A Domain Extender for the Ideal Cipher

Jean-Sébastien Coron[2], Yevgeniy Dodis[1], Avradip Mandal[2], and Yannick Seurin[3,4]

[1] New York University
[2] University of Luxembourg
[3] University of Versailles
[4] Orange Labs

Abstract. We describe the first domain extender for ideal ciphers, *i.e.* we show a construction that is indifferentiable from a $2n$-bit ideal cipher, given a n-bit ideal cipher. Our construction is based on a 3-round Feistel, and is more efficient than first building a n-bit random oracle from a n-bit ideal cipher (as in [6]) and then a $2n$-bit ideal cipher from a n-bit random oracle (as in [7], using a 6-round Feistel). We also show that 2 rounds are not enough for indifferentiability by exhibiting a simple attack. We also consider our construction in the standard model: we show that 2 rounds are enough to get a $2n$-bit tweakable block-cipher from a n-bit tweakable block-cipher and we show that with 3 rounds we can get beyond the birthday security bound.

Key-words: ideal cipher model, indifferentiability, tweakable block-cipher.

1 Introduction

A block cipher is a primitive that encrypts a n-bit string using a k-bit key. The standard security notion for block-ciphers is to be indistinguishable from a random permutation, for a polynomially bounded adversary, when the key is generated at random in $\{0,1\}^k$. A block-cipher is said to be a strong pseudo-random permutation (or chosen-ciphertext secure) when computational indistinguishability holds even when the adversary has access to the inverse permutation.

When dealing with block-ciphers, it is sometimes useful to work in an idealized model of computation, in which a concrete block-cipher is replaced by a publicly accessible random block-cipher (or ideal cipher); this is a block cipher with a k-bit key and a n-bit input/output, that is chosen uniformly at random among all block ciphers of this form; this is equivalent to having a family of 2^k independent random permutations. All parties including the adversary can make both encryption and decryption queries to the ideal block cipher, for any given key; this is called the Ideal Cipher Model (ICM). Many schemes have been proven secure in the ICM [3, 8, 10, 12, 16, 17, 24]; however, it is possible to construct artificial schemes that are secure in the ICM but insecure for any concrete block cipher (see [2]). Still, a proof in the ideal cipher model seems useful because it shows that a scheme is secure against generic attacks, that do not exploit specific weaknesses of the underlying block cipher.

It was shown in [6, 7] that the Ideal Cipher Model and the Random Oracle Model are equivalent; the random oracle model is similar to the ICM in that a concrete hash function is replaced by a publicly accessible random function (the random oracle). The authors of [6] proved that a random oracle (taking arbitrary long inputs) can be replaced by a block cipher-based construction, and the resulting scheme will remain secure in the ideal cipher model. Conversely, it was shown in [7] that an ideal cipher can be replaced by a 6-round Feistel construction, and the resulting scheme will remain secure in the random oracle model. Both directions were obtained using an extension of the classical notion of indistinguishability, called *indifferentiability*, introduced by Maurer *et al.* in [21].

Since a block cipher can only encrypt a string of fixed length, one must consider the encryption of longer strings. A *mode of operation* of a block-cipher is a method used to extend the domain of applicability from fixed length strings to variable length strings. Many modes of operations have been defined that provide both privacy and authenticity (such as OCB [25]). A mode of operation can also be a permutation; in this case, one obtains an extended block cipher that must satisfy the same property as the underlying block-cipher, *i.e.* it must be a (strong) pseudo-random permutation. Many constructions of domain extender for block-ciphers have been defined that satisfy this security notion, for example PEP [4], XCB [11], HCTR [27], HCH [5] and TET [15].

However, it is easy to see that none of those constructions provide the indifferentiability property that enables to get a $2n$-bit ideal cipher from a n-bit ideal cipher. This is because these constructions were proposed with privacy concerns in mind (mainly for disk encryption purposes) and proven secure only in the classical pseudo-random permutation model. Therefore, these constructions cannot be used when security must hold under the random permutation model (or ideal cipher model). Consider for example the public-key encryption scheme described by Pointcheval and Phan in [24]. The scheme requires a public random permutation with the same size as the RSA modulus, say 1024 bits. In order to replace a 1024-bit random permutation by a construction based on a smaller primitive (for example a 128-bit block cipher), indifferentiability with respect to a 1024-bit random permutation is required. Given a 128-bit block-cipher, none of the previous constructions can provide such property; therefore if one of these constructions is plugged into the Pointcheval and Phan scheme, nothing can be said about the security of the resulting scheme.

In this paper we construct the first domain extender for the ideal cipher; that is we provide a construction of an ideal cipher with $2n$-bit input from an ideal cipher with n-bit input. Given an ideal cipher with n-bit input/output, one could in principle use the construction in [6] to get a random oracle with n-bit output, and then use the 6-round Feistel in [7] to obtain an ideal cipher with $2n$-bit input/output, but that would be too inefficient. Moreover the security bound in [7] is rather loose, which implies that the construction only works for large values of n.[1] In this paper we describe a more efficient construction, based on a 3-round Feistel only, and with a better security bound; we view this as the main result of the paper. More precisely, we show that the 3-round construction in Figure 1 (left) is enough to get a $2n$-bit random permutation from a n-bit ideal cipher, and that its variant in Figure 1 (right) provides a $2n$-bit ideal cipher. We also show that 2 rounds are not enough by providing a simple attack. Interestingly, in the so called honest-but-curious model of indifferentiability [9], we show that 2 rounds are sufficient.

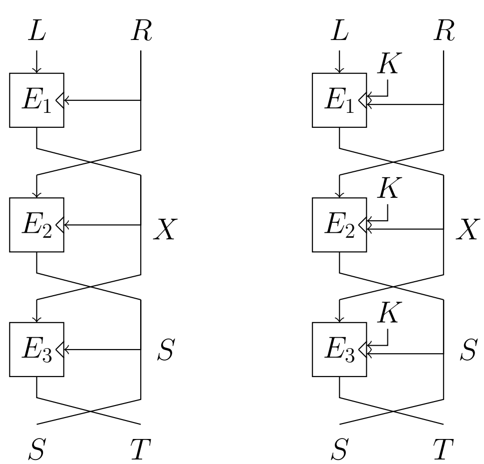

Fig. 1. Construction of a $2n$-bit permutation given a n-bit ideal cipher with n-bit key (left). Construction of a $2n$-bit ideal cipher with k-bit key, given a n-bit ideal cipher with $(n+k)$-bit key (right).

[1] The security bound in [7] for the 6-round Feistel random oracle based construction is $q^{16}/2^n$, where q is the number of distinguisher's queries. This implies that for $q = 2^{64}$, one must take at least $n = 1024$, which corresponds to a 2048-bit permutation.

Finally, we also analyze our construction in the standard model. In this case, we use a *tweakable* block-cipher as the underlying primitive. Tweakable block-ciphers were introduced by Liskov, Rivest and Wagner in [19] and provide an additional input - the tweak - that enables to get a *family* of independent block-ciphers; efficient constructions of tweakable block-ciphers were described in [19], given ordinary block-ciphers. In this paper we show that our construction with 2 rounds enables to get a $2n$-bit tweakable block-cipher from a n-bit tweakable block-cipher. Moreover we show that with 3 rounds we achieve a security guarantee beyond the birthday paradox.

1.1 Related Work

At FSE 2009, Minematsu [22] provided two constructions of a $2n$-bit block-cipher from an n-bit tweakable block-cipher :

1. A 3-round Feistel construction with universal hashing in the 1st round and tweakable block ciphers in the 2nd and the 3rd rounds. This construction is a secure pseudo-random permutation beyond the birthday bound.
2. A 4-round Feistel with universal hashing in the 1st and the 4th rounds and tweakable block ciphers in the 2nd and the 3rd rounds. This construction is a secure strong pseudo-random permutation beyond the birthday bound.

On the other hand, our construction in this paper is a 3-round Feistel, with tweakable block ciphers in every round, and it gives a secure (tweakable) strong pseudo-random permutation beyond the birthday bound. Therefore, the construction in [22] is more efficient as only 2 calls are required to the underlying tweakable block-cipher, instead of 3 calls in our construction (this is assuming very fast universal hashing, *e.g.* [18]). However, we stress that the constructions in [22] are secure only in the symmetric-key setting; it is easy to see that none of the two constructions from [22] can achieve the indifferentiability property (the attack is similar to the attack against 2-round Feistel described in Section 3).

2 Definitions

We first recall the notion of indifferentiability of random systems, introduced by Maurer *et al.* in [21]. This is an extension of the classical notion of indistinguishability, where one or more oracles are publicly available, such as random oracles or ideal ciphers.

As in [21], we define an *ideal primitive* as an algorithmic entity which receives inputs from one of the parties and delivers its output immediately to the querying party. In this paper, we consider ideal primitives such as random oracle, random permutation and ideal cipher. A *random oracle* [1] is an ideal primitive which provides a random output for each new query; identical input queries are given the same answer. A *random permutation* is an ideal primitive that provides oracle access to a random permutation $P : \{0,1\}^n \to \{0,1\}^n$ and to P^{-1}. An *ideal cipher* is a generalization of a random permutation that models a random block cipher $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$. Each key $k \in \{0,1\}^k$ defines an independent random permutation $E_k = E(k,\cdot)$ on $\{0,1\}^n$. The ideal primitive also provides oracle access to E and E^{-1}; that is, on query $(0,k,m)$, the primitive answers $c = E_k(m)$, and on query $(1,k,c)$, the primitive answers m such that $c = E_k(m)$. We stress that in the ideal cipher model, the adversary has oracle access to a publicly available ideal cipher and must send both the key and the plaintext in order to obtain the ciphertext; this is different from the standard model in which the key is privately generated by the system.

The notion of indifferentiability [21] enables to show that an ideal primitive $\mathcal{P}$ (for example, a random permutation) can be replaced by a construction C that is based on some other ideal primitive $\mathcal{E}$; for example, C can be the Feistel construction illustrated in Fig. 1 (left).

Definition 1 ([21]). *A Turing machine C with oracle access to an ideal primitive $\mathcal{E}$ is said to be $(t_D, t_S, q, \varepsilon)$-indifferentiable from an ideal primitive $\mathcal{P}$ if there exists a simulator S with oracle access to $\mathcal{P}$ and running in time at most t_S, such that for any distinguisher D running in time at most t_D and making at most q queries, it holds that:*

$$\left|\Pr\left[D^{C^{\mathcal{E}},\mathcal{E}} = 1\right] - \Pr\left[D^{\mathcal{P},S^{\mathcal{P}}} = 1\right]\right| < \varepsilon$$

$C^{\mathcal{E}}$ is simply said to be indifferentiable from $\mathcal{P}$ if ε is a negligible function of the security parameter n, for polynomially bounded q, t_D and t_S.

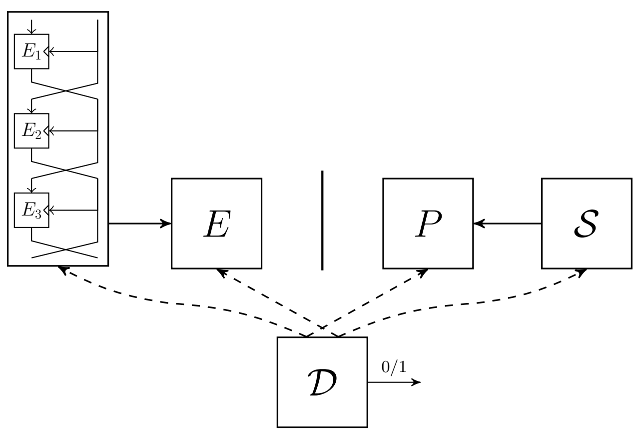

Fig. 2. The indifferentiability notion.

The previous definition is illustrated in Figure 2, where C is our 3-round construction of Figure 1 (left), E is an ideal cipher, $\mathcal{P}$ is a random permutation and $\mathcal{S}$ is the simulator. In this paper, for a 3-round construction, we denote these ideal ciphers by E_1, E_2, E_3 (see Fig. 1). Equivalently, one can consider a single ideal cipher E and encode in the first 2 key bits which round ideal cipher E_1, E_2, or E_3 is actually called. The distinguisher has either access to the system formed by the construction C and the ideal cipher E, or to the system formed by the random permutation P and a simulator $\mathcal{S}$. In the first system (left), the construction C computes its output by making calls to the ideal cipher E (equivalently the 3 ideal ciphers E_1, E_2 and E_3); the distinguisher can also make calls to E directly. In the second system (right), the distinguisher can either query the random permutation P, or the simulator that can make queries to P. If the distinguisher first makes a call to the construction C, and then makes the corresponding calls to ideal cipher E, he will get the same answer. This must remain true when the distinguisher interacts with permutation P and simulator $\mathcal{S}$. The role of simulator $\mathcal{S}$ is then to simulate the ideal ciphers E_i's so that 1) the output of $\mathcal{S}$ should be indistinguishable from that of ideal ciphers E_i's and 2) the output of $\mathcal{S}$ should look "consistent" with what the distinguisher can obtain independently from P. We note that in this model the simulator does not see the distinguisher's queries to P; however, it can call P directly when needed for the simulation.

