work. Liskov, Rivest, and Wagner [13] give two constructions for tweakable blockciphers, each one constructed from an underlying blockcipher. Subsequent work has also taken this approach; Halevi and Rogaway's EMD and EME modes [10, 11] and Rogaway's XEX mode [22] were all blockcipher modes of operation. The only examples of specific tweakable blockciphers are the Hasty Pudding [23] and the Mercy [4] ciphers.

One open problem proposed by Liskov et al. was to study how to incorporate tweaks into existing blockciphers, or design tweakable blockciphers directly. In this work, we perform a systematic study of issues relating to directly tweaking Luby-Rackoff blockciphers. We analyze the approach of including a tweak by XOR-ing the tweak value into one or more places in the dataflow. This natural model for adding a tweak changes the cipher minimally. Also, approaches involving more direct cryptographic processing of the tweak (e.g. hashing the tweak) have a significant additional cost associated with changing the tweak.

Our Contributions. We present tweakable Luby-Rackoff blockciphers, for both CPA and CCA security, and against both polynomial-time adversaries, and against unbounded adversaries with $q \ll 2^k$ queries[1], where k is half the size of the input (matching the best result for ordinary blockciphers [19]). Specifically, we construct tweakable blockciphers:

- CPA-secure against polynomial adversaries in 4 rounds (Theorem 3)
- CCA-secure against polynomial adversaries in 6 rounds (Theorem 8)
- CPA-secure against $q \ll 2^k$ queries in 7 rounds (Theorem 4)
- CCA-secure against $q \ll 2^k$ queries in 10 rounds (Theorem 11)

Recall that for polynomial adversaries CPA-security requires only 3 rounds whereas CCA-security requires 4. It is thus natural to wonder if our constructions are optimal. We prove our constructions against polynomial adversaries are indeed round-optimal in our model (Theorems 1 and 7). Furthermore, we show that any construction of 6 or fewer rounds in our model can be attacked with $O(2^{k/2})$ queries (Table 1), so our construction of Theorem 4 is also round-optimal. In addition, the attacks used to prove the round-optimality of our constructions, as well as our extension of

[1] That is, any non-negative $q < 2^k$ such that $q2^{-k}$ is negligible.

the proof methods of Naor and Reingold, help to form the theoretical foundation necessary for the secure design of tweakable blockciphers regardless of construction, as well as shedding light on the difficulties in adding a tweak to Feistel-based blockciphers such as RC6 [21] and MARS [3].

We also explicitly address the problem of incorporating tweaks of arbitrary length, an important issue not addressed in the literature.[2] We show that our CPA-secure constructions can incorporate additional blocks of tweak at the cost of 1 round per block (Theorems 13 and 16), and that our CCA-secure constructions may be similarly extended at the cost of 2 rounds per block of tweak (Theorems 14 and 17).

2 Definitions

A *tweakable blockcipher* is a triple of algorithms $(\widetilde{G}, \widetilde{E}, \widetilde{D})$ for key generation, encryption, and decryption, respectively. We restrict our attention to tweakable blockciphers where $\widetilde{G}(\cdot)$, $\widetilde{E}_K(\cdot,\cdot)$, and $\widetilde{D}_K(\cdot,\cdot)$ are all efficiently computable algorithms; and where the correctness property holds; that is, for all M, T, and for all keys $K \in \widetilde{G}(1^k)$, $\widetilde{D}_K(\widetilde{E}_K(M,T),T) = M$. We also generally assume that $\widetilde{G}(1^k)$ draws keys uniformly at random from $\{0,1\}^{p(k)}$ for some polynomial p.

We have two notions of security: (1) chosen-plaintext secure (CPA) and (2) chosen-ciphertext secure (CCA). Security is defined in terms of both a polynomial and an exponential adversary; polynomial adversaries are limited to a number of queries and computations polynomial in the message size, whereas an exponential adversary is allowed unlimited computation, but is bounded by an exponential number of queries relative to the message size.

Definition 1. *Over all adversaries with access to an encryption oracle, the maximum advantage is defined as:*

$$\mathsf{ADV\text{-}TBC_K}(\widetilde{E}, \widetilde{D}, q, t) = \max_{\mathcal{A}} : |\Pr[\mathcal{A}^{\widetilde{E}_K(\cdot,\cdot)}(1^k) = 1] - \Pr[\mathcal{A}^{\Pi(\cdot,\cdot)}(1^k) = 1]|$$

where (1) *for all k, K is generated by $\widetilde{G}(1^k)$,* (2) *$\Pi(\cdot,\cdot)$ is a random permutation family parameterized by its second input, and* (3) *$\mathcal{A}$ is allowed to run for t steps and make at most q oracle queries.*

Definition 2. *Over all adversaries with access to an encryption and decryption oracle, the maximum advantage is defined as:*

$$\mathsf{ADV\text{-}STBC_K}(\widetilde{E}, \widetilde{D}, q, t) = \max_{\mathcal{A}} : |\Pr[\mathcal{A}^{\widetilde{E}_K(\cdot,\cdot), \widetilde{D}_K(\cdot,\cdot)}(1^k) = 1] - \Pr[\mathcal{A}^{\Pi(\cdot,\cdot), \Pi^{-1}(\cdot,\cdot)}(1^k) = 1]|$$

where (1) *for all k, K is generated by $\widetilde{G}(1^k)$,* (2) *Π, Π^{-1} are a pseudorandom permutation family and its inverse, and* (3) *$\mathcal{A}$ is allowed to run for t steps and make at most q oracle queries.*

A tweakable blockcipher is CPA secure if for all k, for q queries and time t, $\mathsf{ADV\text{-}TBC_K}(\widetilde{E}, \widetilde{D}, q, t)$ is negligible in k. A tweakable cipher is said to be polynomially-secure if q and t are polynomial in k. If t is unspecified, then it may be unbounded. We define CCA security in the same manner.

[2] Using tweaks of arbitrary length has been considered for tweakable symmetric encryption [9], but not for one-block constructions. Certain applications require different, specific tweak sizes. It may make sense for the tweak size to be the same as the input or output. In TAE mode [13] each tweak holds a variety of information such that each tweak is unique. Thus, one may want to allow longer tweaks to include more information. Indeed, this was the motivation for Schroeppel to allow spice values of 512 bits in the Hasty Pudding Cipher [23].

3 The Feistel Blockcipher

Recall the formula for the Feistel blockcipher [7] on input $M = (L^0, R^0)$:

$$L^{i+1} = R^i$$
$$R^{i+1} = f_{i+1}(R^i) \oplus L^i$$

where the output after n rounds is (L^n, R^n), and each f_i is a pseudorandom function specified by the key. Further recall that the 3-round Feistel construction is secure against chosen plaintext attacks, and the 4-round construction is secure against chosen ciphertext attack [14].

3.1 Notation

In order to talk about where to add a tweak, we must first establish some notation. Unless otherwise specified, the tweaks we refer to are a *half-block* in length; that is, on input M of size $2k$, the tweak is of size k. As we will later see, a blockcipher may allow for longer tweaks; we think of these as "multiple tweaks," as conceptually, the longer tweak can be thought of as being composed of multiple tweaks, each of the same size.

For an n-round Luby-Rackoff construction, a single half-block of tweak can conceivably be XOR-ed in at any of the following unique locations: $\mathcal{L}_0$, $\mathcal{L}_1$, ..., $\mathcal{L}_n$, $\mathcal{R}_0, \mathcal{R}_{0.5}$, $\mathcal{R}_1$, ..., $\mathcal{R}_{n-0.5}, \mathcal{R}_n$. Let this set be denoted by Λ_n. We illustrate the Λ_3 (3-round) locations in Figure 1.

Let T^λ be the XOR of all the tweaks used at location $\lambda \in \Lambda_n$. The formula for our construction is:

$$L^{i+1} = R^i \oplus T^{\mathcal{R}_i}$$
$$R^{i+1} = f_{i+1}(R^i \oplus T^{\mathcal{R}_i} \oplus T^{\mathcal{R}_{i+0.5}}) \oplus L^i \oplus T^{\mathcal{L}_i}$$

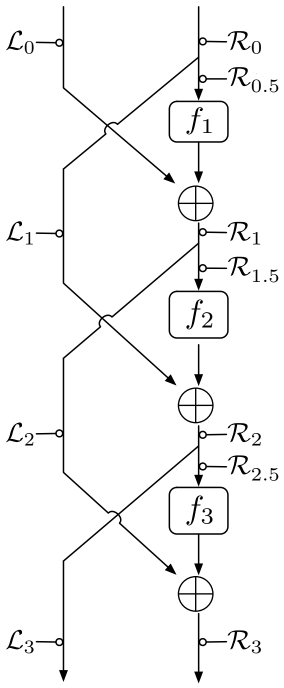

Fig. 1. An illustration of Λ_3; the locations at which to XOR a tweak of length $|M|/2$ for 3-round LR.

We use "$\mathcal{BC}(n, \lambda)$" to refer to the tweakable blockcipher construction where the number of Luby-Rackoff rounds is n and a tweak T^λ is XOR-ed in at some location $\lambda \in \Lambda_n$. To denote adding multiple tweaks, we write "$\mathcal{BC}(n, \lambda_1, \ldots, \lambda_t)$", where $T^{\lambda_i} = T_i$ is the tweak for location λ_i and different locations each have their own independent tweak. Thus, in such a construction, the tweak size is tk.

We might also want to denote adding the *same* tweak value at two or more locations. We write this as "$\mathcal{BC}(n, \lambda_1 + \lambda_2)$", where the implication of using the *compound* location $\lambda_1 + \lambda_2$ is that $T^{\lambda_1} = T^{\lambda_2}$. Of course, we may also consider a construction with multiple tweaks, each of which may be a compound location; we use the obvious notation for this. We use the symbol Γ to denote a (possibly) compound tweak location.

In Λ_n, we have listed all tweaks at ".5" locations, i.e., $\mathcal{R}_{l+0.5}$ for some l. However, we do not have to consider these locations.