It is shown in [21] that the indifferentiability notion is the "right" notion for substituting one ideal primitive with a construction based on another ideal primitive. That is, if $C^{\mathcal{E}}$ is indifferentiable from an ideal primitive $\mathcal{P}$, then $C^{\mathcal{E}}$ can replace $\mathcal{P}$ in any cryptosystem, and the resulting cryptosystem is at least as secure in the $\mathcal{E}$ model as in the $\mathcal{P}$ model; see [21] or [6] for a proof.

3 An Attack against 2 Rounds

In this section we show that 2 rounds are not enough when the inner ideal ciphers are publicly accessible, that is we exhibit a property for 2 rounds that does not exist for a random permutation.

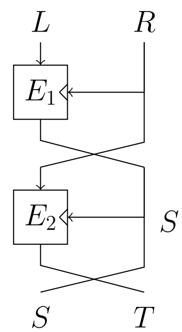

Fig. 3. The 2-round Feistel construction $\Psi_2(L, R)$.

Formally, the 2 round construction is defined as follows (see Fig. 3). Let $E_1 : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher, where $c = E_1(K, m)$ is the n-bit ciphertext corresponding to n-bit key K and n-bit input message m; let E_2 be defined similarly. We define the permutation $\Psi_2 : \{0,1\}^{2n} \to \{0,1\}^{2n}$ as:

$$\Psi_2(L, R) := \big(E_1(R, L), E_2(E_1(R, L), R)\big)$$

It is easy to see that this defines an invertible permutation over $\{0,1\}^{2n}$. Namely, given a ciphertext (S, T) the value R is recovered by "decrypting" T with block-cipher E_2 and key S, and the value L is recovered by "decrypting" S with block-cipher E_1 and key R.

The attack against permutation Ψ_2 is straightforward; it is based on the fact that the attacker can arbitrarily choose both R and S. More precisely, the attacker selects $R = 0^n$ and $S = 0^n$ and queries $L = E_1^{-1}(R, S)$ and $T = E_2(S, R)$. This gives $\Psi_2(L, R) = (S, T)$ as required. However, it is easy to see that with a random permutation P and a polynomially bounded number of queries, it is impossible to find L, R, S, T such that $P(L\|R) = S\|T$ with both $R = 0^n$ and $S = 0^n$, except with negligible probability. Therefore, the 2-round construction cannot replace a random permutation.

Theorem 1. *The 2-round Feistel construction Ψ_2 is not indifferentiable from a random permutation.*

In Appendix A we also analyse existing constructions of domain extender for block ciphers and show that they are not indifferentiable from an ideal cipher; more precisely, we show that the CMC [13] and EME [14] constructions are not indifferentiable from an ideal cipher. We stress that our observations do not imply anything concerning their security in the standard pseudo-random permutation model.

4 Indifferentiability of 3-round Feistel Construction

We now prove our first main result: the 3-round Feistel construction is indifferentiable from a random permutation. To get an ideal cipher, it suffices to prepend a key K to the 3 ideal ciphers E_1, E_2 and E_3; one then gets a family of independent random permutation, parametrised by K, *i.e.* an ideal cipher (see Fig. 1 for an illustration).

Formally, the 3 round permutation $\Psi_3 : \{0,1\}^{2n} \to \{0,1\}^{2n}$ is defined as follows, given block ciphers E_1, E_2 and E_3 with n-bit key (first variable) and n-bit input/output (second variable):

$$X = E_1(R, L)$$
$$S = E_2(X, R)$$
$$T = E_3(S, X)$$
$$\Psi_3(L, R) := (S, T)$$

The 3 round block cipher $\Psi_3' : \{0,1\}^k \times \{0,1\}^{2n} \to \{0,1\}^{2n}$ is defined as follows, given block ciphers E_1, E_2 and E_3 with $(k+n)$-bit key and n-bit input/output:

$$X = E_1(K\|R, L)$$
$$S = E_2(K\|X, R)$$
$$T = E_3(K\|S, X)$$
$$\Psi_3'(K, (L, R)) := (S, T)$$

Theorem 2. *The 3-round Feistel construction Ψ_3 is $(t_D, t_S, q, \varepsilon)$-indifferentiable from a random permutation, with $t_S = \mathcal{O}(qn)$ and $\varepsilon = 5q^2/2^n$. The 3-round block-cipher construction Ψ_3' is $(t_D, t_S, q, \varepsilon)$-indifferentiable from an ideal cipher, with $t_S = \mathcal{O}(qn)$ and $\varepsilon = 5q^2/2^n$.*

Proof. We only consider the 3-round permutation Ψ_3; the extension to block-cipher Ψ_3' is straightforward. We must construct a simulator $\mathcal{S}$ such that the two systems formed by (Ψ_3, E) and $(P, \mathcal{S})$ are indistinguishable (see Fig. 2).

Our simulator maintains an history of already answered queries for E_1, E_2 and E_3. Formally, when the simulator answers X for a $E_1(R, L)$ query, it stores $(1, R, L, X)$ in history; the simulator proceeds similarly for E_2 and E_3 queries. We write that the simulator "simulates" $E_1(R, L) \leftarrow X$ when it first generates a random $X \in \{0,1\}^n \setminus \mathcal{B}$, where $\mathcal{B}$ is the set of already defined values for $E_1(R, \cdot)$, and then stores $(1, R, L, X)$ in history, meaning that $E_1(R, L) = X$; we use similar notations for E_2 and E_3. The distinguisher's queries are answered as follows by the simulator:

$E_1(R, L)$ query:
1. Simulate $E_1(R, L) \leftarrow X$
2. $(S, T) \leftarrow \mathsf{Adapt}(L, R, X)$
3. Return X

$E_1^{-1}(R, X)$ query
1. Simulate $E_1^{-1}(R, X) \leftarrow L$
2. $(S, T) \leftarrow \mathsf{Adapt}(L, R, X)$
3. Return L

$E_2(X, R)$ query:
1. Simulate $E_1^{-1}(R, X) \leftarrow L$
2. $(S, T) \leftarrow \mathsf{Adapt}(L, R, X)$
3. Return S

$\mathsf{Adapt}(L, R, X)$:
1. $S\|T \leftarrow P(L\|R)$
2. Store $(2, X, R, S)$ in history $(E_2(X, R) = S)$
3. Store $(3, S, X, T)$ in history $(E_3(S, X) = T)$
4. Return (S, T).

The procedure for answering the other queries is essentially symmetric; we provide it for completeness:

$E_3^{-1}(S, T)$ query:
1. Simulate $E_3^{-1}(S, T) \leftarrow X$
2. $(L, R) \leftarrow \mathsf{Adapt}^{-1}(S, T, X)$
3. Return X

$E_3(S, X)$ query
1. Simulate $E_3(S, X) \leftarrow T$
2. $(L, R) \leftarrow \mathsf{Adapt}^{-1}(S, T, X)$
3. Return T

$E_2^{-1}(X, S)$ query:
1. Simulate $E_3(S, X) \leftarrow T$
2. $(L, R) \leftarrow \mathsf{Adapt}^{-1}(S, T, X)$
3. Return R

$\mathsf{Adapt}^{-1}(S, T, X)$:
1. $L\|R \leftarrow P^{-1}(S\|T)$
2. Store $(2, X, R, S)$ in history $(E_2(X, R) = S)$

3. Store $(1, R, L, X)$ in history ($E_1(R, L) = X$)
4. Return (L, R)

Finally, the simulator aborts if for some E_i and some key K, it has not defined a permutation for $E_i(K, \cdot)$; that is the simulator aborts if it has defined $E_i(K, X) = E_i(K, Y)$ for some $X \neq Y$ or it has defined $E_i^{-1}(K, X) = E_i^{-1}(K, Y)$ for some $X \neq Y$. This completes the description of the simulator.

As a consistency check, it is easy to see that if the distinguisher makes a single query for $P(L\|R)$ and then queries the simulator for $X \leftarrow E_1(R, L)$, $S \leftarrow E_2(X, R)$ and $T \leftarrow E_3(S, X)$, then the distinguisher obtains $S\|T = P(L\|R)$ as required.

We now proceed to prove that the systems (Ψ_3, E) and $(P, \mathcal{S})$ are indistinguishable. We consider a distinguisher $\mathcal{D}$ making at most q queries to the system (Ψ_3, E) or $(P, \mathcal{S})$ and outputting a bit γ. We define a sequence $\mathtt{Game}_0$, $\mathtt{Game}_1$, ... of modified distinguisher games. In the first game the distinguisher interacts with the system (Ψ_3, E). We incrementally modify the system so that in the last game the distinguisher interacts with the system $(P, \mathcal{S})$, where $\mathcal{S}$ is the previously defined simulator. We denote by S_i the event that in game i the distinguisher outputs $\gamma = 1$.

- $\mathtt{Game}_0$: the distinguisher interacts with Ψ_3 and the ideal ciphers E_i.

- $\mathtt{Game}_1$: we modify the way E_i queries are answered, without actually changing the value of the answer. We also maintain an history of already answered queries for E_1, E_2 and E_3. We proceed as follows:

$E_1(R, L)$ query:	$E_1^{-1}(R, X)$ query	$E_2(X, R)$ query:
1. Let $X \leftarrow E_1(R, L)$	1. Let $L \leftarrow E_1^{-1}(R, X)$	1. Let $L \leftarrow E_1^{-1}(R, X)$
2. $(S, T) \leftarrow \mathsf{Adapt}'(L, R, X)$	2. $(S, T) \leftarrow \mathsf{Adapt}'(L, R, X)$	2. $(S, T) \leftarrow \mathsf{Adapt}'(L, R, X)$
3. Return X	3. Return L	3. Return S

$\mathsf{Adapt}'(L, R, X)$:
1. $S\|T \leftarrow \Psi_3(L\|R)$
2. Store $(2, X, R, S)$ in history ($E_2(X, R) = S$)
3. Store $(3, S, X, T)$ is history ($E_3(S, X) = T$).
4. Return (S, T)

The queries to $E_2^{-1}(X, S)$, $E_3(S, X)$ and $E_3^{-1}(S, T)$ are answered symmetrically.

For example, when given a query to $E_1(R, L)$, we first query ideal cipher E_1 for $X \leftarrow E_1(R, L)$; then instead of X being returned immediately as in $\mathtt{Game}_0$, we let $S\|T = \Psi_3(L\|R)$, which gives $S = E_2(X, R)$ and $E_3(S, X) = T$; we then store $(2, X, R, S)$ and $(3, S, X, T)$ in history. Therefore, the value that get stored in history is exactly the same as the value from ideal ciphers E_2 and E_3; the only difference is that this value was obtained indirectly by querying Ψ_3 instead of directly by querying E_2 and E_3. It is easy to see that this holds for any query made by the distinguisher, who receives exactly the same answers in $\mathtt{Game}_0$ and $\mathtt{Game}_1$; this implies:

$$\Pr[S_1] = \Pr[S_0]$$

As illustrated in Fig. 4, we have actually constructed a simple simulator $\mathcal{S}'$ that makes queries to a subsystem $\mathcal{T}$ that comprises the construction Ψ_3 and the ideal ciphers E_1, E_2 and E_3. The difference between $\mathcal{S}'$ in $\mathtt{Game}_1$ and the main simulator $\mathcal{S}$ defined previously is that 1) $\mathcal{S}'$ calls ideal cipher $E_1(R, L)$ instead of simulating it and 2) $\mathcal{S}'$ makes calls to $\Psi_3(L\|R)$ instead of $P(L\|R)$.

- $\mathtt{Game}_2$: we modify the way the permutation queries are answered. Instead of using Ψ_3 as in system $\mathcal{T}$, we use the random permutation P in the new system $\mathcal{T}'$ (see Fig. 4).

We must show that the distinguisher's view has statistically close distribution in $\mathtt{Game}_1$ and $\mathtt{Game}_2$. For this, we consider the subsystem $\mathcal{T}$ with the 3-round Feistel Ψ_3 and the ideal ciphers E_i's in $\mathtt{Game}_1$, and the subsystem $\mathcal{T}'$ with the random permutation P and ideal ciphers E_i's in $\mathtt{Game}_2$. We show that

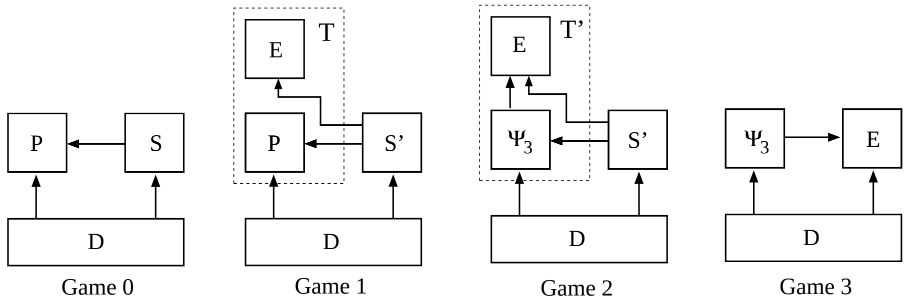

Fig. 4. Sequence of games for proving indifferentiability.

the output of systems $\mathcal{T}$ and $\mathcal{T}'$ is statistically close; this in turn shows that the distinguisher's view has statistically close distribution in $\mathtt{Game}_1$ and $\mathtt{Game}_2$. Note that the indistinguishability of $\mathcal{T}$ and $\mathcal{T}'$ only holds for the particular set of queries made by the distinguisher and the simulator; it could not hold for any possible set of queries.

In the following, we assume that the distinguisher eventually makes a sequence of E_i queries corresponding to all previous Ψ_3 queries that he has made. More precisely, if the distinguisher has made a $\Psi_3(L,R)$ query, then eventually the distinguisher makes the sequence of queries $X \leftarrow E_1(R,L)$, $S \leftarrow E_2(X,R)$ and $T \leftarrow E_3(S,X)$ to the simulator; the same holds for $\Psi_3^{-1}(S,T)$ queries. This is without loss of generality, because from any distinguisher $\mathcal{D}$ we can build a distinguisher $\mathcal{D}'$ with the same output that satisfies this property.

The outputs to E_i queries provided by subsystem $\mathcal{T}$ in $\mathtt{Game}_1$ and by subsystem $\mathcal{T}'$ in $\mathtt{Game}_2$ are the same, since in both cases these queries are answered by ideal ciphers E_i. Therefore, we must show that the output to P/P^{-1} queries provided by $\mathcal{T}$ and $\mathcal{T}'$ have statistically close distribution, when the outputs to E_i queries provided by $\mathcal{T}$ or $\mathcal{T}'$ are fixed.

We consider a forward permutation query $L\|R$ made by either the distinguisher or the simulator $\mathcal{S}'$. If this $L\|R$ query is made by the distinguisher, since we have assumed that the distinguisher eventually makes the E_i queries corresponding to all his permutation queries, this $L\|R$ query will also be made by the simulator $\mathcal{S}'$, by definition of $\mathcal{S}'$. Therefore we can consider $L\|R$ queries made by the simulator $\mathcal{S}'$ only.

We first consider the answer to $S\|T = \Psi_3(L\|R)$ in $\mathtt{Game}_1$. In this case the answer $S\|T$ is computed as follows:

$$\begin{aligned} X &= E_1(R,L) \\ S &= E_2(X,R) \\ T &= E_3(S,X) \end{aligned}$$

By definition of the simulator $\mathcal{S}'$, when the simulator $\mathcal{S}'$ makes a query for $\Psi_3(L\|R)$, it must have made an ideal cipher query to $E_1(R,L)$ before, or an ideal cipher query to $E_1^{-1}(R,X)$ before, with $L = E_1^{-1}(R,X)$.

If the simulator $\mathcal{S}'$ has made an ideal cipher query for $E_1(R,L)$ to subsystem $\mathcal{T}$, then from the definition of the simulator a call to $\mathsf{Adapt}'(L,R,X)$ has occurred, where $X = E_1(R,L)$; in this Adapt' call the values $E_2(X,R)$ and $E_3(S,T)$ are defined by the simulator; therefore the simulator does not make these queries to sub-system $\mathcal{T}$. This implies that the values of $E_2(X,R)$ and $E_3(S,X)$ are not included in the subsystem $\mathcal{T}$ output; therefore these values are not fixed in the probability distribution that we consider; only the value $X = E_1(R,L)$ is fixed.

Moreover, for fixed X, R the distribution of $S = E_2(X,R)$ is uniform in $\{0,1\}^n \setminus \mathcal{B}$, where $\mathcal{B}$ is the set of already defines values for $E_2(X,\cdot)$. Since there are at most q queries, the statistical distance between the distribution of $E_2(X,R)$ and the uniform distribution in $\{0,1\}^n$ is at most $2q/2^n$; the same holds for the distribution of $T = E_3(S,X)$. Therefore, we obtain that for a fixed X, the distribution of (S,T) is statistically close to the uniform distribution in $\{0,1\}^{2n}$, with statistical distance at most $4q/2^n$.

If the simulator has made an ideal cipher query for $E_1^{-1}(R,X)$, then the same analysis applies and we obtain that for a fixed $L = E_1^{-1}(R,X)$ the distribution of (S,T) is statistically close to the uniform distribution in $\{0,1\}^{2n}$, with statistical distance at most $4q/2^n$. Therefore we obtain that in $\mathtt{Game}_1$ the statistical distance of $S\|T = \Psi_3(L\|R)$ with the uniform distribution is always at most $4q/2^n$.

In $\mathtt{Game}_2$, the output to permutation query $L\|R$ is $S\|T = P(L\|R)$; since there are at most q queries to P/P^{-1}, the statistical distance between $P(L\|R)$ and the uniform distribution in $\{0,1\}^{2n}$ is at most $2q/2^{2n}$.

Therefore the statistical distance between $\Psi_3(L,R)$ in $\mathtt{Game}_1$ and $P(L\|R)$ in $\mathtt{Game}_2$ is at most $4q/2^n + 2q/2^{2n} \le 5q/2^n$. The same argument applies to inverse permutation queries. This holds for a single permutation query; since there are at most q such queries, we obtain that the statistical distance between outputs of systems $\mathcal{T}$ and $\mathcal{T}'$ to permutation queries and E_i queries, is at most $5q^2/2^n$; this implies:

$$|\Pr[S_2] - \Pr[S_1]| \le \frac{5q^2}{2^n}$$

• $\mathtt{Game}_3$: eventually the distinguisher interacts with system $(P, \mathcal{S})$. The only difference between the simulator $\mathcal{S}'$ in $\mathtt{Game}_2$ and the simulator $\mathcal{S}$ in $\mathtt{Game}_3$ is that instead of querying ideal ciphers E_i in $\mathtt{Game}_2$, these ideal ciphers are simply simulated in $\mathtt{Game}_3$, while the answer to permutation queries are exactly the same. Therefore, the distinguisher's view has the same distribution in $\mathtt{Game}_2$ and $\mathtt{Game}_3$, which gives:

$$\Pr[S_2] = \Pr[S_3]$$

and finally:

$$|\Pr[S_3] - \Pr[S_0]| \le \frac{5q^2}{2^n}$$

which terminates the proof of Theorem 2. □

We note that the security bound in $q^2/2^n$ for our 3-round ideal cipher based construction is much better than the security bound in $q^{16}/2^n$ obtained for the 6-round Feistel construction in [7] (based on random oracles).

4.1 Practical Considerations

Extending the Key. So far, we showed how to construct an ideal cipher Ψ_3 with $2n$-bit message and k-bit key from three ideal ciphers E_1, E_2, E_3 on n-bit message and $(n+k)$-bit key. As already mentioned, we can actually implement E_1, E_2, E_3 from a single n-bit ideal cipher E whose key length is $n+k+2$.

However, if only a block-cipher with n-bit key and n-bit message is available (for example AES-128), we need a procedure to extend the key size. To handle such cases, we notice that it suffices to first hash the key using a random oracle, and the resulting block cipher remains indifferentiable from an ideal cipher.

Lemma 1. *Assume $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ is an ideal cipher and $H : \{0,1\}^t \to \{0,1\}^k$ is a random oracle. Define $E' : \{0,1\}^t \times \{0,1\}^n \to \{0,1\}^n$ by $E'(K',X) = E(H(K'),X)$, $E'^{-1}(K',Y) = E^{-1}(H(K'),Y)$. Then E' is $(t_D, t_S, q, \varepsilon)$-indifferentiable from an ideal cipher, where $t_S = \mathcal{O}(q(n+t))$ and $\varepsilon = \mathcal{O}(q^2/2^k)$.*

Proof. See Appendix B.

Using this observation, given a single ideal cipher E on n-bit messages and k-bit key and a random oracle H with output size k bits, we can first build an ideal cipher E' with n-bit message and $(n+k'+2)$-bit key, and then from Theorem 2 we can obtain an ideal cipher Ψ_3 on $2n$-bit messages and k'-bit key. It remains to remove the assumption of having random oracle H; this can easily be accomplished by sacrificing 1 key bit from E, and then using one of the two resulting (independent) ideal ciphers to efficiently implement H using any of the methods from [6].

Going Beyond Double? Another natural question is to extend the domain of the ideal cipher beyond doubling it. One way to accomplish this task is to apply our 3-round construction recursively, each time doubling the domain. However, in this case it is not hard to see that, to extend the domain by a factor of t, the original block cipher E will have to be used $\mathcal{O}(t^{\log_2 3})$ times.[2] This makes the resulting constructions somewhat impractical for large t.

In contrast, assume that we use the 2-step construction: first build a length-preserving random oracle H on $nt/2$ bits (using [6]), and then use the 6-round Feistel construction [7] to get a nt-bit permutation. To construct a random oracle from $nt/2$-bit to $nt/2$-bit, only $\mathcal{O}(t)$ calls to the n-bit ideal cipher are required (first hash from $nt/2$-bit to n-bit using [6], then expand back to $nt/2$-bits using counter mode). Therefore the 2-step construction requires only $\mathcal{O}(t)$ calls to E, instead of $\mathcal{O}(t^{\log_2 3})$ when iterating our construction. This implies that for large t, the 2-step construction is more efficient.

To give a practical example, let us consider the applications of [12, 24], where one needs to apply a random permutation to the domain of an RSA modulus. We take the length of modulus N to be 1024 bits and the underlying block-cipher E to be $n = 128$ with 128-bit key (as in AES-128). One can see that to obtain a 1024-bit permutation from E, only 48 calls to E are required for the 2-step construction, instead of 243 when iterating our construction. However for 1024-bit, the exact security of the 2-step construction is dominated by the term $\mathcal{O}(q^{16}/2^{512})$ from [7], which requires $q \ll 2^{32}$, whereas the exact security of the recursive construction is $\mathcal{O}(q^2/2^{128})$, which requires $q \ll 2^{64}$. Therefore, for a 1024-bit permutation our recursive construction still provides a better security bound; however, for any size larger than 2048 bits, the two constructions have the same $q \ll 2^{64}$ bound [3].

To summarize, our construction is more efficient than the 2-step construction when doubling only once ($t = 2$). However for a large expansion factor t the 2-step construction is more efficient than the recursive method.

4.2 Indifferentiability for 2 Rounds in the Honest-but-curious Model

In this section we also consider the *honest-but-curious* model of indifferentiability introduced by Dodis and Puniya [9], which is a variant of the general indifferentiability model. We show that in the honest-but-curious model, 2 rounds as depicted in Fig 3 are actually sufficient to get indifferentiability.

First, we briefly recall the model; for more details we refer to [9]. In the honest-but-curious model of indifferentiability, the distinguisher cannot make direct queries to the inner primitive E. Instead it can only query the global construction C and get the results of the internal queries made by the construction to the inner primitive E. There are actually two types of queries made by the distinguisher: those for which it asks for the transcript of the queries made by the construction to the primitive E, and those for which it does not. When the distinguisher interacts with $(\mathcal{P}, \mathcal{S}^{\mathcal{P}})$, the second queries are sent directly to $\mathcal{P}$ (and are not seen by the simulator), while the first ones are sent to the simulator

[2] In essence, this is because we call E three times for each doubling. Actually, this is not counting the calls to the independent variable length random oracle H to hash down the key, as above. However, because the constructions of such an H in [6] are so efficient, it is not hard to see that, even when implementing H using E itself, the dominant term remains $\mathcal{O}(t^{\log_2 3})$ (although the constant is slightly worse).

[3] The length-preserving random oracle used in the 6-round Feistel has the birthday bound of $q^2/2^{128}$

$\mathcal{S}$, which must simulate the transcript of the construction's inner queries to E. Another important difference with general indifferentiability is that here the simulator cannot make its own additional queries to $\mathcal{P}$.

Theorem 3. *The 2-round construction is (t_D, t_S, q, ϵ)-indifferentiable in the honest-but-curious model from a random permutation, with $t_S = \mathcal{O}(qn)$ and $\epsilon = 2q^2/2^n$, where q is the total number of distinguisher queries and n is the domain size of the inner ciphers.*

Proof. The proof is given in Appendix C.

Remark 1. Indifferentiability in the honest-but-curious model has been shown to imply indifferentiability in the general model for so-called transparent constructions [9]. A construction is said to be transparent if there exists an efficient algorithm which can compute the value of the inner primitive E on any input x by making a polynomial number of queries to the construction and receiving the transcript of the inner queries of the construction to E. Since the 2-round construction is not indifferentiable in the general model, this shows that it is also not transparent: namely it is impossible to efficiently compute $E_2(S, R)$ for some arbitrary value S, or $E_1^{-1}(R, S)$ for some arbitrary value R, given only oracle access + transcript to $\Psi_2(L, R)$ and $\Psi_2^{-1}(S, T)$.

5 Domain Extension of Tweakable Block Cipher

In this section, we also analyse our construction in the standard model, and we use a *tweakable* block-cipher as the underlying primitive. The main result of this section is that a 3-round Feistel enables to get a security guarantee beyond the birthday paradox.

Tweakable block-ciphers were introduced by Liskov, Rivest and Wagner in [19] and provide an additional input - the tweak - that enables to get a *family* of independent block-ciphers. Efficient constructions of tweakable block-ciphers were described in [19], given ordinary block-ciphers.