Lemma 1. *For all m, $\mathcal{R}_{m+0.5}$ is equivalent to $\mathcal{R}_m + \mathcal{L}_{m+1}$.*

Lemma 2. *For all $0 \le m < n$, $\mathcal{L}_m$ is equivalent to $\mathcal{R}_{m+1}$.*

Since $\mathcal{L}_m$ and $\mathcal{R}_{m+1}$ are equivalent, we will use them interchangeably. This starts us off with a reduced set of tweakable constructions to study including tweaks at locations $\mathcal{L}_n$, $\mathcal{R}_0$, …, $\mathcal{R}_n$ and all combinations thereof.

4 Tweakable Blockciphers With CPA Security

In this section, we focus on achieving CPA security. In the next section, we will discuss the stronger CCA notion of security.

We begin by presenting some general results that hold for an arbitrary number of rounds. These results will help us to narrow down the possibilities for secure constructions and to prove the optimality of our final construction. As stated in Section 3, the set of possibly secure constructions includes those with tweaks at locations $\mathcal{L}_n$, $\mathcal{R}_0$, …, $\mathcal{R}_n$ and all combinations thereof. However, we remark in Lemma 3 that we do not need to consider all possible locations, and that some locations can be simulated without directly tweaking the blockcipher; this important observation is used frequently throughout the paper.

Lemma 3. *For all n, without loss of generality, we can consider only constructions that never use the tweak locations $\mathcal{L}_n$ $\mathcal{R}_n$, $\mathcal{R}_0$, or $\mathcal{R}_1$, even in compound locations, and even when considering CCA security.*

Proof. We can simulate oracle queries with or without the tweaks in $\mathcal{L}_n$ $\mathcal{R}_n$, $\mathcal{R}_0$, or $\mathcal{R}_1$. To simulate a query $(L^0, R^0, T_1, \ldots, T_t)$ to a construction with these tweaks, we make a query $(L^0 \oplus T^{\mathcal{R}_1}, R^0 \oplus T^{\mathcal{R}_0}, T_1, \ldots, T_t)$ to the construction without these tweaks to obtain (L^n, R^n), and we return $(L^n \oplus T^{\mathcal{L}_n}, R^n \oplus T^{\mathcal{R}_n})$. Decryption queries can be simulated similarly. □

The set of tweak locations we need to consider is thus reduced to $\{\mathcal{R}_2 \ldots, \mathcal{R}_{n-1}\}$. From here on, we consider Λ_n to be $\{\mathcal{R}_2, \ldots, \mathcal{R}_{n-1}\}$.

Lemma 4. *For all n, $\mathcal{BC}(n, \mathcal{R}_{n-1})$ is not CPA-secure.*

Proof. We use a 2-query attack. If we query (L, R, T) to get (L_1^n, R_1^n), and then query (L, R, T') to get (L_2^n, R_2^n), then $L_1^n \oplus L_2^n = T \oplus T'$. □

Thus, we arrive at our first round-specific conclusion.

Theorem 1 (No Tweakable 3-Round Constructions). *For all $n < 4$ and all compound locations Γ of elements in Λ_n, $\mathcal{BC}(n, \Gamma)$ is not CPA-secure.*

Proof. This follows from Lemmas 3 and 4, and the set $\{\mathcal{R}_2, \ldots, \mathcal{R}_{n-2}\}$ being empty for $n = 3$. □

4.1 Secure Locations

We have reduced the set of possible secure single tweak locations to $\{\mathcal{R}_2, \ldots, \mathcal{R}_{n-2}\}$. We now show that each of these locations are secure for $n \geq 4$. However, first we must define $\epsilon - \mathsf{ARCU}_2$ hash functions and introduce some related work.

Definition 3. *An $\epsilon - \mathsf{ARCU}_2$ ("Almost Right-Collision-avoiding Universal") hash function family is a hash function family given a range of $\{0, 1\}^{2k}$ with the property that for all $x \neq y$, the probability that $h_R(x) = h_R(y)$ is at most $2^{-k} + \epsilon$, over the choice of h, where h_R denotes the right half of the output of h.*

Naor and Reingold [17] create a secure blockcipher using two Luby-Rackoff rounds in combination with a potentially less expensive function.

Theorem 2 (Naor-Reingold). *If E denotes two Luby-Rackoff rounds with truly random round functions, and h is drawn from an $\epsilon - \mathsf{ARCU}_2$ hash function family, then $E \circ h$ is indistinguishable (in a CPA attack) from a random function.*

Using Definition 3 and Theorem 2, we are able construct CPA-secure tweakable blockciphers.

Theorem 3 (Several Tweakable n-Round Constructions (for $n \geq 4$)). *For all $n \geq 4$ and $m \in \{2, \ldots, n-2\}$, $\mathcal{BC}(n, \mathcal{R}_m)$ is CPA-secure against polynomially bounded adversaries.*

Proof. We can capitalize on Theorem 2 as follows. We will prove that when we let $h(L, R, T) = (L \oplus f_{m-1}(R) || R \oplus T \oplus f_m(L \oplus f_{m-1}(R)))$ over random choice of f_{m-1} and f_m, these conditions hold. Here, h is comprised of the last two rounds of the construction before the tweak, including the tweak. Once we prove this, the result will follow: the first $m-2$ rounds are a permutation, so if h' is comprised of the first m rounds, it will be $\epsilon - \mathsf{ARCU}_2$ if h is. Furthermore, since $m \leq n-2$, there are at least 2 more rounds to follow; any further rounds are another permutation and pseudorandomness will be maintained.

Lemma 5. *The family $h(L, R, T) = (L \oplus f_1(R) || R \oplus T \oplus f_2(L \oplus f_1(R)))$, where f_1 and f_2 are randomly chosen over the domain of all functions from k bits to k bits, is $\epsilon - \mathsf{ARCU}_2$, for $\epsilon = 2^{-k} + 2^{-2k}$.*

Proof. Let $x = (L, R, T)$ and $y = (L', R', T')$, where $x \neq y$. Note that if $R \neq R'$ then the probability that $L \oplus f_1(R) = L' \oplus f_1(R')$ is the probability that $f_1(R) = L \oplus L' \oplus f_1(R')$ which is 2^{-k}. Similarly, if $R = R'$ but $L \neq L'$ then $L \oplus f_1(R) \neq L' \oplus f_1(R')$. In either case, the probability that $L \oplus f_1(R) = L' \oplus f_1(R')$ is at most 2^{-k}. Finally, if $R = R'$ and $L = L'$ then $T \neq T'$ so $h_R(L, R, T) = h_R(L, R, T') \oplus T \oplus T' \neq h_R(L, R, T')$.

The probability that $h_R(L, R, T) = h_R(L', R', T')$ given that $L \oplus f_1(R) \neq L' \oplus f_1(R')$ is the probability that $f_2(L \oplus f_1(R)) = R \oplus R' \oplus f_2(L' \oplus f_1(R')$, which is 2^{-k}, so the probability we hit a collision is at most $(1 - 2^{-k})(2^{-k}) + 2^{-k} = 2^{-k} + 2^{-2k} + 2^{-k} = 2^{-k} + \epsilon$. □

From the Lemma, if all the round functions are random, then the h we are interested in is $\epsilon - \mathsf{ARCU}_2$. By Theorem 2, $\mathcal{BC}(n, \mathcal{R}_m)$ is indistinguishable from a random function if all round functions are random. Therefore, $\mathcal{BC}(n, \mathcal{R}_m)$ must be CPA secure if its round functions are pseudorandom (since random functions are indistinguishable from random permutation families). This completes the proof of Theorem 3. □

Corollary 1 (CPA Security In 4 Rounds). *$\mathcal{BC}(4, \mathcal{R}_2)$ is CPA-secure and round-optimal.*

Proof. This follows directly from Theorems 1 and 3. □

4.2 Exponential Attacks

In this section, we investigate the security of tweakable blockcipher constructions against an adversary who is capable of making an exponential number of queries. We provide general attacks against several types of tweakable constructions built from Luby-Rackoff permutations. In this section, we assume all round functions are ideal, in other words, that they are uniform random functions.[3] We consider a construction secure against exponentially many queries if the probability of any computationally unbounded adversary allowed $q \ll 2^k$ queries to distinguish the construction from a random permutation family is negligible in k. These attacks appertain to constructions with both single and compound tweak locations (where the same tweak value is XOR-ed in multiple locations) and are used to prove that all constructions of less than 7 rounds can be distinguished from a random permutation family in $O(2^{\frac{k}{2}})$ queries.

Lemma 6. *For any $0 \leq r < n$, $\mathcal{BC}(n, \mathcal{R}_{r+0.5})$ is insecure against $O(2^{\frac{k}{2}})$ queries.*

Proof. The attack is as follows: fix the message and query with $2^{\frac{k}{2}}$ different tweaks. The probability that two different queries lead to the same output is negligible for a random permutation family. However, the probability that two queries lead to a collision in this construction is not negligible. On each query, the internal values stay constant until the input to f_{r+1}. Since we have made $2^{\frac{k}{2}}$ queries to an ideal round function, we can expect with non-negligible probability to get a collision on the output of f_{r+1} for two distinct queries. If we get such a collision, notice the entire output ciphertext will collide. □

Corollary 2. *For any $0 \leq r < n$, $\mathcal{BC}(n, \mathcal{R}_{r+0.5} + \mathcal{R}_{r+1})$ is insecure against $O(2^{\frac{k}{2}})$ queries.*

Proof. The attack is identical to that used in Lemma 6, except that instead of expecting a collision of the type $f_{r+1}(R^r \oplus T) = f_{r+1}(R^r \oplus T')$, we expect a collision of the type $f_{r+1}(\mathcal{R}^r \oplus T) \oplus T = f_{r+1}(\mathcal{R}^r \oplus T') \oplus T'$. □

Lemma 7. *For any $0 \leq r < n$, $\mathcal{BC}(n, \mathcal{R}_{r+0.5} + \mathcal{R}_{n-1})$ is insecure against $O(2^{\frac{k}{2}})$ queries.*

Proof. For this proof we will first need a result from probability.

Lemma 8 (Strong Birthday Lemma). *For all $k > 1$, there exists an $m < 1.2 \times 2^{\frac{k}{2}}$ such that if p is the probability of picking an element twice when selecting m elements from a 2^k-element set with replacement uniformly at random, then p and $1-p$ are both non-negligible in k.*

Proof. Let p_i be the probability that some element is picked twice in i tries.