Definition 2. *A tweakable block-cipher is an efficiently computable function $\tilde{E} : \{0,1\}^k \times \{0,1\}^\omega \times \{0,1\}^n \to \{0,1\}^n$ that takes as input a key $K \in \{0,1\}^k$, a tweak $W \in \{0,1\}^\omega$ and a message $m \in \{0,1\}^n$ and returns a ciphertext $c \in \{0,1\}^n$. For every $K \in \{0,1\}^k$ and $W \in \{0,1\}^\omega$, the function $\tilde{E}(K, W, \cdot)$ is a permutation over $\{0,1\}^n$.*

The security notion for a tweakable block-cipher is a straightforward extension of the corresponding notion for block-ciphers. A classical block-cipher E is a strong pseudo-random permutation if no adversary can distinguish $E(K, \cdot)$ from a random permutation, where $\mathcal{A}$ can make calls to both E and E^{-1}, and $K \leftarrow \{0,1\}^k$. For tweakable block-ciphers, the adversary can additionally choose the tweak, and $E(K, \cdot, \cdot)$ should be indistinguishable from a *family* of random permutations, parametrised by $W \in \{0,1\}^\omega$:

Definition 3. *A tweakable block-cipher is said to be (t, q, ε)-secure if for any adversary $\mathcal{A}$ running in time at most t and making at most q queries, the adversary's advantage in distinguishing $\tilde{E}(K, \cdot, \cdot)$ with $K \leftarrow \{0,1\}^k$ from a family of independent random permutation $\tilde{\Pi}(\cdot, \cdot)$ is at most ε, where $\mathcal{A}$ can make calls to both $\tilde{E}$ and $\tilde{E}^{-1}$.*

We first show that 2 rounds are enough to get a $2n$-bit tweakable block-cipher from a n-bit tweakable block-cipher (see Fig. 5, left). Formally, our 2-round domain extender for tweakable block-cipher works as follows. Let E_1 and E_2 be two tweakable block-ciphers with the same signature:

$$\tilde{E}_i : \{0,1\}^k \times \{0,1\}^\omega \times \{0,1\}^n \to \{0,1\}^n$$

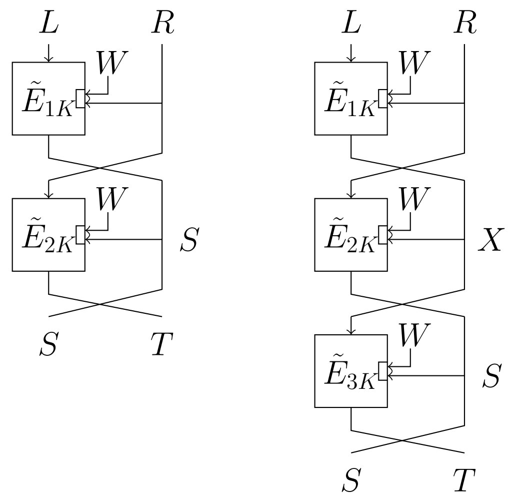

Fig. 5. The tweakable block ciphers $\tilde{\Psi}_2$ (left) and $\tilde{\Psi}_3$ (right), with key K and tweak W

The tweakable block cipher $\tilde{\Psi}_2 : \{0,1\}^k \times \{0,1\}^{\omega-n} \times \{0,1\}^{2n} \to \{0,1\}^{2n}$ is then defined as follows; the difference with Fig. 3 is that the R and S inputs go to the tweak (concatenated with the main tweak W) instead of the key.

$$S = E_1(K, W\|R, L)$$
$$T = E_2(K, W\|S, R)$$
$$\tilde{\Psi}_2(K, W, (L,R)) = (S,T)$$

Theorem 4. *The tweakable block-cipher $\tilde{\Psi}_2$ is a (t', q, ε')-secure tweakable block-cipher, if $\tilde{E}_1$ and $\tilde{E}_2$ are both (t, q, ε)-secure tweakable block-ciphers, where $\varepsilon' = 2 \cdot \varepsilon + q^2/2^n + q^2/2^{2n}$ and $t' = t - \mathcal{O}(qn)$.*

Proof. See Appendix D.

Now we consider the 3 round tweakable block cipher $\tilde{\Psi}_3$, defined in a similar manner as $\tilde{\Psi}_2$ (see Fig. 5 for an illustration). The 3-round construction enables to go beyond the birthday security bound. Namely instead of having a bound in $q^2/2^n$ as in the 2-round construction, the bound for the 3-round construction is now $q^2/2^{2n}$, which shows that the construction remains secure until $q < 2^n$ instead of $q < 2^{n/2}$.

Theorem 5. *The tweakable block-cipher $\tilde{\Psi}_3$ is a (t', q, ε')-secure tweakable block-cipher, if $\tilde{E}_1$, $\tilde{E}_2$ and $\tilde{E}_3$ are all (t, q, ε)-secure tweakable block-ciphers, where $\varepsilon' = 3 \cdot \varepsilon + q^2/2^{2n}$ and $t' = t - \mathcal{O}(qn)$.*

Proof. See Appendix E.

One drawback of our construction is that it shrinks the tweak size from ω bits to $\omega - n$ bits. We show a simple construction that extends the tweak size, using a keyed universal hash function; this construction can be of independent interest.

Definition 4. *A family $\mathcal{H}$ of functions with signature $\{0,1\}^{\omega'} \to \{0,1\}^{\omega}$ is said to be ε-almost universal if $\Pr_h[h(x) = h(y)] \le \varepsilon$ for all $x \ne y$, where the probability is taken over h chosen uniformly at random from $\mathcal{H}$.*

Let $\tilde{E}$ be a tweakable block-cipher with tweak in $\{0,1\}^{\omega}$. Given a family $\mathcal{H}$ of hash functions h with signature $\{0,1\}^{\omega'} \to \{0,1\}^{\omega}$ and $\omega' > \omega$, our tweakable block-cipher $\tilde{E}$ with extended tweak length ω' is defined as:

$$\tilde{E}'((K,h), W', m) = \tilde{E}(K, h(W'), m)$$

Theorem 6. *The tweakable block cipher $\tilde{E}'$ is a (q, t', ε')-secure tweakable block cipher if $\tilde{E}$ is a (q, t, ε_1)-secure tweakable block cipher and the hash function family $\mathcal{H}$ is ε_2-almost universal, with $\varepsilon' = \varepsilon_1 + q^2 \cdot \varepsilon_2$ and $t' = t - \mathcal{O}(q)$.*

Proof. See Appendix F.

We note that many efficient constructions of universal hash function families are known, with $\varepsilon_2 \simeq 2^{-\omega}$. Therefore the new tweakable block-cipher can have the same level of security as the original one, up to the birthday bound for the tweak, *i.e.* for $q \leq 2^{\omega/2}$.

6 Conclusion

We have described the first domain extender for ideal ciphers, *i.e.* we have showed a construction that is indifferentiable from a $2n$-bit ideal cipher, given a n-bit ideal cipher. Our construction is based on a 3-round Feistel, and is more efficient and more secure than first building a n-bit random oracle from a n-bit ideal cipher (as in [6]) and then a $2n$-bit ideal cipher from a n-bit random oracle (as in [7]). We have also shown that in the standard model, our construction with 2 rounds enables to get a $2n$-bit tweakable block-cipher from a n-bit tweakable block-cipher and that with 3 rounds we get a security guarantee beyond the birthday paradox.

References

1. M. Bellare and P. Rogaway, *Random oracles are practical: A paradigm for designing efficient protocols*, In Proceedings of the 1st ACM Conference on Computer and Communications Security (1993), 62 -73.
2. J. Black, *The Ideal-Cipher Model, Revisited: An Uninstantiable Blockcipher-Based Hash Function*, Proceedings of FSE 2006: 328-340.
3. J. Black, P. Rogaway, T. Shrimpton, *Black-Box Analysis of the Block Cipher-Based Hash-Function Constructions from PGV*, in Advances in Cryptology - CRYPTO 2002, California, USA.
4. D. Chakraborty and P. Sarkar. A new mode of encryption providing a tweakable strong pseudo-random permutation. In *Proceedings of FSE '06*, LNCS 4047, pp. 293–309, 2006.
5. D. Chakraborty and P. Sarkar. HCH: A new tweakable enciphering scheme using the hash-encrypt-hash approach. In *Proceedings of Indocrypt '06*, LNCS 4329, pp. 287–302, 2006.
6. J.S. Coron, Y. Dodis, C. Malinaud and P. Puniya, *Merkle-Damgård Revisited: How to Construct a Hash Function*. Proceedings of CRYPTO 2005: 430-448.
7. J.S. Coron, J. Patarin and Y. Seurin, *The Random Oracle Model and the Ideal Cipher Model are Equivalent*. Proceedings of CRYPTO 2008. Full version available at Cryptology ePrint Archive, Report 2008/246, http://eprint.iacr.org/.
8. A. Desai, *The security of all-or-nothing encryption: Protecting against exhaustive key search*, In Advances in Cryptology - Crypto' 00 (2000), LNCS vol. 1880, Springer-Verlag.
9. Y. Dodis and P. Puniya, *On the Relation Between the Ideal Cipher and the Random Oracle Models*. Proceedings of TCC 2006: 184-206.
10. S. Even and Y. Mansour, *A construction of a cipher from a single pseudorandom permutation*, In Advances in Cryptology - ASIACRYPT' 91 (1992), LNCS vol. 739, Springer-Verlag, pp. 210 -224.
11. S.R. Fluhrer and D.A. McGrew. The extended codebook (XCB) mode of operation. Technical Report 2004/078, IACR eprint archive, 2004.
12. L. Granboulan, *Short signature in the random oracle model*. Proceedings of Asiacrypt 2002, LNCS 2501.
13. S. Halevi and P. Rogaway. A tweakable enciphering mode. In D. Boneh, editor, *Advances in Cryptology, CRYPTO '03*, 2007.
14. S. Halevi and P. Rogaway. A parallelizable enciphering mode. In *Proceedings of CT-RSA 2004*, LNCS 2964, pp. 292–304, 2004.
15. S. Halevi. Invertible Universal hashing and the TET Encryption Mode. In *Proceedings of CRYPTO '07*, LNCS 4622, pp. 412–429, 2007.
16. J. Jonsson, *An OAEP variant with a tight security proof*, available at `http://eprint.iacr.org/2002/034/`.
17. J. Kilian and P. Rogaway, *How to protect DES against exhaustive key search (An analysis of DESX)*, Journal of Cryptology 14, 1 (2001), 17 -35.
18. T. Krovetz, *Message Authentication on 64-Bit Architectures*. In: Biham, E., Youssef, A.M. (eds.) SAC 2006. LNCS, vol. 4356. Springer, 2007.

19. M. Liskov, R. Rivest and D. Wagner, *Tweakable Block Ciphers*. Proceedings of CRYPTO 2002, LNCS vol. 2442.
20. M. Luby and C. Rackoff, *How to construct pseudorandom permutations from pseudorandom functions*, SIAM Journal of Computing, 17(2):373-386, 1988.
21. U. Maurer, R. Renner, and C. Holenstein, *Indifferentiability, Impossibility Results on Reductions, and Applications to the Random Oracle Methodology*. Theory of Cryptography - TCC 2004, Lecture Notes in Computer Science, Springer-Verlag, vol. 2951, pp. 21-39, Feb 2004.
22. K. Minematsu, *Beyond-Birthday-Bound Security Based on Tweakable Block Cipher*. Proceedings of FSE 2009. Springer.
23. M. Naor and O. Reingold, *On the construction of pseudorandom permutations: Luby-Rackoff revisited*, J. of Cryptology, 1999. Preliminary Version: STOC 1997.
24. D. H. Phan and D. Pointcheval. *Chosen-Ciphertext Security without Redundancy*. Proceedings of Asiacrypt '03, LNCS 2894.
25. P. Rogaway, M. Bellare and J. Black, *OCB: A block-cipher mode of operation for efficient authenticated encryption*. ACM Conference on Computer and Communication Security 2001: 196-205.
26. V. Shoup, *Sequences of games: a tool for taming complexity in security proofs*. Available electronically at `http://eprint.iacr.org/2004/332/`.
27. P. Wang, D. Feng, and W. Wu. HCTR: A variable-input-length enciphering mode. In *Proceedings of CISC '05*, LNCS 3822, pp. 175–188, 2005.

A Previous Constructions are not Indifferentiable

We analyse previous constructions of domain extender for block ciphers and show that they are not indifferentiable from an ideal cipher. This is not surprising as all these constructions were proposed with privacy concerns in mind (mainly for disk encryption purposes) and proven secure in the classical Luby-Rackoff model. Most of this constructions use two layers of keyed universal hashing and cannot be analysed in the indifferentiability framework: this is the case for example of PEP [4], XCB [11], HCTR [27], HCH [5] and TET [15].

Other constructions however use nothing more than the underlying block cipher. The two most prominent of them are CMC [13] and EME [14] proposed by Halevi and Rogaway. We now show that these two constructions are not indifferentiable from an ideal cipher.

A.1 The CMC construction

CMC was proposed by Halevi and Rogaway [13] and uses two layers of CBC and an intermediate mixing layer. This is a tweakable mode but we don't consider the tweak in our description (that is we set the tweak to $T = 0^n$) since it is not relevant for our attack.

CMC uses a block cipher $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ and turns it into a tweakable block cipher $\mathbf{E}$ with tweak space $\{0,1\}^n$, key space $\{0,1\}^k \times \{0,1\}^k$, and message space $\bigcup_{m\geq 2}\{0,1\}^{mn}$. A message $P_1 \cdots P_m$ of m n-bit blocks is encrypted under key K, K' and tweak T as follows:

1. $\mathbb{T} \leftarrow E_{K'}(T)$
2. $PPP_0 \leftarrow \mathbb{T}$
3. for $i = 1$ to m do $PPP_i \leftarrow E_K(P_i \oplus PPP_{i-1})$
4. $M \leftarrow 2(PPP_1 \oplus PPP_m)$
5. for $i = 1$ to m do $CCC_i \leftarrow PPP_{m+i-1} \oplus M$
6. $CCC_0 \leftarrow 0^n$
7. for $i = 1$ to m do $C_i = E_K(CCC_i) \oplus CCC_{i-1}$
8. $C_1 \leftarrow C_1 \oplus \mathbb{T}$
9. return $C_1 \cdots C_m$

The attack on CMC proceeds as follows (we describe the attack for two blocks only, it can be easily extended to any number of blocks).