We know that for $m \geq 1.2 \times 2^{k/2}$, $1 - p_m < .5$ (since the birthday threshold is approximately $1.1774 \times 2^{k/2}$). Let m be the first value for which $1 - p_m$ is less than .5, and let $p = p_m$. Then,

$$.5\left(1 - \frac{1.2 \times 2^{k/2}}{2^k}\right) \leq 1 - p \leq .5$$

Since, similarly to our above analysis, $1-p$ will be equal to $(1 - p_{m-1})(1 - \frac{m-1}{2^k})$. Since m is the first value for which $1-p$ is less than .5, $1 - p_{m-1} \leq .5$ and $m - 1 \leq 1.2 \times 2^{k/2}$.

Since $1 - p \leq .5$, we know that $p \geq .5$, so p is non-negligible. But since $1 - p \geq .5(1 - \frac{1.2}{2^{k/2}})$, we know that $1-p$ is non-negligible, as required. □

[3] This is the standard assumption when we want to prove security in a setting where the adversary has beyond-polynomial capabilities [18, 19].

The attack is as follows: Compute the m described in Lemma 8. Keep the message constant and query with m different tweaks. The probability that two ciphertexts are such that $L^n \oplus T = L'^n \oplus T'$ is significantly higher for the actual construction than for a random permutation family. Since $m \leq 1.2 \times 2^{\frac{k}{2}}$, this attack can be performed by an exponential adversary.

Notice that the internal values of any pair of queries are the same up to the input of f_{r+1}. For every query, f_{r+1} receives a different input (as the input is a fixed value XOR-ed by the tweak). Since the round functions are ideal, the event of getting a collision on two outputs of f_{r+1} with m different queries reduces to the event of picking the same element twice as described in Lemma 8; say that probability is p. Notice that if such a collision happens, we always get a collision of the type, $L^n \oplus T = L'^n \oplus T'$.

Assume that the outputs of f_{r+1} are distinct for each of the m queries. Notice that in order to have a collision of two R^{n-2} values, it must be true that the L^{n-2} values differ for both queries, because the intervening rounds act as a permutation. Therefore, we will get a collision on R^{n-2} if and only if we have a collision of the type:

$$f_{n-2}(L^{n-2}) \oplus L^{n-3} = f_{n-2}(L'^{n-2}) \oplus L'^{n-3}.$$

Since the probability of such a collision for any two queries is either 2^{-k} or 0 (in the case that the L^{n-2} values coincide), we can bound the probability of having such a collision above by $\frac{(1.2)^2 2^k}{2 \times 2^k} = .72$ since $m \leq 1.2 \times 2^{\frac{k}{2}}$. Therefore, in this case, with probability greater equal to .28, we can assume all R^{n-2} values are distinct. Notice:

$$L^n \oplus T = L'^n \oplus T' \Leftrightarrow f_{n-1}(R^{n-2}) \oplus L^{n-2} \oplus T = f_{n-1}(R'^{n-2}) \oplus L'^{n-2} \oplus T'.$$

The probability of such an event occurring over m queries with distinct R^{n-2} and ideal round functions is, again, p. Therefore, the overall probability of getting at least two ciphertexts with the described property is at least $p + (1-p)(.28p)$.

If the construction we are given is the random permutation family, the probability of getting the coincidence described is clearly p. Therefore the difference in probabilities of this event happening for the tweakable construction and the random permutation family is at least $p + .28p(1-p) - p = .28p(1-p)$. Since p and $1-p$ are non-negligible in k (by Lemma 8), this value is also non-negligible, and therefore our attack successfully distinguishes the two constructions. $\square$

Corollary 3. *$\mathcal{BC}(n, \mathcal{R}_{r+0.5} + \mathcal{R}_{r+1} + \mathcal{R}_{n-1})$ is insecure against $O(2^{\frac{k}{2}})$ queries.*

Proof. The generalization of Lemma 7 to Lemma 3 is identical to the extension of Lemma 6 to Lemma 2. $\square$

These four attacks can be used to attack every tweakable Luby-Rackoff blockcipher of 6 or fewer rounds. A rundown of which general attack applies for each construction can be found in Table 1. We do not include $\mathcal{L}_1, \mathcal{R}_1, \mathcal{L}_6$ or $\mathcal{R}_6$ in the possible locations, or their equivalent constructions of Table 1 since they can be simulated away by Lemma 3.

4.3 A Tweakable Construction Secure for $q \ll 2^k$ Queries

We now show a 7-round Luby-Rackoff construction that is secure against an adversary allowed $q \ll 2^k$ queries.

Theorem 4. *$\mathcal{BC}(7, \mathcal{R}_3+\mathcal{L}_3)$ is CPA-secure for $q \ll 2^k$ queries.*

Proof. To prove that this construction is a secure tweakable blockcipher we utilize the following theorem from Patarin [18]:

Theorem 5 (Patarin). *Let F be a function from $2k$ bits to $2k$ bits. If F has the property that for $q \ll 2^k$ queries, the probability of having $l > O(k)$ indices such that $R_{i_1} = R_{i_2} = R_{i_3} = ...R_{i_l}$ is negligible, (where R_{i_j} is the right half of the j'th output of F), and on distinct inputs F has only a negligible probability of a full collision on its outputs, then $E \circ F$, (where E is a four-round Luby-Rackoff function), is indistinguishable from random for $q \ll 2^k$ input queries.*

Tweak Locations		
Location	Equivalent	Attack
$\mathcal{R}_2$	$\mathcal{R}_{0.5}$	Lemma 6
$\mathcal{R}_3$	$\mathcal{R}_{1.5}$	Lemma 6
$\mathcal{R}_4$	$\mathcal{R}_{4.5}$	Lemma 6
$\mathcal{R}_5$	N/A	Lemma 4
$\mathcal{R}_2 + \mathcal{R}_3$	$\mathcal{R}_{1.5} + \mathcal{R}_2$	Corollary 2
$\mathcal{R}_2 + \mathcal{R}_4$	$\mathcal{R}_{2.5}$	Lemma 6
$\mathcal{R}_2 + \mathcal{R}_5$	$\mathcal{R}_{0.5} + \mathcal{R}_5$	Lemma 7
$\mathcal{R}_3 + \mathcal{R}_4$	$\mathcal{R}_{3.5} + \mathcal{R}_4 + \mathcal{R}_5$	Corollary 3
$\mathcal{R}_3 + \mathcal{R}_5$	$\mathcal{R}_{3.5}$	Lemma 6
$\mathcal{R}_4 + \mathcal{R}_5$	$\mathcal{R}_{4.5} + \mathcal{R}_5$	Corollary 2
$\mathcal{R}_2 + \mathcal{R}_3 + \mathcal{R}_4$	$\mathcal{R}_{2.5} + \mathcal{R}_3$	Corollary 2
$\mathcal{R}_2 + \mathcal{R}_3 + \mathcal{R}_5$	$\mathcal{R}_{1.5} + \mathcal{R}_2 + \mathcal{R}_5$	Corollary 3
$\mathcal{R}_2 + \mathcal{R}_4 + \mathcal{R}_5$	$\mathcal{R}_{2.5} + \mathcal{R}_5$	Lemma 7
$\mathcal{R}_3 + \mathcal{R}_4 + \mathcal{R}_5$	$\mathcal{R}_{3.5} + \mathcal{R}_4$	Corollary 2
$\mathcal{R}_2 + \mathcal{R}_3 + \mathcal{R}_4 + \mathcal{R}_5$	$\mathcal{R}_{2.5} + \mathcal{R}_3 + \mathcal{R}_5$	Corollary 3

Table 1. All possible 6-round tweakable blockcipher constructions and the corresponding lemmas that prove the constructions are insecure.

We decompose our 7-round construction into two functions, F and E, where F is the first three rounds, including the XOR-ed tweak at both $\mathcal{L}_3$ and $\mathcal{R}_3$,[4] and E is the last four rounds. It is obvious that E is a four-round Luby-Rackoff function. To prove that F has the properties enumerated in Theorem 5, we need to prove the following two properties about F.

Lemma 9. *F is such that for any two distinct queries, the probability of the outputs being equal is $O(2^{-2k})$ and the probability of the right halves of the outputs being equal is $O(2^{-k})$.*

Proof. We show here that given two queries, the probability of an equality in the right half of the output is at most 2^{-k+1}, and that the probability of both outputs being equal is at most 2^{-2k+1}.

We call the two queries L^0, R^0, T and L'^0, R'^0, T' respectively. We also assume that these queries are distinct, that is that either $L^0 \neq L'^0$ or $R^0 \neq R'^0$ or $T \neq T'$. For ease of notation, we define δR^n as $R^n \oplus R'^n$, and $\delta f_i(R^n) = f_i(R^n) \oplus f_i(R'^n)$. We divide this proof down into three cases, $\delta R^0 \neq 0$, $\delta R^0 = 0$ but $\delta L^0 \neq 0$, and finally, $\delta R^0 = \delta L^0 = 0$ and but $\delta T \neq 0$.

Case 1: $\delta R^0 \neq 0$. In order for $\delta R^3 \oplus \delta T = 0$ to be true, (i.e. the right halves of the outputs are equal), we must have that $\delta f_1(R^0) = \delta L^0 \oplus \delta f_3(R^2) \oplus \delta T$. Since $\delta R^0 \neq 0$, $\delta f_1(R^0)$ is a random value. Therefore the probability that the equation is true, which is the probability that the right half of any two outputs are equal, is 2^{-k}.

In order for $\delta L^3 \oplus \delta T = 0$, (i.e. the left halves of the outputs are equal), we must have that $\delta f_2(L^0 \oplus f_1(R^0)) = \delta R^0 \oplus \delta T$. If $\delta L^0 \oplus \delta f_1(R^0) \neq 0$, this occurs with probability 2^{-k}. Furthermore, given this, because $\delta L^0 \oplus \delta f_1(R^0) = \delta R^2$, the probability that $\delta f_3(R^2) = \delta f_1(R^0) \oplus \delta L^0 \oplus \delta T$ is 2^{-k}, and therefore, the probability of a full collision is 2^{-2k}.