If first fixes two arbitrary keys K' and K, and computes $\mathbb{T} = E_{K'}(T)$. It then simply consists in computing $P_1 = E_K^{-1}(0^n)$. One can then verify that the encryption of $(P_1 \oplus \mathbb{T})|P_1$ is $\mathbf{E}_{K,K'}((P_1 \oplus$

$\mathbb{T})\|P_1) = E_K(0)\|(E_K(0)\oplus\mathbb{T})$. Hence one has been able to find to values A and B such that $\mathbf{E}_{K,K'}((A\oplus\mathbb{T})\|A) = B\|(B\oplus\mathbb{T})$ for some fixed value $\mathbb{T}$, which would be possible with only negligible advantage for a random permutation.

A.2 The EME construction.

EME was proposed as CMC by Halevi and Rogaway [14], and improves on CMC since it is parallelizable. It uses to layers of ECB and an intermediate mixing layer. As CMC it is tweakable but we will set the tweak to 0^n in our attack.

CMC uses a block cipher $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ and turns it into a tweakable block cipher $\mathbf{E}$ with tweak space $\{0,1\}^n$, key space $\{0,1\}^k$, and message space $\bigcup_{m\geq 2}\{0,1\}^{mn}$. A message $P_1 \cdots P_m$ of m n-bit blocks is encrypted under key K and tweak $T = 0^n$ as follows:

1. $L \leftarrow 2E_K(0^n)$
2. for $i = 1$ to m do $PPP_i = E_k(P_i \oplus 2^{i-1}L)$
3. $MP \leftarrow PPP_1 \oplus PPP_2 \oplus \cdots \oplus PPP_m$
4. $MC \leftarrow E_K(MP)$
5. $M \leftarrow MP \oplus MC$
6. for $i = 2$ to m do $CCC_i = PPP_i \oplus 2^{i-1}M$
7. $CCC_1 \leftarrow MC \oplus CCC_2 \oplus \cdots \oplus CCC_m$
8. for $i = 1$ to m do $C_i \leftarrow E_K(CCC_i) \oplus 2^{i-1}L$
9. return $C_1 \cdots C_m$

The attack on EME for two-blocks messages proceeds as follows:

1. choose an arbitrary key K and compute $L = 2E_K(0^n)$
2. compute the value MP corresponding to $MC = 0^n$, $MP = E_K^{-1}(0^n)$; note that consequently $M = MP \oplus MC = MP$
3. fix $P_1 = 0^n$ and compute $PPP_1 = E_K(P_1 \oplus L)$
4. compute $PPP_2 = MP \oplus PPP_1$ and deduce $P_2 = E_K^{-1}(PPP_2) \oplus 2L$
5. compute $CCC_1 = CCC_2 = PPP_2 \oplus 2MP$
6. compute $C_1 = E_K(CCC_1) \oplus L$ and $C_2 = E_K(CCC_2) \oplus 2L = E_K(CCC_1) \oplus 2L$

Hence this attack enables to find P_1, P_2, C_1, C_2 such that $\mathbf{E}_K(P_1\|P_2) = C_1\|C_2$, $P_1 = 0^n$ and $C_1 \oplus C_2 = L \oplus 2L$ for some fixed value L. This would be possible with only negligible advantage for a truly random permutation.

B Proof of Lemma 1

We need to construct a simulator $\mathcal{S}$ for H and E, such that the two systems formed by $(E', (H, E))$ and $(\mathcal{E}', \mathcal{S})$ are indistinguishable, where $\mathcal{E}'$ is an ideal cipher with t-bit key and n-bit message.

Our simulator maintains an "H-table" of pairs (K', K) corresponding to answered queries $K = H(K')$; it also maintains an "E-table" of triples (K, X, Y) of answered queries $Y = E(K, X)$. Our simulator $\mathcal{S}$ answers the distinguisher's queries as follows:

1. $H(K')$ query: pick a random $K \leftarrow \{0,1\}^k$, record the pair (K', K) in the "H-table" and return K.
2. $E(K, X)$ query: if there exists a tuple (K, X, Y) in the E-table, return Y. Else, if there exists a tuple (K', K) in the "H-table", query the value $Y = \mathcal{E}'(K', X)$, record (K, X, Y) in the "E-table" and return Y. Else, pick a random $Y \leftarrow \{0,1\}^n$, record (K, X, Y) in the "E-table", while making sure that no collision is created for $E(K, \cdot)$; otherwise, a new Y is generated. The simulator returns Y.

3. $E^{-1}(K, Y)$ query: if there exists a tuple (K, X, Y) in the E-table (see below), return X. Else, if there exists a tuple (K', K) in the "H-table", query the value $Y = \mathcal{E}'^{-1}(K', Y)$, record (K, X, Y) in the "E-table" and return X. Else, pick a random $X \leftarrow \{0,1\}^n$, record (K, X, Y) in the "E-table", while making sure that no collision is created for $E^{-1}(K, \cdot)$, and return X.

This completes the description of the simulator. Now we show that the system $(E', (H, E))$ is indistinguishable from the system $(\mathcal{E}', \mathcal{S})$, where:

$$E'(K', X) = E(H(K'), X)$$

is the construction with extended key-size. We consider a distinguisher $\mathcal{D}$ making at most q queries and outputting a bit γ. We define a sequence $\mathtt{Game}_0$, $\mathtt{Game}_1$, ... of modified distinguisher games. In the first game $\mathtt{Game}_0$, the distinguisher interacts with the system formed by $(\mathcal{E}', \mathcal{S})$. We denote by S_i the event in game i that the distinguisher outputs $\gamma = 1$.

$\mathtt{Game}_0$: the distinguisher interacts with the simulator $\mathcal{S}$ and the ideal cipher $\mathcal{E}'$.

$\mathtt{Game}_1$: we slightly modify the way H and E queries are answered by the simulator. In $\mathtt{Game}_1$, given a query K' for H, instead of letting $K \leftarrow \{0,1\}^k$, the new simulator $\mathcal{S}'$ makes a query for random oracle H and returns $K = H(K')$. Similarly, for a $E(K, X)$ query, instead of generating a random $Y \leftarrow \{0,1\}^n$, the simulator queries ideal cipher E and returns $E(K, X)$; similarly for E^{-1}. Since we have simply replaced one set of random variables by a different, but identically distributed, set of random variables, we have:

$$\Pr[S_0] = \Pr[S_1]$$

$\mathtt{Game}_2$: we modify the way $\mathcal{E}'$ queries are answered by the system. Instead of returning $\mathcal{E}'(K', m)$ with ideal cipher $\mathcal{E}'$, the system returns $E'(K', m) = E(H(K'), m)$ by calling ideal cipher E and random oracle H.

We must show that the distinguisher's view has statistically close distribution in $\mathtt{Game}_1$ and $\mathtt{Game}_2$. For this, we consider the subsystem $\mathcal{T}$ with the ideal cipher $\mathcal{E}'$ and ideal cipher E and random oracle H in $\mathtt{Game}_1$, and the subsystem $\mathcal{T}'$ with construction E' and ideal cipher E and random oracle H in $\mathtt{Game}_2$. We show that the output of systems $\mathcal{T}$ and $\mathcal{T}'$ is statistically close; this in turn shows that the distinguisher's view has statistically close distribution in $\mathtt{Game}_1$ and $\mathtt{Game}_2$.

The outputs to E queries provided by subsystem $\mathcal{T}$ in $\mathtt{Game}_1$ and by subsystem $\mathcal{T}'$ in $\mathtt{Game}_2$ are the same, since in both cases these queries are answered by ideal cipher E. Therefore, we must show that the output to $\mathcal{E}'$ queries provided by $\mathcal{T}$ and $\mathcal{T}'$ have statistically close distribution, when the outputs to E and H queries provided by $\mathcal{T}$ or $\mathcal{T}'$ are fixed.

We consider a $\mathcal{E}'(K', m)$ query made either by the distinguisher or by the simulator (the argument for a $\mathcal{E}'^{-1}$ query is similar). In $\mathtt{Game}_2$ the answer c is computed as $E(H(K'), m)$; we have that conditioned on the event that no collision occurs for H, the output distribution of $E(H(K'), m)$ in $\mathtt{Game}_2$ is exactly the same as the distribution of $\mathcal{E}'(K', m)$ in $\mathtt{Game}_1$. Let denote by bad the event that a collision occurs for H; since there are at most q queries from the distinguisher, we have:

$$\Pr[\mathsf{bad}] \leq \frac{q^2}{2^k}$$

and we obtain:

$$|\Pr[S_2] - \Pr[S_1]| \leq \Pr[\mathsf{bad}] \leq \frac{q^2}{2^k}$$

$\mathtt{Game}_3$: the distinguisher interacts with system $(E', (H, E))$. We have that the system $(E', (H, E))$ provides the same output as the system in $\mathtt{Game}_2$, which gives:

$$\Pr[S_3] = \Pr[S_2]$$

From the previous inequalities, we obtain the following upper bound on the distinguisher's advantage:

$$|\Pr[S_3] - \Pr[S_0]| \leq \frac{q^2}{2^k}$$

which terminates the proof of Lemma 1.

C Proof of Theorem 3

We restrict ourself to distinguishers which do not make twice the same query (or the inverse query corresponding to a previous query). Note however that the distinguisher could query $L\|R$ first as a type I query (*i.e.* without asking for the transcript, and not seen by the simulator) and then as a type II query (when asking for the transcript, and sent to the simulator).

We first describe our simulator $\mathcal{S}$. It maintains an history of already defined values for E_1 and E_2. Upon a query of the distinguisher, it runs as follows:

- on input a direct query $(+, L\|R)$:
  1. query $P(L\|R) = S\|T$
  2. if $E_1(R, L)$ or $E_1^{-1}(R, S)$ is already defined, abort
  3. else $E_1(R, L) \leftarrow S$ and add $E_1(R, L) = S$ to the history
  4. if $E_2(S, R)$ or $E_2^{-1}(S, T)$ is already defined, abort
  5. else $E_2(S, R) \leftarrow T$ and add $E_2(S, R) = T$ to the history
  6. return $E_1(R, L) = S$, $E_2(S, R) = T$
- on input an inverse query $(-, S\|T)$:
  1. query $P^{-1}(S\|T) = L\|R$
  2. if $E_2(S, R)$ or $E_2^{-1}(S, T)$ is already defined, abort
  3. else $E_2^{-1}(S, T) \leftarrow R$ and add $E_2(S, R) = T$ to the history
  4. if $E_1(R, L)$ or $E_1^{-1}(R, S)$ is already defined, abort
  5. else $E_1^{-1}(R, S) \leftarrow L$ and add $E_1(R, L) = S$ to the history
  6. return $E_2^{-1}(S, T) = R$, $E_1^{-1}(R, S) = L$

We prove the indifferentiability through a sequence of games $\texttt{Game}_i$. We will note S_i the event that the distinguisher outputs 1 in $\texttt{Game}_i$. We start with:

$\texttt{Game}_0$: the distinguisher $\mathcal{D}$ interacts with $(P, \mathcal{S})$

$\texttt{Game}_1$: it is similar to $\texttt{Game}_0$ except that P now returns uniformly random answers. Looking at $\mathcal{D}$ and $\mathcal{S}$ as a distinguisher $\mathcal{D}'$ making at most q queries to P, it is easy to see that

$$|\Pr[S_1] - \Pr[S_0]| \leq \frac{q^2}{2 \cdot 2^{2n}}.$$

$\texttt{Game}_2$: we modify the way the answers to type I queries (those not seen by the simulator $\mathcal{S}$) are computed. Instead of being asked directly to the permutation P, they are "intercepted" by an algorithm $\mathcal{M}$ which forwards them to the simulator $\mathcal{S}$. $\mathcal{M}$ then computes the answer to $\mathcal{D}$ using the values returned by $\mathcal{S}$.