[4] Although $\mathcal{L}_3$ is equivalent to $\mathcal{R}_4$, we think of this construction as using $\mathcal{L}_3$, so that we can conceptually split the function this way.

However, $\delta L^0 \oplus \delta f_1(R^0) = 0$ occurs with probability 2^{-k}. In that case, in order to have $\delta L^3 \oplus \delta T = 0$, we must have $\delta T = \delta R^0$. If $\delta R^3 \oplus \delta T = 0$ as well, we know $\delta f_1(R^0) = \delta L^0 \oplus \delta f_3(R^2) \oplus \delta T$, but since $\delta L^0 = \delta f_1(R^0)$ in this case, this implies that $\delta f_3(R^2) = \delta T = \delta R^0 \neq 0$, yet, this can occur with probability at most 2^{-k}. Therefore, the probability of an overall collision is at most $2(2^{-2k}) = 2^{-2k+1}$.

Case 2: $\delta R^0 = 0$ and $\delta L^0 \neq 0$. In order for $\delta R^3 \oplus \delta T = 0$ to hold, we must have that $\delta f_3(R^2) = \delta f_1(R^0) \oplus \delta L^0 \oplus \delta T$ holds. Note that $\delta R^2 = \delta R^0 \oplus \delta f_2(R^1) = \delta f_2(R^1)$, and $\delta R^1 = \delta L^0 \oplus \delta f_1(R^0) = \delta L^0 \neq 0$. If $\delta R^2 \neq 0$, there is a collision on the right only with probability 2^{-k}. However, the probability that $\delta R^2 = 0$ is 2^{-k}, so the probability of a collision on the right is at most $2 \cdot 2^{-k}$.

In order for the $\delta L^3 \oplus \delta T = 0$ to be true, we must have $\delta f_2(L^0 \oplus f_1(R^0)) = \delta R^0 \oplus \delta T = \delta T$. Because $\delta L^0 \neq 0$ and $\delta f_1(R^0) = 0$, $\delta f_2(L^0 \oplus f_1(R^0))$ is random. Therefore the equation is true with probability 2^{-k}. So the probability of the left halves of the output being equal is 2^{-k}.

If the left halves are equal, we know that $\delta f_2(R^1) = \delta T$. Recall that $\delta R^2 = \delta f_2(R^1)$, so if $\delta T = 0$, then $\delta R^2 = 0$ so $\delta R^3 \oplus \delta T = \delta L^0 \neq 0$. However, if $\delta T \neq 0$ then the probability that $\delta R^3 = \delta T$ is at most 2^{-k}. Therefore, the overall probability of a collision in this case is at most 2^{-2k}.

Case 3: $\delta R^0 = 0$ and $\delta L^0 = 0$. This case is trivial. Since the message queries are equal, $\delta R^3 = \delta L^3 = 0$. However, $\delta T \neq 0$, therefore $\delta R^3 \oplus \delta T = \delta L^3 \oplus \delta T \neq 0$. Therefore the outputs are never equal in either half of the output.

The overall probability that two distinct queries will have the same output is at most $O(2^{-2k})$ and the probability that the right half of the outputs will be equal is at most $O(2^{-k})$. Thus, we have proven Lemma 9. ☐

So long as the queries the adversary makes do not produce a full collision on F or a multi-collision on the right half of the output of F, the responses are indistinguishable from random. Therefore, the queries of the adversary are independent of the outputs of F so long as the required conditions hold. By Lemma 9, the probability of an overall collision in $q \ll 2^k$ queries is $O(q^2 2^{-2k})$ which is negligible. Similarly, the probability of an l-way multicollision on the right is $O(q^l 2^{-(l-1)k}) = O(2^k (q 2^{-k})^l)$. Since $q < 2^{k(1-\epsilon)}$ for some ϵ, we know that $(q 2^{-k})^l < (2^{-k\epsilon})^l = 2^{-kl\epsilon}$. If $l \geq k \geq 2/\epsilon$, which will be true for sufficiently large k, this probability is bounded by 2^{-k}. Thus, F satisfies the necessary properties with all but a negligible probability, which completes our proof of Theorem 4. ☐

5 Tweakable Blockciphers With CCA Security

In this section, we study the problem of achieving CCA security. An important observation to make in constructing a CCA-secure tweakable blockcipher is a distinguishing attack we will call the *four-message attack*, which is a type of Boomerang attack [25]. The attack can be performed by any adversary with access to encryption and decryption oracles, E and D respectively. To perform the attack, the adversary makes four queries:

1. For an arbitrary message M and tweak T, obtain $C = E(M, T)$.
2. For an arbitrary tweak $T' \neq T$, obtain $C' = E(M, T')$.
3. Obtain $M' = D(C', T)$.
4. Obtain $C'' = E(M', T')$. If $C = C''$; output 1, otherwise output 0.

A wide class of tweakable blockciphers fall to the four-message attack:

Theorem 6 (Four Message Attack). *Suppose that* $g_1 : \{0,1\}^n \to \{0,1\}^l$ *is an injective function that is invertible on its domain, that* $g_2 : \{0,1\}^t \to \{0,1\}^l$ *is any deterministic function, and that* $g_3 : \{0,1\}^l \to \{0,1\}^n$ *is a function such that for all* C *and* T *there exists a unique* A *such that* $g_3(A \oplus g_2(T)) = C$. *Then the construction* $\widetilde{E}_K(M,T) = g_3(g_2(T) \oplus g_1(M))$ *is not CCA-secure.*

Proof. Note that $C = g_3(g_2(T) \oplus g_1(M))$, $C' = g_3(g_2(T') \oplus g_1(M))$. Now if we decrypt C' with tweak T, we obtain $M' = g_1^{-1}(g_2(T') \oplus g_2(T) \oplus g_1(M))$. When we encrypt M' under tweak T', we get $C'' = g_3(g_2(T') \oplus g_1(g_1^{-1}(g_2(T') \oplus g_2(T) \oplus g_1(M))) = g_3(g_2(T') \oplus g_2(T') \oplus g_2(T) \oplus g_1(M)) = g_3(g_2(T) \oplus g_1(M)) = C$. □

Note in particular that if both g_1 and g_3 are permutations, the conditions are satisfied. This has immediate consequences:

Corollary 4. *For all* $n, \mathcal{R}_m \in \Lambda_n$, *both* $\mathcal{BC}(n, \mathcal{R}_m)$ *and* $\mathcal{BC}(n, \mathcal{R}_m + \mathcal{R}_{m+1})$ *are not CCA-secure.*

Proof. Here, g_1 is the permutation described by the m rounds of Luby-Rackoff before the tweak, $g_2(T) = 0^k||T$ for $\mathcal{BC}(n, \mathcal{R}_m)$ and $g_2(T) = T||T$ for $\mathcal{BC}(n, \mathcal{R}_m + \mathcal{R}_{m+1})$, and g_3 is the remaining $n - m$ rounds. Clearly g_1 and g_3 are permutations, so the four message attack applies. □

This shows that if we are to be able to add a half-block of tweak to the construction anywhere, it must be used at multiple locations, and those locations must be separated by at least one round.[5] In fact, however, a one round distance will not suffice:

Lemma 10. *For all* $n, \mathcal{R}_m \in \Lambda_n$, $\mathcal{BC}(n, \mathcal{R}_m + \mathcal{R}_{m+2})$ *is not CCA-secure, and* $\mathcal{BC}(n, \mathcal{R}_m + \mathcal{R}_{m+1} + \mathcal{R}_{m+2})$ *is also not CCA-secure.*

Proof. To simplify, recall that $\mathcal{R}_m + \mathcal{R}_{m+2}$ is equivalent to $\mathcal{R}_{m+0.5}$ by Lemma 1. Noticing this makes it clear why this is unlikely to be secure, in light of the previous two corollaries, but we still have some work to do.

Here, we use the four-message attack again, but this time, suppose g_1 and g_3 are not permutations. Rather, if (L, R) is the output of the first m rounds of the Luby-Rackoff permutations, then $g_1(M)$ is the $3k$ bit response (L, R, R). Notice that $g_2(T)$ is $0^{2k}||T$, and $g_3(A, B, C)$ computes the remaining rounds, computing $L^{m+1} = B$ and $R^{m+1} = f_m(C) \oplus A$, and continuing from there. Note that $g_3(g_2(T) \oplus g_1(M))$ is the output we get from applying $\mathcal{BC}(n, \mathcal{R}_{m+0.5})$ to M with tweak T. For the $\mathcal{BC}(n, \mathcal{R}_m + \mathcal{R}_{m+1} + \mathcal{R}_{m+2})$ construction, this is just the same as $\mathcal{BC}(n, \mathcal{R}_{m+0.5} + \mathcal{L}_m)$, and change g_2 so that it produces $T||0^k||T$ rather than $0^{2k}||T$. Clearly g_1 is injective and invertible, and g_3 has unique inverses of the proper form, which we can find by inverting the tweakable blockcipher and noting the values in the proper place. Doing so requires the tweak T, but the answer is unique regardless, or we wouldn't have unique decryption. By Theorem 6, neither of these constructions are CCA-secure. □

Theorem 7. *For all* $n < 6$ *and all compound locations* Γ *of elements in* Λ_n, $\mathcal{BC}(n, \Gamma)$ *is not CCA-secure.*

Proof. In order to construct a CCA-secure tweakable blockcipher, we must use the tweak at (minimally) $\mathcal{R}_m$ and $\mathcal{R}_{m+d}$ for some $d \geq 3$. And naturally, m and $m+d$ must be in the range $2, \ldots, n-1$ since all other locations can be simulated. For $n \leq 5$ no such pair of locations exists. □

[5] This shows, along with Lemma 10, that an adversary making a CCA attack with XOR injection will be able to succeed, regardless of the location of the XOR.

Therefore, the first construction that can be CCA-secure is $\mathcal{BC}(6, \mathcal{R}_2 + \mathcal{R}_5)$, and is in fact a secure construction!

Theorem 8. *$\mathcal{BC}(6, \mathcal{R}_2 + \mathcal{R}_5)$ is a CCA-secure tweakable blockcipher.*

Proof. We will actually prove a slightly stronger version of this theorem:

Theorem 9. *If $2 \leq m_1 \leq m_2 - 3 \leq n - 4$ then $\mathcal{BC}(n, \mathcal{R}_{m_1} + \mathcal{R}_{m_2})$ is a CCA-secure tweakable blockcipher.*

Proof. Again, we will modify the proof from Naor and Reingold to prove that our construction is CCA-secure. First, we define the following.

Definition 4. *An $\epsilon - \mathsf{ALICU}_2$ ("Almost Left-Inverse-Collision-avoiding Universal") hash function family is a hash function family with the property that for all $x \neq y$, the probability that $h_L^{-1}(x) = h_L^{-1}(y)$ is at most $2^{-k} + \epsilon$, where h is chosen randomly from the family, and h_L^{-1} denotes the left half of the output of h^{-1}.*

The following theorem is proven by Naor and Reingold:

Theorem 10. *If h_1 is drawn from an $\epsilon - \mathsf{ARCU}_2$ family of hash functions and h_2 is drawn from an $\epsilon - \mathsf{ALICU}_2$ family of hash functions, and E is two rounds Luby-Rackoff with a random round function, then the pair of oracles $(h_2 \circ E \circ h_1, h_1^{-1} \circ E^{-1} \circ h_2^{-1})$ are indistinguishable from random.*

Since we know that 2 rounds of Luby-Rackoff with a random round function followed by a tweak is $\epsilon - \mathsf{ARCU}_2$ (by Lemma 5), all we need to prove is that two rounds with a random round function *preceded* by a tweak is $\epsilon - \mathsf{ALICU}_2$.

Lemma 11. *The function family defined by $h(L, R, T) = (L \oplus T \oplus f(R), R \oplus f'(L \oplus T \oplus f(R)))$, where f and f' are random functions, is $\epsilon - \mathsf{ALICU}_2$, for $\epsilon = 2^{-n} + 2^{-2n}$*

Proof. Let (L, R, T) be inputs to h^{-1}. The outputs will be $h_L^{-1} = L \oplus T \oplus f(R \oplus f'(L))$, and $h_R^{-1} = R \oplus f'(L)$. In other words, h^{-1} is actually just like the h function from Lemma 5, except with f' as the first random function and f as the second one, and with the right and left halves switched. Thus, h is $\epsilon - \mathsf{ALICU}_2$ from Lemma 5. □

Once we have this, we merely note that we have the properties needed by the Naor and Reingold proof to establish that our construction, with random functions in place of pseudorandom ones, is indistinguishable from a random function.

If we think of h_1 as the first $m_2 - 2$ rounds of $\mathcal{BC}(n, \mathcal{R}_{m_1} + \mathcal{R}_{m_2})$ for $2 \leq m_1 \leq m_2 - 3$, then h_1 is $\pi_1 \circ h_1' \circ \pi_2$ where π_1 and π_2 are permutations, and h_1' is an $\epsilon - \mathsf{ARCU}_2$ hash function, so h_1 is also $\epsilon - \mathsf{ARCU}_2$. If we think of h_2 as the last $n - m_2$ rounds of $\mathcal{BC}(n, \mathcal{R}_{m_1} + \mathcal{R}_{m_2})$ for $m_2 \leq n - 1$, then h_2 is $\pi_3 \circ h_2'$ where h_2' is an $\epsilon - \mathsf{ALICU}_2$ hash function, so h_2 is also $\epsilon - \mathsf{ALICU}_2$. Since $\mathcal{BC}(n, \mathcal{R}_{m_1} + \mathcal{R}_{m_2})$ for $2 \leq m_1 \leq m_2 - 3 \leq n - 4$ is equal to $h_2 \oplus h_1$, where h_1 is an $\epsilon - \mathsf{ARCU}_2$ function and h_2 is an $\epsilon - \mathsf{ALICU}_2$ function, this construction is CCA-secure. □

In particular, $2 \leq 2 \leq 5 - 3 \leq 6 - 4$, so $\mathcal{BC}(6, \mathcal{R}_2 + \mathcal{R}_5)$ is CCA-secure. □

5.1 CCA Security Against Exponential Attacks

Theorem 11. *$\mathcal{BC}(10, \mathcal{L}_3 + \mathcal{R}_3 + \mathcal{L}_7 + \mathcal{R}_7)$ is CCA-secure for $q \ll 2^k$ queries.*

Proof. In order to construct a tweakable blockcipher secure against CCA exponential attacks, we use a theorem of Patarin [19]:

Theorem 12 (Patarin). *Let F and F' be functions from $2k$ bits to $2k$ bits. If F and F'^{-1} each have the property that for $q \ll 2^k$ queries, the probability of having $l > O(k)$ indices such that $R_{i_1} = R_{i_2} = R_{i_3} = ...R_{i_l}$ is negligible, (where R_{i_j} is the right half of the j'th output of F or F'^{-1}), and on distinct inputs F (and F'^{-1}) has only a negligible probability of a full collision on its outputs, then $F' \circ E \circ F$,(where E is a four-round Luby-Rackoff function), is indistinguishable from random against chosen-ciphertext attack for $q \ll 2^k$ input queries.*

In our construction, the first three rounds, including the tweaks at $\mathcal{L}_3$ and $\mathcal{R}_3$, form F, and the last three rounds, including the tweaks at $\mathcal{L}_7$ and $\mathcal{R}_7$, form F'. F'^{-1} is just the same as F, except with distinct round functions. Both F and F'^{-1} meet the properties of Theorem 12, as we have shown in our proof of Lemma 9. $\mathcal{BC}(10, \mathcal{L}_3 + \mathcal{R}_3 + \mathcal{L}_7 + \mathcal{R}_7) = F' \circ E \circ F$, and is therefore CCA-secure against $q \ll 2^k$ queries. □

6 Allowing Longer Tweaks

In our previous results, all tweaks were assumed to be a half block in length. It may be desirable however, to have tweaks of arbitrary lengths. We can always lengthen a tweak that is less than a half block, by padding it in a deterministic way. However, increasing the length of a tweak beyond a half block in length does not follow easily. It may be useful to have constructions that are still secure with longer tweaks, as one usual way of choosing a tweak is to include data with it that makes it unique [23]. The longer the tweak, the more data can be included.

In this section, we demonstrate that secure (CPA-secure) tweakable blockciphers exist with arbitrary tweak length at the cost of one additional round per half-block of tweak, and that the constructions we give are round-optimal. We then demonstrate in Section 6.2 that CCA-secure tweakable blockciphers exist with arbitrary tweak length at the cost of two additional rounds per half-block of tweak. (The optimality of this construction is an open problem.)

First, we prove several lemmas about multiple tweak and compound tweak locations. We adopt the notation that Λ_n^* is the set of all compound tweak locations over Λ_n.

Lemma 12. *For all n and $\Gamma_1, \ldots, \Gamma_t \in \Lambda_n^*$, if $\mathcal{BC}(n, \Gamma_1, \ldots, \Gamma_t)$ is secure, then for all $i = 1$ to t, $\mathcal{BC}(n, \Gamma_i)$ is secure.*

Proof. Suppose not; let $j \in [1, t]$ be such that $\mathcal{BC}(n, \Gamma_j)$ is insecure. We can attack $\mathcal{BC}(n, \Gamma_1, \ldots, \Gamma_t)$ by following the attack on $\mathcal{BC}(n, \Gamma_j)$, but setting all tweaks other than T_j to 0^k. □

We can define $\Gamma = \sum_{i \in S_\Gamma} \lambda_i$, where S_Γ is the set of locations used in Γ. If we do so, then clearly $\Gamma + \Gamma' = \sum_{i \in S_\Gamma \Delta S_{\Gamma'}} \lambda_i$ where Δ represents symmetric difference. We now show a generalization of Lemma 12.

Lemma 13. *For all n and $\Gamma_1, \ldots, \Gamma_t \in \Lambda_n^*$, if $\mathcal{BC}(n, \Gamma_1, \ldots, \Gamma_t)$ is secure, then for all $\emptyset \neq S \subset \{1, \ldots, t\}$, $\mathcal{BC}(n, \sum_{i \in S} \Gamma_i)$ is secure.*

Proof. If not, let S be such that $\mathcal{BC}(n, \sum_{i\in S} \Gamma_i)$ is insecure. We can attack $\mathcal{BC}(n, \Gamma_1, \ldots, \Gamma_t)$ by following the attack on $\mathcal{BC}(n, \sum_{i\in S} \Gamma_i)$ by setting all tweaks T_j for $j \notin S$ equal to 0, and all tweaks T_i for $i \in S$ equal to each other. □

Lemma 14 (Combinations With The Same Tweak). *For all n and $\lambda_1, \ldots, \lambda_r \in \Lambda_n$, $\mathcal{BC}(n, \lambda_1 + \ldots + \lambda_r)$ is secure, then $\lambda_i \in \{\mathcal{R}_2, \ldots, \mathcal{R}_{n-2}\}$ for some $1 \le i \le r$.*

Proof. Since without loss of generality, all λ_i are in $\{\mathcal{R}_2, \ldots, \mathcal{R}_{n-1}\}$, the only way the condition is not met is if all λ_i are $\mathcal{R}_{n-1}$. If r is even, the construction is equivalent to $\mathcal{BC}(n, \emptyset)$, while if r is odd, the construction is equivalent to $\mathcal{BC}(n, \mathcal{R}_{n-1})$, both of which are insecure. □

These three lemmas apply for any type of security.

6.1 CPA-Secure Tweakable Blockciphers with Longer Tweaks

Now we prove our result.

Theorem 13. *For all n, one can use $n-3$ half-blocks of tweak but no more. Specifically, $\mathcal{BC}(n, \mathcal{R}_2, \ldots, \mathcal{R}_{n-2})$ is secure, but any construction $\mathcal{BC}(n, \Gamma_1, \ldots, \Gamma_t)$ for $t > n-3$ is not secure.*

Proof. First we prove that no construction with more than $n-3$ half-blocks of tweak can be secure. Consider the construction $\mathcal{BC}(n, \Gamma_1, \ldots, \Gamma_t)$. Following from the proof of Lemma 14, we can assume without loss of generality that each Γ_i contains each location in Λ_n at most once. Therefore, we can think of each Γ_i as a vector of boolean coefficients $\boldsymbol{a}_i = (a_{2,i}, \ldots a_{n-1,i})$ where $a_{m,i} = 1$ implies that $\mathcal{R}_m$ is included in Γ_i. Since there are more than $n-3$ of these vectors, we can find a nontrivial linear combination of vectors $\sum_{i=1}^t b_i \boldsymbol{a}_i$ such that the sum is zero everywhere except the last term, and such that not all b_i are 0. Let S be the subset of $\{1, \ldots, t\}$ such that $b_i = 1$ for all $i \in S$.