As long as the simulator does not abort, the output of $\mathcal{M}$ in $\texttt{Game}_2$ is the same as the output of P in $\texttt{Game}_1$. Moreover as long as the simulator does not abort, its output is also the same in $\texttt{Game}_2$ as in $\texttt{Game}_1$ since it does not depend on the additional queries made by $\mathcal{M}$. Hence:

$$|\Pr[S_2] - \Pr[S_1]| \leq \Pr_{\texttt{Game}_2}[\mathcal{S} \text{ aborts}].$$

Let $\texttt{bad}$ denote the event that there exists $1 \le j < i \le q$ such that the i-th and j-th queries of the distinguisher are such that

$$(R_i = R_j) \wedge (S_i = S_j) \wedge (L_i \neq L_j \vee T_i \neq T_j).$$

It is easy to see that as long as $\texttt{bad}$ does not happen, the simulator does not abort since it is always able to define the values of the internal ciphers. Therefore:

$$\Pr_{\texttt{Game}_2}[\mathcal{S} \text{ aborts}] \le \Pr_{\texttt{Game}_2}[\texttt{bad}]$$

Moreover, defining $\texttt{bad}_i$ as the event that $\texttt{bad}$ happens exactly at the i-th query of the distinguisher, we get:

$$\Pr_{\texttt{Game}_2}[\texttt{bad}] = \sum_{i=1}^{q} \Pr_{\texttt{Game}_2}[\texttt{bad}_\texttt{i}]$$

Assume that the i-th query is a direct one: $(+, L_i|R_i)$; the argument for inverse queries is similar. Note that this query cannot have been done to P yet. Since there are at most $i-1$ values S_j in the history of P and since P returns uniformly random answers, we obtain:

$$\Pr_{\texttt{Game}_2}[\texttt{bad}_\texttt{i}] \le \frac{i-1}{2^n}$$

which gives:

$$\Pr_{\texttt{Game}_2}[\texttt{bad}] = \sum_{i=1}^{q} \Pr_{\texttt{Game}_2}[\texttt{bad}_\texttt{i}] \le \frac{q^2}{2 \cdot 2^n}$$

and eventually:

$$|\Pr[S_2] - \Pr[S_1]| \le \frac{q^2}{2 \cdot 2^n}.$$

$\texttt{Game}_3$: we remove the permutation P and modify the simulator into a new simulator $\mathcal{S}'$ which, upon reception of a direct query $L\|R$, defines $S = E_1(R, L)$ uniformly at random and $T = E_2(S, R)$ uniformly at random, and symmetrically for inverse queries. Looking at $\mathcal{D}$ and $\mathcal{M}$ as a distinguisher $\mathcal{D}'$, one can see that the output of $\mathcal{S}$ in $\texttt{Game}_2$ and $\mathcal{S}'$ in $\texttt{Game}_3$ are exactly the same, which gives:

$$\Pr[S_3] = \Pr[S_2]$$

$\texttt{Game}_4$: the distinguisher interacts with the construction and the ideal ciphers E_1, E_2. We have that $\texttt{Game}_3$ and $\texttt{Game}_4$ are identical unless some collision happens in $\texttt{Game}_3$ when defining two values for the same key. Hence:

$$|\Pr[S_4] - \Pr[S_3]| \le 2\frac{q^2}{2 \cdot 2^n} = \frac{q^2}{2^n}.$$

Putting everything together yields

$$|\Pr[S_4] - \Pr[S_0]| \le 2\frac{q^2}{2 \cdot 2^{2n}} + \frac{q^2}{2^n} \le \frac{2q^2}{2^n}.$$

D Proof of Theorem 4

We consider an adversary making a sequence of exactly q queries. There are two types of queries $\mathcal{A}$ can make: either $(+, W, L, R)$ which is a query to $\tilde{\Psi_2}(K, W, L\|R)$, or $(-, W, S, T)$ which is a query to $\tilde{\Psi_2}^{-1}(K, W, S\|T)$. For the i-th query, we denote the by (W, L_i, R_i, S_i, T_i) the corresponding 5-uple.

$\mathtt{Game}_0$: the queries are answered using $\tilde{\Psi_2}$, as illustrated in Fig. 5.

$\mathtt{Game}_1$: we replace the tweakable block-ciphers E_1 and E_2 by 2 independent family of random permutations. From an attacker against $\tilde{\Psi_2}$ running in time t', we can construct an attacker against E_1 or E_2 running in time at most:

$$t = t' + \mathcal{O}(qn)$$

Since by assumption E_1 and E_2 are both (t, q, ε)-secure, we must have:

$$|\Pr[S_1] - \Pr[S_0]| \le 2 \cdot \varepsilon$$

$\mathtt{Game}_2$: the queries are now answered using the following process R. Given the i-th query:

1. If $(+, W, L, R)$ is queried and for some $1 \le j < i$ the j-th 4-uple is (W, L, R, S, T), then $S\|T$ is answered.
2. If $(-, W, S, T)$ is queried and for some $1 \le j < i$ the j-th 4-uple is (W, L, R, S, T), then $L\|R$ is answered.
3. If neither 1 nor 2 holds, then a uniformly distributed $2n$-bit string is returned.

We denote by bad the following event: there exists $1 \le i < j \le q$ such that the i-th answer $(W_i, L_i, R_i, S_i, T_i)$ and the j-th answer $(W_j, L_j, R_j, S_j, T_j)$ satisfy one of the following conditions:

1. $W_i = W_j$ and $R_i = R_j$ and $L_i \neq L_j$ and $S_i = S_j$
2. $W_i = W_j$ and $L_i = L_j$ and $R_i = R_j$ and $S_i \neq S_j$
3. $W_i = W_j$ and $S_i = S_j$ and $T_i = T_j$ and $R_i \neq R_j$

We have that conditioned on $\neg\mathsf{bad}$, the output of R in $\mathtt{Game}_2$ has the same distribution as the output of $\tilde{\Psi_2}$ in $\mathtt{Game}_1$, which gives:

$$\Pr[S_2|\neg\mathsf{bad}] = \Pr[S_1]$$

Moreover we have that $\Pr[\mathsf{bad}] \le q^2/2^n$, which gives using the Difference Lemma [26]:

$$|\Pr[S_2] - \Pr[S_1]| \le \Pr[\mathsf{bad}] \le \frac{q^2}{2^n}$$

$\mathtt{Game}_3$: the adversary interacts with a family of random permutation $\tilde{\Pi}'$. We consider the following event bad' in $\mathtt{Game}_2$: there exists $1 \le i < j \le q$ such that the i-th answer $(W_i, L_i, R_i, S_i, T_i)$ and the j-th answer $(W_j, L_j, R_j, S_j, T_j)$ satisfy one of the following conditions:

1. $W_i = W_j$ and $(L_i, R_i) = (L_j, R_j)$ and $(S_i, T_i) \neq (S_j, T_j)$
2. $W_i = W_j$ and $(L_i, R_i) \neq (L_j, R_j)$ and $(S_i, T_i) = (S_j, T_j)$

We have that conditioned on $\neg\mathsf{bad}'$, the distribution of R in $\mathtt{Game}_2$ and the distribution of P in $\mathtt{Game}_3$ are the same; therefore:

$$\Pr[S_2|\neg\mathsf{bad}'] = \Pr[S_3]$$

Moreover, we have $\Pr[\mathsf{bad}'] \le q^2/2^{2n}$, which gives:

$$|\Pr[S_3] - \Pr[S_2]| \le \Pr[\mathsf{bad}'] \le \frac{q^2}{2^{2n}}$$

Combining the previous inequalities, we get:

$$|\Pr[S_3] - \Pr[S_0]| \le 2 \cdot \varepsilon + \frac{q^2}{2^n} + \frac{q^2}{2^{2n}}$$

Therefore we can take $\varepsilon' = 2 \cdot \varepsilon + q^2/2^n + q^2/2^{2n}$, which terminates the proof of Theorem 4.

E Proof of Theorem 5

We prove the following theorem.

Theorem 7. *The 3-round block-cipher construction Ψ_3 (see Figure 6) is ϵ-indistinguishable from an ideal cipher with $\epsilon = (\frac{q}{2^n})^2$ for an attacker making q block-cipher queries with $q < 2^n$.*

The above theorem and the following sequence of games completes the proof of Theorem 5. We denote S_i the event that the distinguisher outputs 1 in $\texttt{Game}_i$.

$\texttt{Game}_0$: the queries are answered using $\tilde{\Psi}_3$, as illustrated in Fig. 5.

$\texttt{Game}_1$: we replace the tweakable block-ciphers E_1, E_2, E_3 by 3 independent family of random permutations. From an attacker against $\tilde{\Psi}_3$ running in time t', we can construct an attacker against E_1, E_2 or E_3 running in time at most:

$$t = t' + \mathcal{O}(qn)$$

Since by assumption E_1, E_2 and E_3 are all (t, q, ε)-secure, we must have:

$$|\Pr[S_1] - \Pr[S_0]| \leq 3 \cdot \varepsilon$$

$\texttt{Game}_2$: the adversary interacts with a family of random permutation $\tilde{\Pi}'$. By Theorem 7 we must have:

$$|\Pr[S_2] - \Pr[S_1]| \leq \frac{q^2}{2^{2n}}$$

E.1 Proof of Theorem 7

E_i's are actually random permutaion such that $E_i : Y \times Y \to Y$. So, $\Psi_3, \tilde{\Pi}' : Y \times Y \to Y \times Y$. Here $Y = \{0,1\}^n$. The $(i+1)^{th}$ query can either be a forward permutation query or a backward permutation query. Without loss of generality we can assume if $(i+1)^{th}$ query is a forward query (L_{i+1}, R_{i+1}) is distinct from (L, R) tuples in previous queries (responses), and similarly for a backward query (S_{i+1}, T_{i+1}) is distinct from (S, T) tuples in previous queries (responses). Whether the attacker interacts with Ψ_3 or ideal cipher $\tilde{\Pi}'$, input collision means output collision and output collision means input collision. So we can also assume $(i+1)^{th}$ output pair (s_{i+1}, t_{i+1}) is distinct from previous output pairs, previous input pairs are distinct among themselves and previous output pairs are distinct among themselves.

When the attacker interacts with Ψ_3 after i queries the underlying permutations E_1, E_2, E_3 have been fixed at some points, and at other points E_1, E_2, E_3's behave randomly. Also input-output of j^{th} query is actually a 4-tuple (L_j, R_j, s_j, t_j). We let $\mathcal{V}_i = ((L_1, R_1, s_1, t_1), \cdots, (L_i, R_i, s_i, t_i))$ be the attacker view after making the i^{th} query.

To prove Theorem 7 we will use the following lemma which shows for $i = 1, \cdots, q-1$ the advantage for $(i+1)^{th}$ query is actually bounded by $\frac{2i}{|Y|^2 - i}$.

Lemma 2. *For $i \in \{1, \cdots, q-1\}$, $\mathtt{Adv}_{i+1}$ be the distinguishing advantage for the $(i+1)^{th}$ query, then,*

$$\mathtt{Adv}_{i+1} = \frac{1}{2} \sum_{(s,t) \in Y^2 \setminus \mathcal{OP}_i} \left| \Pr[\Psi_3(L_{i+1}, R_{i+1}) = (s,t) | V_i] - \Pr[\tilde{\Pi}'(L_{i+1}, R_{i+1}) = (s,t) | V_i] \right| \leq \frac{2i}{|Y|^2 - i}$$

where $\mathcal{OP}_i = \{(s_1, t_1), \cdots, (s_i, t_i)\}$.

If for any attacker making q queries $\mathtt{Adv}(q)$ is the distinguishing advantage, then it is not hard to show that $\mathtt{Adv}(q) \leq \sum_{i=1}^{q-1} \mathtt{Adv}_{i+1}$. Hence by Lemma 2 we get,

$$\mathtt{Adv}(q) \leq \sum_{i=1}^{q-1} \mathtt{Adv}_{i+1} \leq \sum_{i=1}^{q-1} \frac{2i}{|Y|^2 - i} < \sum_{i=1}^{q-1} \frac{2i}{|Y|^2 - q} = \frac{q^2 - q}{|Y|^2 - q} < \frac{q^2}{|Y|^2} \qquad As\ q < |Y| = 2^n$$

$\square$

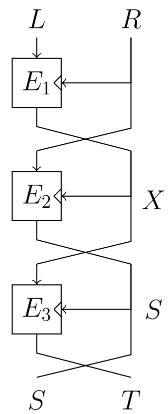

Fig. 6. The 3-round Feistel construction $\Psi_3(L, R)$.

E.2 Proof of Lemma 2

We will give a proof when the $(i+1)^{th}$ query is forward permutaion query, for backward permutation query the proof works in a similar fashion. From the attacker point of view

$$\bar{X} = (X_1, \cdots, X_i) = (E_1(R_1, L_1), \cdots, E_1(R_i, L_i))$$

is actually a random variable which *satisfies* $\mathcal{V}_i$.

Now we say any i-tuple $\bar{x} = (x_1, \cdots, x_i)$ is *feasible* if $\Pr[\bar{X} = \bar{x}|\mathcal{V}_i]$ is non zero. $\mathcal{F}$ be the set of all feasible $\bar{x}$. Now we will state another lemma, loosely speaking which gives an estimate of $\texttt{Adv}_{i+1}$ for an fixed $\bar{x} \in \mathcal{F}$.