If the sum $\sum_{i=1}^t b_i \boldsymbol{a}_i$ is 0, we can break the cipher by querying $(L, R, 0^{tk})$ and (L, R, T) where T is a tweak that uses the same half-block tweak T_0 for each of the compound locations corresponding to S, but is 0^k for all other tweaks. The outputs will be the same in either case. If the sum is 1, we can make the same two queries to obtain (L_1^n, R_1^n) and (L_2^n, R_2^n). In this case, $L_1^n \oplus L_2^n$ will be T_0.

Next we prove that $\mathcal{BC}(n, \mathcal{R}_2, \mathcal{R}_3, \ldots, \mathcal{R}_{n-2})$ is secure. Following our proof of Theorem 3, we let h be the function that represents the first $n-2$ rounds, including all the tweaks. We need only prove that h generated this way is $\epsilon - \mathsf{ARCU}_2$. We can prove this by induction, regarding Lemma 5 as the base case. The inductive step will be:

Lemma 15. *If $h(L, R, T)$ denotes a random member of an $\epsilon - \mathsf{ARCU}_2$ family, where h_L and h_R denote the left and right half of h, respectively, then $h'(L, R, T, U) = (h_R(L, R, T), h_L(L, R, T) \oplus f(h_R(L, R, T) \oplus U)$ is also $\delta - \mathsf{ARCU}_2$ for randomly chosen function f, for $\delta = (\epsilon + 2^{-n})(1 - 2^{-n})$*

Proof. Let $x = (L, R, T, U)$ and let $y = (L', R', T', U')$ such that $x \neq y$.

If $(L, R, T) \neq (L', R', T')$ then the probability that $h_R(L, R, T) = h_R(L', R', T')$ is at most $2^{-n} + \epsilon$. Given that $h_R(L, R, T) \neq h_R(L', R', T')$, the probability that $h'_R(x) = h'_R(y)$ is the probability that $f(h_R(L, R, T)) = U \oplus U' \oplus f(h_R(L', R', T'))$, which is 2^{-n}. On the other hand, if $(L, R, T) = (L', R', T')$ then $h_R(x) = h_R(y) \oplus U \oplus U' \neq h_R(y)$.

Thus the probability of a collision is at most $(2^{-n} + \epsilon) + (1 - (2^{-n} + \epsilon))(2^{-n}) = 2^{-n} + \delta$. □

Thus, the h we are interested in is $\epsilon - \mathsf{ARCU}_2$, for $\epsilon < n2^{-n}$. By the proof of Naor and Reingold, $\mathcal{BC}(n, \mathcal{R}_2, \ldots, \mathcal{R}_{n-2})$ is secure. □

6.2 CCA-Secure Tweakable Blockciphers with Longer Tweaks

It is not hard to increase the length of tweaks by generalizing the $\mathcal{BC}(6, \mathcal{R}_2 + \mathcal{R}_5)$ construction.

Theorem 14. *For all n, the tweakable blockcipher $\mathcal{BC}(2n, \mathcal{R}_2 + \mathcal{R}_{2n-1}, \mathcal{R}_3 + \mathcal{R}_{2n-2}, \dots, \mathcal{R}_{n-1} + \mathcal{R}_{n+2})$ is a CCA-secure tweakable blockcipher.*

Proof. The key point in the proof is that we can still conceptually divide the construction into three phases: $h_2 \circ E \circ h_1$, where this time, h_1 represents the first $n-1$ rounds, including the tweaks, and h_2 represents the last $n-1$ rounds, and E represents the middle 2 rounds.

We know that h_1 is $\epsilon - \mathsf{ARCU}_2$ from Lemma 15, and proving that h_2 is $\epsilon - \mathsf{ALICU}_2$ follows from much the same proof as is given for Lemma 11. □

6.3 Longer Tweaks with Exponential Security

In this section we focus on the problem of constructing a Luby-Rackoff tweakable blockcipher secure against an unbounded adversary with $q \ll 2^k$ queries. For t half-blocks of tweak, we show how to construct a Luby-Rackoff based tweakable blockcipher in $t+6$ rounds that meets this security goal. The construction is based on a $t+2$-round function F designed to meet the properties required by Patarin.

Theorem 15. *Let $\mu_i = \mathcal{L}_{i+2}$ if $i \equiv 1$ or $i \equiv 2 \bmod 4$, let $\mu_i = \mathcal{L}_{i+2} + \mathcal{L}_1$ if $i \equiv 3 \bmod 4$, and $\mu_i = \mathcal{L}_{i+2} + \mathcal{L}_2$ if $i \equiv 0 \bmod 4$. Let $\mu'_i = \mu_i + \mathcal{R}_i$ if $i \not\equiv 2 \bmod 4$, and $\mu'_i = \mu_i + \mathcal{R}_i + \mathcal{L}_1$ otherwise. Then let F be $\mathcal{BC}(n+2, \mu_1, \dots, \mu_{n-1}, \mu'_n)$. F is a function such that for $q \ll 2^k$ queries, the probability of having $l = O(k)$ indices such that $R_{i_1} = R_{i_2} = R_{i_3} = \dots R_{i_l}$ is negligible, (where R_{i_j} is the right half of the j'th output of F), and on q distinct inputs F has only a negligible probability of a full collision on its outputs.*

Proof. To clarify, the construction of F for all $i \leq t$, uses tweak T_i at $\mathcal{L}_{i+2}$, and also at the following locations:

- If $i = t$, then at $\mathcal{R}_{t+2}$,
- If $i \equiv 3 \bmod 4$, then at $\mathcal{L}_1$.
- If $i \equiv 0 \bmod 4$, then at $\mathcal{L}_2$.
- If $i = t$ and $i \equiv 2 \bmod 4$, then at $\mathcal{L}_1$.

Before we give the proof, it will be helpful to explain a little of the intuition behind our construction. The basic idea is that each tweak is included at its own round; they are included on the left only for simplicity of presentation: apart from the tweak included at $\mathcal{L}_{t+2}$, all tweaks could be included on the right instead.

Tweaks included at locations involving only $\mathcal{L}_1$ and $\mathcal{L}_2$ allow for a full collision attack, as we explain in Section 6.4, so those locations are not useful as the sole location of a tweak. However, if two tweaks were to appear at $\mathcal{L}_i$ and $\mathcal{L}_{i+2}$, respectively, we could set them equal to each other; this would be equivalent to having one fewer tweak, but with one tweak at $\mathcal{R}_{i+1.5}$, which leads to the attack described in Lemma 6. So, to avoid that kind of attack, we use $\mathcal{L}_1$ and $\mathcal{L}_2$ to ensure that no two tweaks are used *only* at two individual locations that are two apart. We include every other odd tweak (starting with T_3) at $\mathcal{L}_1$ and every other even tweak (starting with T_4) at $\mathcal{L}_2$.

With those locations, F would be successful at preventing full collisions, but large multi-collisions on the right could be forced: simply put, the last tweak may not affect the right half of the output. Therefore, we include T_t at $\mathcal{R}_{t+2}$, so that every block of tweak affects the right half of the output. However, this leads to an attack if we aren't careful: if T_t is used at $\mathcal{R}_{t+2}$ alone, and T_{t-1} is used at $\mathcal{L}_{t+1}$ alone, they effectively occur at the same spot. Our solution is to force one of the two to appear at either $\mathcal{L}_1$ or $\mathcal{L}_2$ while the other does not: we do this by adjusting the only case that is a problem, namely when $t \equiv 2 \bmod 4$. □

We now prove that the F we have given above has the properties we need. Define T_0 to be $\oplus_{i\equiv 0 \bmod 4} T_i$. If $t \equiv 2 \bmod 4$ then define T_{-1} to be $T_t \oplus_{i\equiv 3 \bmod 4} T_i$; otherwise, define T_{-1} to be $\oplus_{i\equiv 3 \bmod 4} T_i$. Define $T_{ev} = \oplus_{i=0}^{\lfloor t/2 \rfloor} T_{2i}$, and define $T_{od} = \oplus_{i=-1}^{\lfloor (t+1)/2 \rfloor} T_{2i+1}$. Define T_{te} to be T_t if t is even, and 0 otherwise; similarly, define T_{to} to be T_t if t is odd, and 0 otherwise. We are also using the δ notation where $\delta R^i = R^i \oplus R'^i$. $\delta f_i(R^i) = f_i(R^i) \oplus f_i(R'^i)$ and δL^i and δT^i are defined similarly.

First, we focus on the probability of a full collision on two distinct queries.

Lemma 16. *On any pair of distinct inputs, the probability that F will produce the same output on each is $O(2^{-2k})$.*

Proof. In order for two queries to yield a collision of F, we must have the following two equations.[6]

$$\begin{aligned} 0 &= \delta R^0 \oplus \delta T_{te} \oplus \delta T_{od} \oplus \delta f_2(R^1) \oplus \delta f_4(R^3) \oplus \ldots \oplus \delta f_{2\lfloor t/2 \rfloor+2}(R^{2\lfloor t/2 \rfloor+1}) \\ 0 &= \delta L^0 \oplus \delta T_{to} \oplus \delta T_{ev} \oplus \delta f_1(R^0) \oplus \delta f_3(R^2) \oplus \ldots \oplus \delta f_{2\lfloor (t+1)/2 \rfloor+1}(R^{2\lfloor (t+1)/2 \rfloor}) \end{aligned}$$

Note that R^i is involved in one of the two equations above for every $0 \le i \le t-1$; R^i for odd i are involved in the first equation, while R^i for even i are involved in the second equation. We consider three cases.

Case i: There is an even $i < t+2$ such that $\delta R^i \neq 0$, and there is an odd $j < t+2$ such that $\delta R^j \neq 0$. In this case, the probability of both equations holding is 2^{-2k}.