Lemma 3. *For all $\bar{x} \in \mathcal{F}$,*

$$\sum_{(s,t)\in Y^2\setminus\mathcal{OP}_i} \left| \Pr[\Psi_3(L_{i+1}, R_{i+1}) = (s,t)|\mathcal{V}_i \wedge \bar{X} = \bar{x}] - \Pr[\tilde{\Pi}'(L_{i+1}, R_{i+1}) = (s,t)|\mathcal{V}_i] \right| \leq \frac{4i}{|Y|^2 - i}$$

Lemma 3 actually almost immediately proves Lemma 2 as follows,

$$\begin{aligned}
&\texttt{Adv}_{i+1} \\
=&\frac{1}{2} \sum_{(s,t)\in Y^2\setminus\mathcal{OP}_i} \left| \Pr[\Psi_3(L_{i+1}, R_{i+1}) = (s,t)|\mathcal{V}_i] - \Pr[\tilde{\Pi}'(L_{i+1}, R_{i+1}) = (s,t)|\mathcal{V}_i] \right| \\
\leq&\frac{1}{2} \sum_{(s,t)\in Y^2\setminus\mathcal{OP}_i} \sum_{\bar{x}\in\mathcal{F}} \left| \Pr[\Psi_3(L_{i+1}, R_{i+1}) = (s,t)|\mathcal{V}_i \wedge \bar{X} = \bar{x}] - \Pr[\tilde{\Pi}'(L_{i+1}, R_{i+1}) = (s,t)|\mathcal{V}_i] \right| \times \Pr[\bar{X} = \bar{x}|\mathcal{V}_i] \\
=&\frac{1}{2} \sum_{\bar{x}\in\mathcal{F}} \sum_{(s,t)\in Y^2\setminus\mathcal{OP}_i} \left| \Pr[\Psi_3(L_{i+1}, R_{i+1}) = (s,t)|\mathcal{V}_i \wedge \bar{X} = \bar{x}] - \Pr[\tilde{\Pi}'(L_{i+1}, R_{i+1}) = (s,t)|\mathcal{V}_i] \right| \times \Pr[\bar{X} = \bar{x}|\mathcal{V}_i] \\
\leq&\frac{1}{2} \sum_{\bar{x}\in\mathcal{F}} \frac{4i}{|Y|^2 - i} \times \Pr[\bar{X} = \bar{x}|\mathcal{V}_i] = \frac{2i}{|Y|^2 - i}
\end{aligned}$$

E.3 Proof of Lemma 3

At first we would like define some notations and state some important observations. For a feasible i-tuple $\bar{x} = (x_1, \cdots, x_i)$ we define $\mathcal{X}_{\bar{x}}^j = \{\alpha \in Y | \alpha \textit{ appears in } \bar{x} \textit{ exactly } j \textit{ times } \}$. Note,

$$\sum_{j=1}^{i} j|\mathcal{X}_{\bar{x}}^j| = i \tag{1}$$

Similarly considering the tuple $\bar{s} = (s_1, \cdots, s_i)$, we define $\mathcal{S}^j = \{\alpha \in Y | \alpha \textit{ appears in } \bar{s} \textit{ exactly } j \textit{ times }\}$. Also we have,

$$\sum_{j=1}^{i} j|\mathcal{S}^j| = i \tag{2}$$

When the attacker is making $(i+1)^{th}$ query the tuple $\bar{s}$ is already fixed. So we do not include the subscript $\bar{s}$ in the defination of $\mathcal{S}^j$. We define,

$$\mathcal{X}_{\bar{x}} = \bigcup_{j=1}^{i} \mathcal{X}_{\bar{x}}^j \qquad \textit{and} \qquad \mathcal{S} = \bigcup_{j=1}^{i} \mathcal{S}^j$$

Hence we also have,

$$|\mathcal{X}_{\bar{x}}| = \sum_{j=1}^{i} |\mathcal{X}_{\bar{x}}^j| \qquad \textit{and} \qquad |\mathcal{S}| = \sum_{j=1}^{i} |\mathcal{S}^j| \tag{3}$$

For the query (L_{i+1}, R_{i+1}), we say X_{i+1} is *new* with respect to $\bar{x}$ if $X_{i+1} \notin \mathcal{X}_{\bar{x}}$.We also say X_{i+1} is *k-collision* with respect to $\bar{x}$ if $X_{i+1} \in \mathcal{X}_{\bar{x}}^k$.

Now for a fixed $\bar{x} = (x_1, \cdots, x_i)$, depending on the value of X_{i+1} we define $\mathcal{S'}_{\bar{x}}(X_{i+1}) \subseteq \mathcal{S}$ as follows.

$$\mathcal{S'}_{\bar{x}}(X_{i+1}) = \{\alpha \in Y | \alpha = s_j \textit{ and } x_j = X_{i+1} \textit{ for some } j \in [1, i]\}$$

Intuitively $\mathcal{S'}_{\bar{x}}(X_{i+1})$ is the set of fixed outputs for $E_2(X_{i+1}, .)$. If X_{i+1} is new then $\mathcal{S'}_{\bar{x}}(X_{i+1})$ is empty, and if $X_{i+1} \in \mathcal{X}_{\bar{x}}^k$ then $|\mathcal{S'}_{\bar{x}}(X_{i+1})| = k$. This is true because if $|\mathcal{S'}_{\bar{x}}(X_{i+1})| < k$, then we would have $x_{j_1} = x_{j_2} = X_{i+1}$ and $s_{j_1} = s_{j_2}$ for some $j_1, j_2 \in [1, i]$ and $j_1 \neq j_2$. As $E_2(X_{i+1}, .)$ is a permutation this implies $R_{j_1} = R_{j_2} = r$. $E_1(r, .)$ being a permutation this implies $L_{j_1} = Lj_2$ as well, which is a contradiction because we have assumed previous input tuples are distinct.

Now we partition $\mathcal{S'}_{\bar{x}}(X_{i+1})$ as follows,

$$\mathcal{S'}_{\bar{x}}(X_{i+1}) = (\mathcal{S'}_{\bar{x}}(X_{i+1}) \cap \mathcal{S}^1) \cup (\mathcal{S'}_{\bar{x}}(X_{i+1}) \cap \mathcal{S}^2) \cup \cdots \cup (\mathcal{S'}_{\bar{x}}(X_{i+1}) \cap \mathcal{S}^i)$$

If we denote $|\mathcal{S'}_{\bar{x}}(X_{i+1}) \cap \mathcal{S}^j| = k_j$, and if $X_{i+1} \in \mathcal{X}_{\bar{x}}^k$ then clearly $\sum_{j=1}^{i} k_j = k$. We state this result as Lemma 4.

Lemma 4. *If X_{i+1} is new, then $\mathcal{S'}_{\bar{x}}(X_{i+1})$ is empty, and if $X_{i+1} \in \mathcal{X}_{\bar{x}}^k$ then $k = |\mathcal{S'}_{\bar{x}}(X_{i+1})| = \sum_{j=1}^{i} k_j$, where $k_j = |\mathcal{S'}_{\bar{x}}(X_{i+1}) \cap \mathcal{S}^j|$.*

Say $B_{i+1} \subseteq [1, i]$ be the set such that for all $j \in B_{i+1}$ we have $R_{i+1} = R_j$. As all the previous input tuples are distinct all x_j's are also distinct for any feasible $\bar{x} = (x_1, \cdots, x_i)$ and $j \in B_{i+1}$. Hence we get the following Lemma.

Lemma 5. *If $B_{i+1} \subseteq [1, i]$ be the set such that for all $j \in B_{i+1}$ we have $R_{i+1} = R_j$, then $|B_{i+1}| \leq |\mathcal{X}_{\bar{x}}|$.*

Proof.

$$|B_{i+1}| \leq \textit{ number of distinct elements in any feasible tuple } \bar{x} = |\mathcal{X}_{\bar{x}}|$$

□

Now we will break the expression

$$\sum_{(s,t) \in Y^2 \setminus \mathcal{OP}_i} \left| \Pr[\Psi_3(L_{i+1}, R_{i+1}) = (s,t) | \mathcal{V}_i \wedge \bar{X} = \bar{x}] - \Pr[\tilde{\Pi}'(L_{i+1}, R_{i+1}) = (s,t) | \mathcal{V}_i] \right|$$

in some separate terms so it will help us to compute the desired bound.

$$\begin{aligned}
&\sum_{(s,t)\in Y^2\setminus\mathcal{OP}_i} \big|\Pr[\Psi_3(L_{i+1},R_{i+1})=(s,t)|\mathcal{V}_i\wedge\bar{X}=\bar{x}] - \Pr[\tilde{\Pi}'(L_{i+1},R_{i+1})=(s,t)|\mathcal{V}_i]\big| \\
&\le \sum_{(s,t)\in Y^2\setminus\mathcal{OP}_i} \big|\Pr[\Psi_3(L_{i+1},R_{i+1})=(s,t)|\mathcal{V}_i\wedge\bar{X}=\bar{x}\wedge X_{i+1} \textit{ is new }] \\
&\quad - \Pr[\tilde{\Pi}'(L_{i+1},R_{i+1})=(s,t)|\mathcal{V}_i]\big| \times \Pr[X_{i+1} \textit{ is new } |\mathcal{V}_i\wedge\bar{X}=\bar{x}] \\
&\quad + \sum_{k=1}^{i}\sum_{(s,t)\in Y^2\setminus\mathcal{OP}_i} \big|\Pr[\Psi_3(L_{i+1},R_{i+1})=(s,t)|\mathcal{V}_i\wedge\bar{X}=\bar{x}\wedge X_{i+1} \textit{ is k-collision}] \\
&\quad - \Pr[\tilde{\Pi}'(L_{i+1},R_{i+1})=(s,t)|\mathcal{V}_i]\big| \times \Pr[X_{i+1} \textit{ is k-collision}|\mathcal{V}_i\wedge\bar{X}=\bar{x}] \\
&= A\times\Pr[X_{i+1} \textit{ is new } |\mathcal{V}_i\wedge\bar{X}=\bar{x}] + \sum_{k=1}^{i} C_k\times\Pr[X_{i+1} \textit{ is k-collision}|\mathcal{V}_i\wedge\bar{X}=\bar{x}] \qquad (4)
\end{aligned}$$

Where,

$$A = \sum_{(s,t)\in Y^2\setminus\mathcal{OP}_i} \big|\Pr[\Psi_3(L_{i+1},R_{i+1})=(s,t)|\mathcal{V}_i\wedge\bar{X}=\bar{x}\wedge X_{i+1} \textit{ is new }] - \Pr[\tilde{\Pi}'(L_{i+1},R_{i+1})=(s,t)|\mathcal{V}_i]\big| \qquad (5)$$

$$C_k = \sum_{(s,t)\in Y^2\setminus\mathcal{OP}_i} \big|\Pr[\Psi_3(L_{i+1},R_{i+1})=(s,t)|\mathcal{V}_i\wedge\bar{X}=\bar{x}\wedge X_{i+1} \textit{ is k-collision}] - \Pr[\tilde{\Pi}'(L_{i+1},R_{i+1})=(s,t)|\mathcal{V}_i]\big| \qquad (6)$$

Now our goal is to find good upper bound of A, C_k and $\Pr[X_{i+1}$ *is k-collision*$|\mathcal{V}_i\wedge\bar{X}=\bar{x}]$ for $k=1,\cdots,i$. Clearly,

$$\Pr[X_{i+1} \textit{ is k-collision}|\mathcal{V}_i\wedge\bar{X}=\bar{x}] = \frac{|\mathcal{X}^k_{\bar{x}}|}{|Y|-|B_{i+1}|} \qquad (7)$$

For upper bounding A, C_k, we will use the following lemma which states the value of $\Pr[\Psi_3(L_{i+1},R_{i+1})=(s,t)|\mathcal{V}_i\wedge\bar{X}=\bar{x}\wedge X_{i+1}$ *is new* $]$ and $\Pr[\Psi_3(L_{i+1},R_{i+1})=(s,t)|\mathcal{V}_i\wedge\bar{X}=\bar{x}\wedge X_{i+1}$ *is k-collision*$]$ for $k=1,\cdots,i$.

Lemma 6. *For any* $\mathcal{V}_i=((L_1,R_1,s_1,t_1),\cdots,(L_i,R_i,s_i,t_i))$ *and* $\bar{x}\in\mathcal{F}$, $\Psi_3(L_{i+1},R_{i+1})$ *has the following conditional probability distribution*

$$\Pr[\Psi_3(L_{i+1},R_{i+1})=(s,t)|\mathcal{V}_i\wedge\bar{X}=\bar{x}\wedge X_{i+1} \textit{ is new }] = \begin{cases} \frac{1}{|Y|}\times\frac{1}{|Y|} & \textit{if } (s,t)\in(Y\setminus\mathcal{S})\times Y. \quad \textit{Note } |(Y\setminus\mathcal{S})\times Y| = (|Y|-|\mathcal{S}|)|Y| \\ \frac{1}{|Y|}\times\frac{1}{|Y|-j} & \textit{if } (s,t)\in(\mathcal{S}^j\times Y)\setminus\mathcal{OP}_i. \quad \textit{Note } |(\mathcal{S}^j\times Y)\setminus\mathcal{OP}_i| = |\mathcal{S}^j|(|Y|-j) \end{cases}$$

$$\Pr[\Psi_3(L_{i+1},R_{i+1})=(s,t)|\mathcal{V}_i\wedge\bar{X}=\bar{x}\wedge X_{i+1} \textit{ is k-collision}]$$
$$= \begin{cases} \frac{1}{|Y|-k}\times\frac{1}{|Y|} & \textit{if } (s,t)\in(Y\setminus\mathcal{S})\times Y. \quad \textit{Note } |(Y\setminus\mathcal{S})\times Y| = (|Y|-|\mathcal{S}|)|Y| \\ \frac{1}{|Y|-k}\times\frac{1}{|Y|-j} & \textit{if } (s,t)\in((\mathcal{S}^j\setminus\mathcal{S}'_{\bar{x}}(X_{i+1}))\times Y)\setminus\mathcal{OP}_i. \quad \textit{Note } |((\mathcal{S}^j\setminus\mathcal{S}'_{\bar{x}}(X_{i+1}))\times Y)\setminus\mathcal{OP}_i| = (|\mathcal{S}^j|-k_j)(|Y|-j) \\ 0 & \textit{if } (s,t)\in(\mathcal{S}'_{\bar{x}}(X_{i+1})\times Y)\setminus\mathcal{OP}_i. \quad \textit{Note } |(\mathcal{S}'_{\bar{x}}(X_{i+1})\times Y)\setminus\mathcal{OP}_i| = k|Y|-\sum_{\ell=1}^{i}\ell k_\ell \end{cases}$$

where $\mathcal{OP}_i = \{(s_1,t_1),\cdots,(s_i,t_i)\}$, $\mathcal{S}^j = \{\alpha \in Y | \alpha$ *appears in* $(s_1,\cdots,s_i)$ *exactly* j *times* $\}$, $\mathcal{S} = \bigcup_{j=1}^{i}\mathcal{S}^j$, $\mathcal{S'}_{\bar{x}}(X_{i+1}) = \{\alpha \in Y | \alpha = s_j$ *and* $x_j = X_{i+1}$ *for some* $j \in [1,i]\}$ *and* $k_j = |\mathcal{S'}_{\bar{x}}(X_{i+1}) \cap \mathcal{S}^j|$.