Case ii: For all $i < t+2$, $\delta R^i = 0$. If this is the case, it is easy to see that $\delta R^0 = 0$, and $\delta L^0 = 0$ (since $\delta R^1 = \delta f_1(R^0) \oplus \delta L^0$). Furthermore, for each $1 \le i \le t+1$, $0 = \delta R^{i+1} = \delta f_{i+1}(R^i) \oplus \delta L^i = 0 \oplus \delta L^i = \delta R^{i-1} \oplus \delta T_{i-2} = \delta T_{i-2}$. Therefore, all the tweak values must also remain constant up to T_{t-1}. But if both L^0 and R^0 are the same, and all the tweak values up to T_{t-1} are the same, then the difference on the right of the output will be the difference in T_t, so if there is a collision, $T_t = 0$. Therefore, both queries will be the same. So the probability of a collision in this case is 1, but the probability of two distinct queries leading to this case is 0.

Case iii: Either for all odd $i < t+2$ or for all even $i < t+2$, $\delta R^i = 0$, but there is some $j < t$ such that $\delta R^j \neq 0$. This covers all remaining cases, but this case is one to worry about. A priori, one of the two equations may be true with probability 1, while the other is true with probability 2^{-k}. However, as we will see, either one of a small number of unlikely events occurs, or an overall collision cannot occur unless the two inputs were identical.

Lemma 17. *In two distinct queries to F for which either for all odd $i < t+2$ or for all even $i < t+2$, $\delta R^i = 0$, the probability of a full collision on F is $O(2^{-2k})$.*

Proof. Let j be such that $\delta R^j \neq 0$ but for all $i > j$ of the same parity, $\delta R^i = 0$. Suppose, without loss of generality, that j is even; the case when j is odd is very similar.

[6] Recall our notation: if X represents an internal value computed during a query, then δX denotes the difference in X values between two specific queries.

Since for all odd $i < t+2$, $\delta R^i = 0$, we learn that if $i \geq 2$ is odd, then

$$\begin{aligned} 0 &= \delta R^i \\ &= \delta f_i(R^{i-1}) \oplus \delta L^{i-1} \\ &= \delta f_i(R^{i-1}) \oplus \delta T_{i-3} \oplus \delta R^{i-2} \\ &= \delta f_i(R^{i-1}) \oplus \delta T_{i-3} \end{aligned}$$

And therefore, $\delta f_i(R^{i-1}) = \delta T_{i-3}$. If $i-1 < j$, then for this equation to be satisfied, we either must have a rare event (that $\delta R^{i-1} \neq 0$ but the random outputs happen to have a pre-specified difference), or $\delta R^{i-1} = 0$. If $i-1 > j$, then by our choice of j, $\delta R^{i-1} = 0$. Therefore, all for all even $i-1$ other than j or 0, $\delta R^{i-1} = 0$, or two rare events must occur. We also know that $\delta R^0 = 0$, because

$$\begin{aligned} 0 &= \delta R^1 \\ &= \delta f_1(R^0) \oplus \delta L^0, \end{aligned}$$

so $\delta f_1(R^0) = \delta L^0$. Again, this can only occur without a rare event if $\delta R^0 = 0$.

If none of these rare events occur, then $\delta L^0 = 0$, and $\delta T_{i-3} = 0$ for all odd $2 \leq i < t+2$ other than $j-2$. Furthermore, if i is even such that $\delta R^i = 0$ and $\delta R^{i-2} = 0$, then we can conclude that $\delta T_{i-3} = \delta f_i(R^{i-1}) \oplus \delta R^{i-2} = 0$, by the above deduction, and because $i-1$ is odd.

Thus we have learned that for most i, $\delta T_i = 0$. The exceptions are for $i \in \{j-3, j-2, j-1, t-1, t\}$. For those, we still know something:

- Since $\delta R^j = \delta f_j(R^{j-1}) \oplus \delta R^{j-2} \oplus \delta T_{j-3}$, and $\delta R^{j-1} = \delta R^{j-2} = 0$, we know $\delta T_{j-3} = \delta R^j$.
- Since $0 = \delta R^{j+2} = \delta f_{j+2}(R^{j+1}) \oplus \delta R^j \oplus \delta T_{j-1}$, we know $\delta T_{j-1} = \delta R^j$.
- Similarly, we know that $0 = \delta R^{j+1} = \delta f_{j+1}(R^j) \oplus \delta R^{j-1} \oplus \delta T_{j-2}$ so $\delta T_{j-2} = \delta f_{j+1}(R^j)$.

We also know that $\delta L^0 = 0$ since $\delta L^0 = \delta R^1 \oplus \delta f_1(R^0)$ (except if $j = 0$, which we will handle as a special case.)

Note that because the two output halves are equal, this lets us conclude that $\delta T_t = 0$ and that $\delta T_{t-1} = 0$. Recall that:

$$\begin{aligned} 0 &= \delta R^0 \oplus \delta T_{te} \oplus \delta T_{od} \oplus \delta f_2(R^1) \oplus \delta f_4(R^3) \oplus \ldots \oplus \delta f_{2\lfloor t/2 \rfloor + 2}(R^{2\lfloor t/2 \rfloor + 1}) \\ 0 &= \delta L^0 \oplus \delta T_{to} \oplus \delta T_{ev} \oplus \delta f_1(R^0) \oplus \delta f_3(R^2) \oplus \ldots \oplus \delta f_{2\lfloor (t+1)/2 \rfloor + 1}(R^{2\lfloor (t+1)/2 \rfloor}) \end{aligned}$$

In one of these two equations, both δT_t and δT_{t-1} appear (the one for which T_{te} or T_{to} is nonzero); in the other, only δT_t does. All other terms come out to zero; in one equation, only δT_{j-3} and δT_{j-1} are not guaranteed to be 0, but these are equal to each other. In the other, δT_{j-2} is not guaranteed to be 0, and neither is $f_{j+1}(R^j)$, which appears in the same equation, but again, these two are equal. Thus, from the equation in which T_t appears alone, we conclude $\delta T_t = 0$, and from the other one we then conclude that $\delta T_{t-1} = 0$.

If we assume that $j \geq 4$ then we know that both $\delta T_0 = 0$ and $\delta T_{-1} = 0$. One of those two terms includes *one* of T_{j-3} and T_{j-1}, but not both, and all other tweaks included must remain unchanged. Therefore, both δT_{j-3} and δT_{j-1} are 0. But we know that $0 \neq \delta R^j = \delta T_{j-1}$, so this is a contradiction.

If $j = 3$, then $\delta T_0 = \delta T_2$, but all other even-numbered tweaks are unchanged. Since T_2 is not part of T_0, we note that all the other terms in T_0 are known to have no difference between the two queries. Therefore, $\delta T_0 = 0$ and so $\delta T_2 = 0$, which then implies that $\delta R^3 = 0$, which is a contradiction.

If $j = 2$, then $\delta T_{-1} = \delta T_1$, but all other odd-numbered tweaks are unchanged. Since T_1 is not part of T_{-1}, we can conclude that $\delta T_{-1} = 0$, which is a contradiction.

If $j = 1$, then we conclude that $\delta L^0 = \delta T_0$, via a similar deduction. Because all even-numbered tweaks are 0, we get $\delta T_0 = 0 = \delta L^0$. Since $\delta R^0 = 0$, we know that $\delta R^1 = \delta L^0 \oplus \delta f_1(R^0) = 0$, which is a contradiction.

If $j = 0$, we know that $\delta T = 0$ for all i. Since $0 = \delta R^2 = \delta f_2(R^1) \oplus \delta R^0 \oplus \delta T_{-1}$ and $\delta T_{-1} = \delta R^1 = 0$, we get that $\delta R^0 = 0$, which is a contradiction.

Therefore, if two distinct queries are such that either for all odd $i < t+2$ or for all even $i < t+2$, $\delta R^i = 0$, but there is some $j < t+2$ such that $\delta R^j \neq 0$ then at least two rare events must occur in order for an overall collision to occur: therefore, the probability of a collision in this case is at most $O(2^{-2k})$. This completes the proof of Lemma 17. $\square$

Since in cases i and iii, the probability of a collision is at most $O(2^{-2k})$, and the probability of case ii is 0, the overall probability of a full collision is at most $O(2^{-2k})$. $\square$

Next, we prove that for any pair of distinct queries, the probability of a collision on the right is at most $O(2^{-k})$.

Lemma 18. *On any pair of distinct inputs, the probability that F will produce the same output on the right in each is $O(2^{-k})$.*

Proof. Assume without loss of generality that t is odd; if not, the proof is similar. If t is odd, then whenever two queries lead to a collision on the right, we have

$$\delta L^0 \oplus \delta T_t \oplus \delta T_{ev} \oplus \delta f_1(R^0) \oplus \ldots \oplus \delta f_{t+2}(R^{t+1}) = 0$$

If for some even $i < t+2$ we have $\delta R^i \neq 0$, then the probability of a collision occurring is 2^{-k}.

Let us assume, then, that for all even $i < t+2$, we have $\delta R^i = 0$. If $2 \leq i < t+2$ is even, then $0 = \delta R^i = \delta f_i(R^{i-1}) \oplus \delta L^{i-1} \oplus \delta T_{i-3} = \delta f_i(R^{i-1}) \oplus \delta T_{i-3} \oplus \delta R^{i-2}$, so $\delta f_i(R^{i-1}) = \delta T_{i-3}$, since $\delta R^{i-2} = 0$. Thus, either $\delta R^{i-1} = 0$ or this equation is true with probability 2^{-k}. Let us assume that in all such cases, $\delta R^{i-1} = 0$; if not, the overall probability of a collision is at most $O(2^{-k})$.

If $\delta R^{i-1} = 0$ then $\delta f_i(R^{i-1}) = 0$, so $\delta T_{i-3} = 0$. Thus we know that all the odd-numbered tweaks up to T_{t-2} do not change between the two queries, including the value T_0. But similarly, if $1 \leq i < t+1$ is odd then $\delta T_{i-3} = 0$.

If a collision occurs, we also have that $\delta T_t = \delta R^{t+2} = \delta f_t(R^{t+1}) \oplus \delta R^t \oplus \delta T_{t-1} = \delta T_{t-1}$.

We have been able to deduce that $\delta T = 0$ for $-1 \leq i \leq t-2$. We have included T_t in T_0 and T_{-1} in such a way that regardless of $t \bmod 4$, T_t is involved in one that T_{t-1} is *not* involved in, or vice-versa. Since $\delta T_0 = \delta T_{-1} = 1$, this allows us to conclude that both δT_t and δT_{t-1} are 0. Therefore, at least one "coincidence" must occur in order to produce a collision on the right half of the output, if the two queries are distinct. Therefore, the probability of such a collision is at most $O(2^{-k})$. $\square$

Proof. The proof follows from Lemma 16 and Lemma 18. We can once again apply the principle of deferred decisions, and furthermore, $q \ll 2^k$ queries will not allow a non-negligible probability for the failure of either condition. The reasoning is parallel to that given in the proof of Theorem 4. $\square$

This now allows us to prove two quick theorems:

Theorem 16. *$E \circ F$ is a tweakable blockcipher with t tweaks that is secure against any unbounded adversary with at most $q \ll 2^k$ queries, where E is a four-round Luby-Rackoff cipher.*

Proof. This follows from Theorem 15 and Theorem 5. Note that $E \circ F$ requires a total of $t+6$ rounds.

Theorem 17. *$F' \circ E \circ F$ is a tweakable blockcipher with t tweaks that is CCA-secure against any unbounded adversary with at most $q \ll 2^k$ queries, where E is a four-round Luby-Rackoff cipher, F' is the inverse of the F described above, with new independent round functions.*

Proof. This follows from Theorem 15 and Theorem 12. Here, $F' \circ E \circ F$ requires $2(t+2)+4 = 2t+8$ rounds.

6.4 Minimality of F

Here, we show that the F we have given is minimal in terms of rounds in order to meet the properties required by 5.

Lemma 19. *If F is a Luby-Rackoff based blockcipher incorporating t tweaks, and F has $n < t+2$ rounds, then certain pairs of queries can lead to an overall collision on the output of F with probability $O(2^{-k})$.*

Proof. Without loss of generality, the location for each tweak can be expressed in terms of compound locations based on the locations $\mathcal{L}_1, \ldots, \mathcal{L}_n$. (Again, $\mathcal{L}_0$ and $\mathcal{R}_0$ can be simulated away). Let $\Gamma_1, \ldots, \Gamma_t$ be the compound locations for $T_1, \ldots, T_t$, respectively. Let Γ_i' be defined as the portion of Γ made up of only $\mathcal{L}_3, \ldots, \mathcal{L}_n$, for each i.

Since $n < t+2$, there are fewer than t locations in $\mathcal{L}_3, \ldots, \mathcal{L}_n$. Therefore, there will be some linear dependency among the Γ' values, that is, there will be some i such that for some $S \subset \{1, ...n\}$ such that

$$0 = \sum_{j \in S} \Gamma_j'.$$

Therefore, by Lemma 13, such a construction is insecure; the compound location $\sum_{j \in S} \Gamma_j$ will consist of only locations from $\mathcal{L}_1$ and $\mathcal{L}_2$.

Note that $\mathcal{L}_1$ and $\mathcal{L}_2$ on their own can be thought of as equivalent to $\mathcal{L}_1 + \mathcal{R}_0$ and $\mathcal{L}_2 + \mathcal{L}_0$, respectively. Those constructions fall to the attack of Lemma 6. $\mathcal{L}_1 + \mathcal{L}_2$ can be thought of as equivalent to $\mathcal{L}_0 + \mathcal{L}_1 + \mathcal{L}_2$ which is the same as $\mathcal{R}_{1.5} + \mathcal{R}_2$, which falls to the attack of Corollary 2. $\square$

Security Level	Blockciphers	Prior TBCs [13]	This paper
CPA with polynomial queries	3 rounds [14]	3 + 2 rounds/tweak	3 + 1 round/tweak
CPA with $\ll 2^k$ queries	5 rounds [19]	5 + 2 rounds/tweak	6 + 1 round/tweak
CCA with polynomial queries	4 rounds [14]	4 + 2 rounds/tweak	4 + 2 rounds/tweak
CCA with $\ll 2^k$ queries	5 rounds [19]	5 + 2 rounds/tweak	8 + 2 rounds/tweak

Table 2. Number of rounds required for each construction. The prior tweakable construction we consider is $\widetilde{E_{K,h}}(M,T) = h(T) \oplus E_K(M \oplus h(T))$, where h is an $\epsilon-\mathrm{AXU}_2$ hash function. Subsequent tweakable blockcipher constructions are conceptually similar. The natural way to realize the hash function would be to simply use two random functions on the tweak, one for each half of the data stream. Although Liskov et al. do not explicitly consider arbitrary tweak length, their construction and proof can be easily extended to do so.

7 Conclusion

Table 2 summarizes our constructions, compared to regular blockciphers and the second construction of Liskov et al. [13]. This table shows that our results are better for CPA constructions, equivalent for CCA against polynomial attacks, and worse for CCA against exponential ones.

We have presented a systematic study of issues relating to directly tweaking the large class of Luby-Rackoff blockciphers. Specifically, we have given constructions for both CPA and CCA security, and against both polynomial and exponential attacks, and have considered the problem of incorporating long tweaks. We have proven some of our constructions to be round-optimal in our model.

We conclude with some open problems: (1) incorporating tweaks securely into other blockcipher structures, (2) direct, specific design of tweakable blockciphers (Luby-Rackoff or otherwise) and (3) improving the provable level of security for tweakable blockciphers in general.

Acknowledgments. We thank Ronald L. Rivest and several anonymous reviewers for their helpful comments.

References

1. M. Bellare and T. Kohno. A Theoretical Treatment of Related-Key Attacks: PKA-PRPs, RKA-PRFs, and Applications. In *Advances in Cryptology – EUROCRYPT '03*, volume 2656 of LNCS, pages 491–506, 2003.
2. J. Black, M. Cochran, and T. Shrimpton. On The Impossibility of Highly-Efficient Blockcipher-Based Hash Functions. In *Advances in Cryptology – EUROCRYPT '05*, volume 3494 of *LNCS*, pages 526–541, 2005.
3. C. Burwick, D. Coppersmith, E. D'Avignon, R. Gennaro, S. Halevi, C. Jutla, S. M. Matyas Jr., L. O'Connor, M. Peyravian, D. Safford, and N. Zunic. MARS - A Candidate Cipher for AES. In *NIST AES proposal*, June 1998.
4. P. Crowley. Mercy: A Fast Large Block Cipher for Disk Sector Encryption. In *Fast Software Encryption*, volume 1978 of LNCS, pages 49–63, 2000.
5. Elizabeth Crump. Tweakable Blockciphers Secure Against Generic Exponential Attacks. Masters thesis. College of William and Mary, 2007.
6. Y. Dodis and P. Puniya. Feistel networks made public, and applications. In *EUROCRYPT '07*, volume 4515 of LNCS, pages 534–554, 2007.
7. H. Feistel. Cryptography and Computer Privacy. *Scientific American*, 15–23, 1973.
8. D. Goldenberg, S. Hohenberger, M. Liskov, E. Crump Schwartz, and H. Seyalioglu. Full version of this paper, IACR eprint archive, 2007.
9. S. Halevi EME*: Extending EME to Handle Arbitrary-Length Messages with Associated Data. In *Proceedings of INDOCRYPT 2004*, pp. 315–327, 2004.
10. S. Halevi and P. Rogaway. A Tweakable Enciphering Mode. In *Advances in Cryptology – CRYPTO '03*, volume 2729 of LNCS, pages 482–499, 2003.

11. S. Halevi and P. Rogaway. A Parallelizable Enciphering Mode. In *Topics in Cryptology – CT-RSA '04*, volume 2964 of LNCS, pages 292–304, 2004.
12. A. Joux. Cryptanalysis of the EMD Mode of Operation. In *Advances in Cryptology – EUROCRYPT '03*, volume 2656 of LNCS, pages 1–16, 2003.
13. M. Liskov, R. Rivest, and D. Wagner. Tweakable Block Ciphers. In *Advances in Cryptology – CRYPTO '02*, volume 2442 of LNCS, pages 31–46, 2002.
14. M. Luby and C. Rackoff. How To Construct Pseudorandom Permutations from Pseudorandom Functions. *SIAM J. of Computing*, 17(2):373–386, 1988.
15. S. Lucks. Faster Luby-Rackoff Ciphers. In *Fast Software Encryption*, volume 1039 of LNCS, pages 189–203, 1996.
16. K. Minematsu. Improved Security Analysis of XEX and LRW Modes. In *Selected Areas in Cryptography*, pages 92–109, 2006.
17. M. Naor and O. Reingold. On the Construction of Pseudorandom Permutations: Luby-Rackoff Revisited. *Journal of Cryptology*, 12(1):29–66, 1999.
18. J. Patarin. Luby-Rackoff: 7 Rounds are Enough for $2^{n(1-\varepsilon)}$ Security. In *Advances in Cryptology – CRYPTO*, volume 2729 of LNCS, pages 513–529, 2003.
19. J. Patarin. Security of Random Feistel Schemes with 5 or More Rounds. In *Advances in Cryptology – CRYPTO*, volume 3152 of LNCS, pages 106–122, 2004.
20. Z. Ramzan. *A Study of Luby-Rackoff Ciphers*. PhD thesis, MIT, 2001.
21. R. Rivest, M. Robshaw, R. Sidney, and Y. L. Yin. The RC6 Block Cipher. In *First AES conference*, August 1998.
22. P. Rogaway. Efficient Instantiations of Tweakable Blockciphers and Refinements to Mode OCB and PMAC. In *ASIACRYPT*, vol. 3329, pages 16–31, 2004.
23. R. Schroeppel. The Hasty Pudding Cipher. NIST AES proposal, available `http://www.cs.arizona.edu/~rcs/hpc`, 1998.
24. Hakan Seyalioglu. Exponential Attacks on Blockciper Families. Honors Thesis. College of William and Mary, 2007.
25. D. Wagner. The Boomerang Attack. In *Fast Software Encryption*, vol. 1636 of LNCS, pages 156–170, 1999.