Also we know, $\tilde{\Pi}'$ being a random permutation,

$$\Pr[\tilde{\Pi}'(L_{i+1},R_{i+1}) = (s,t)] = \frac{1}{|Y|^2 - i}$$

for all $(s,t) \in |Y|^2 \setminus \mathcal{OP}_i$. Now we are ready to estimate A and C_k.

By Equation (5), we have:

$$\begin{aligned} A &= (|Y| - |\mathcal{S}|)|Y| \times \Big(\frac{1}{|Y|^2 - i} - \frac{1}{|Y|^2}\Big) + \sum_{j=1}^{i}|\mathcal{S}^j|(|Y| - j) \times \Big(\frac{1}{|Y|(|Y| - j)} - \frac{1}{|Y|^2 - i}\Big) \\ &= \frac{(|Y| - \mathcal{S})i}{|Y|(|Y|^2 - i)} + \sum_{j=1}^{i}\frac{(j|Y| - i)|\mathcal{S}^j|}{|Y|(|Y|^2 - i)} \end{aligned}$$

By Equation (2) and (3), we have $\sum_{j=1}^{i} j|\mathcal{S}^j| = i$ and $\sum_{j=1}^{i}|\mathcal{S}^j| = |\mathcal{S}|$; this gives:

$$A = \frac{(|Y| - \mathcal{S})i}{|Y|(|Y|^2 - i)} + \frac{i|Y| - i|\mathcal{S}|}{|Y|(|Y|^2 - i)} \leq \frac{2i}{|Y|^2 - i}$$

By Equation (6), we have:

$$\begin{aligned} C_k &= \Big(k|Y| - \sum_{\ell=1}^{i}\ell k_\ell\Big) \times \Big(\frac{1}{|Y|^2 - i} - 0\Big) + (|Y| - |\mathcal{S}|)|Y| \times \Big(\frac{1}{(|Y| - k)|Y|} - \frac{1}{|Y|^2 - i}\Big) \\ &\quad + \sum_{j=1}^{i}(|\mathcal{S}^j| - k_j)(|Y| - j) \times \Big(\frac{1}{(|Y| - k)(|Y| - j)} - \frac{1}{|Y|^2 - i}\Big) \\ &\leq \frac{k|Y|}{|Y|^2 - i} + \frac{(k|Y| - i)(|Y| - |\mathcal{S}|)}{(|Y| - k)(|Y|^2 - i)} + \sum_{j=1}^{i}\frac{(|\mathcal{S}^j| - k_j)(k|Y| + j|Y| - i - kj)}{(|Y| - k)(|Y|^2 - i)} \\ &= \frac{k|Y|}{|Y|^2 - i} + \frac{(k|Y| - i)(|Y| - |\mathcal{S}|)}{(|Y| - k)(|Y|^2 - i)} + \sum_{j=1}^{i}\frac{(|\mathcal{S}^j| - k_j)(k|Y| - i)}{(|Y| - k)(|Y|^2 - i)} + \sum_{j=1}^{i}\frac{j(|\mathcal{S}^j| - k_j)}{|Y|^2 - i} \end{aligned}$$

By Equation (2) and (3), we have $\sum_{j=1}^{i} j|\mathcal{S}^j| = i$ and $\sum_{j=1}^{i}|\mathcal{S}^j| = |\mathcal{S}|$, and by Lemma 4, we have $\sum_{j=1}^{i} k_j = k$; this gives:

$$\begin{aligned} C_k &\leq \frac{k|Y|}{|Y|^2 - i} + \frac{(k|Y| - i)(|Y| - |\mathcal{S}|)}{(|Y| - k)(|Y|^2 - i)} + \frac{(k|Y| - i)(|\mathcal{S}| - k)}{(|Y| - k)(|Y|^2 - i)} + \frac{i}{|Y|^2 - i} \\ &= \frac{k|Y|}{|Y|^2 - i} + \frac{k|Y| - i}{|Y|^2 - i} + \frac{i}{|Y|^2 - i} \\ &= \frac{2k|Y|}{|Y|^2 - i} \end{aligned}$$

Now putting the upper bounds of A and C_k in Equation (4) we get,

$$\sum_{(s,t)\in Y^2\setminus\mathcal{OP}_i} |\Pr[\Psi_3(L_{i+1},R_{i+1})=(s,t)|\mathcal{V}_i\wedge\bar{X}=\bar{x}]-\Pr[\tilde{\Pi}'(L_{i+1},R_{i+1})=(s,t)|\mathcal{V}_i]|$$

$$\leq \frac{2i}{|Y|^2-i}\times\Pr[X_{i+1} \textit{ is new } |\mathcal{V}_i\wedge\bar{X}=\bar{x}]+\sum_{k=1}^{i}\frac{2k|Y|}{|Y|^2-i}\times\Pr[X_{i+1} \textit{ is k-collision}|\mathcal{V}_i\wedge\bar{X}=\bar{x}]$$

$$=\frac{2i}{|Y|^2-i}\big(\Pr[X_{i+1} \textit{ is new } |\mathcal{V}_i\wedge\bar{X}=\bar{x}]+\sum_{k=1}^{i}\Pr[X_{i+1} \textit{ is k-collision}|\mathcal{V}_i\wedge\bar{X}=\bar{x}]\big)$$

$$+\sum_{k=1}^{i}\frac{2(k|Y|-i)}{|Y|^2-i}\times\Pr[X_{i+1} \textit{ is k-collision}|\mathcal{V}_i\wedge\bar{X}=\bar{x}]$$

$$=\frac{2i}{|Y|^2-i}+\sum_{k=1}^{i}\frac{2(k|Y|-i)}{|Y|^2-i}\times\frac{|\mathcal{X}_{\bar{x}}^k|}{|Y|-|B_{i+1}|} \qquad \textit{By Equation (7)}$$

$$=\frac{2i}{|Y|^2-i}+\frac{2|Y|\sum_{k=1}^{i}k|\mathcal{X}_{\bar{x}}^k|-2i\sum_{k=1}^{i}|\mathcal{X}_{\bar{x}}^k|}{(|Y|^2-i)(|Y|-|B_{i+1}|)}$$

$$=\frac{2i}{|Y|^2-i}+\frac{2i}{|Y|^2-i}\times\frac{|Y|-|\mathcal{X}_{\bar{x}}|}{|Y|-|B_{i+1}|} \qquad \textit{By Equation (1) \& (3)}, \sum_{k=1}^{i}k|\mathcal{X}_{\bar{x}}^k|=i \textit{ and } \sum_{k=1}^{i}|\mathcal{X}_{\bar{x}}^k|=|\mathcal{X}_{\bar{x}}|$$

$$\leq\frac{4i}{|Y|^2-i} \qquad \textit{By Lemma 5 we have } |\mathcal{X}_{\bar{x}}|\geq|B_{i+1}|$$

E.4 Proof of Lemma 6

Note $\Psi_3(L_{i+1},R_{i+1})=(s,t)$ actually means,

$$s=E_2(X_{i+1},R_{i+1})$$
$$t=E_3(s,X_{i+1})$$

We know E_2,E_3 are random permutations. That means if at some point of time, for some particular key K, $(I_1,O_1),\cdots,(I_\ell,O_\ell)$ input-output pairs have already been fixed for the random permutation $E_2(K,.)$, then at the next invokation of $E_2(K,.)$,

$$\Pr[E_2(K,x)=y]=\frac{1}{|Y|-\ell}$$

for all $x\in Y\setminus\{I_1,\cdots,I_\ell\}$ and $y\in Y\setminus\{O_1,\cdots,O_\ell\}$. The same is true for E_3 random permutation. Hence if X_{i+1} is new, then

$$\Pr[E_2(X_{i+1},R_{i+1})=s]=\frac{1}{|Y|}$$

for all $s\in Y$.
If X_{i+1} is k-collision, then

$$\Pr[E_2(X_{i+1},R_{i+1})=s]=\frac{1}{|Y|-k}$$

for all $s\in Y\setminus\mathcal{S}'_{\bar{x}}(X_{i+1})$. And

$$\Pr[E_2(X_{i+1},R_{i+1})=s]=0$$

for all $s\in\mathcal{S}'_{\bar{x}}(X_{i+1})$, because otherwise we have a duplicate query.
Similarly, if $s\in Y\setminus\mathcal{S}$, then

$$\Pr[E_3(s,X_{i+1})=t]=\frac{1}{|Y|}$$

for all $t \in Y$.
And if $s \in \mathcal{S}^j$, then

$$\Pr[E_3(s, X_{i+1}) = t] = \frac{1}{|Y| - j}$$

for all $t \in Y \setminus$ *Set of t values corresponding to s in $\mathcal{V}_i$*. Using the above probability values it is easy to see why Lemma 6 holds. □

F Proof of Theorem 6

We consider a (q, t', ε')-adversary $\mathcal{A}'$ against our construction $\tilde{E}'$. We must describe a (q, t, ε_1)-adversary $\mathcal{A}$ against the original tweakable block cipher $\tilde{E}$. Our adversary $\mathcal{A}$ has oracle access to F and F^{-1}, where either $F = \tilde{E}(K, \cdot, \cdot)$ or $F = \tilde{\Pi}(\cdot, \cdot)$; it must output a bit γ, representing its guess as to whether $F = \tilde{E}(K, \cdot, \cdot)$ or $F = \tilde{\Pi}(\cdot, \cdot)$.

We first generate a random $h \in \mathcal{H}$. When $\mathcal{A}'$ queries for $F'(W', m)$, we compute $h(W')$ and return $F(h(W'), m)$, and similarly for a F'^{-1} query. Eventually, $\mathcal{A}'$ outputs a bit γ, which is returned by our adversary $\mathcal{A}$.

When $F = \tilde{E}(K, \cdot, \cdot)$, we have that adversary $\mathcal{A}'$ interacts with $F' = \tilde{E}'((K, h), \cdot, \cdot)$, exactly as in the security definition, which gives:

$$\Pr[\gamma = 1 | F = \tilde{E}(K, \cdot, \cdot)] = \Pr[\gamma = 1 | F' = \tilde{E}'((K, h), \cdot, \cdot)]$$

When $F = \tilde{\Pi}(\cdot, \cdot)$ we must show that the view of adversary $\mathcal{A}'$ is statistically close to that of $\mathcal{A}'$ in the original security definition. In the security definition, $\mathcal{A}'$ interacts with a family $\tilde{\Pi}'$ of independent random permutation parametrised with W'. Here instead the adversary $\mathcal{A}'$ interacts with $\tilde{\Pi}(h(\cdot), \cdot)$. The key observation is that if no collision occurs for h, then the distribution seen by $\mathcal{A}'$ is exactly the same as the one obtained from $\tilde{\Pi}'$. Let denote by bad the event that such collision occurs; since $\mathcal{H}$ is a family of ε_2-almost universal hash functions and there are at most q queries, we have:

$$\Pr[\mathsf{bad}] \le q^2 \cdot \varepsilon_2$$

Moreover we obtain:

$$\Pr[\gamma = 1 | F = \tilde{\Pi} \wedge \neg \mathsf{Bad}] = \Pr[\gamma = 1 | F' = \tilde{\Pi}']$$

which gives:

$$|\Pr[\gamma = 1 | F = \tilde{\Pi}] - \Pr[\gamma = 1 | F' = \tilde{\Pi}']| \le \Pr[\mathsf{bad}] \le q^2 \cdot \varepsilon_2$$

Eventually denoting:

$$\delta = |\Pr[\gamma = 1 | F = \tilde{E}(K, \cdot, \cdot)] - \Pr[\gamma = 1 | F = \tilde{\Pi}]|$$
$$\delta' = |\Pr[\gamma = 1 | F' = \tilde{E}'((K, h), \cdot, \cdot)] - \Pr[\gamma = 1 | F' = \tilde{\Pi}']|$$

we obtain:

$$\delta' \le \delta + q^2 \cdot \varepsilon_2$$

Since by assumption $\delta \le \varepsilon_1$, we obtain $\delta' \le \varepsilon_1 + q^2 \cdot \varepsilon_2$; therefore we can take:

$$\varepsilon' = \varepsilon_1 + q^2 \cdot \varepsilon_2$$

which terminates the proof of Theorem 6.