An extended abstract of this work appears in *Advances in Cryptology – Crypto 98 Proceedings*, Lecture Notes in Computer Science Vol. 1462, H. Krawczyk ed., Springer-Verlag, 1998. This is the full version.

Security Amplification by Composition: The case of Doubly-Iterated, Ideal Ciphers

W. Aiello[*] M. Bellare[†] G. Di Crescenzo[‡] R. Venkatesan[§]

June 1998

Abstract

We investigate, in the Shannon model, the security of constructions corresponding to double and (two-key) triple DES. That is, we consider $F_{k_1}(F_{k_2}(\cdot))$ and $F_{k_1}(F^{-1}_{k_2}(F_{k_1}(\cdot)))$ with the component functions being ideal ciphers. This models the resistance of these constructions to "generic" attacks like meet in the middle attacks.

We obtain the first proof that composition actually increases the security in some meaningful sense. We compute a bound on the probability of breaking the double cipher as a function of the number of computations of the base cipher made, and the number of examples of the composed cipher seen, and show that the success probability is the square of that for a single key cipher. The same bound holds for the two-key triple cipher. The first bound is tight and shows that meet in the middle is the best possible generic attack against the double cipher.

Keywords: Ciphers, cascaded ciphers, Shannon model, information theory, DES, Double DES, meet in the middle attacks.

[*] Bellcore, 445 South St., Morristown, NJ 07960, USA. E-Mail: `aiello@bellcore.com`

[†] Dept. of Computer Science & Engineering, University of California at San Diego, 9500 Gilman Drive, La Jolla, CA 92093, USA. E-Mail: `mihir@cs.ucsd.edu`. URL: `http://www-cse.ucsd.edu/users/mihir`. Supported in part by NSF CAREER Award CCR-9624439 and a 1996 Packard Foundation Fellowship in Science and Engineering.

[‡] Dept. of Computer Science & Engineering, University of California at San Diego, 9500 Gilman Drive, La Jolla, CA 92093, USA. E-Mail: `giovanni@cs.ucsd.edu`. Supported in part by above mentioned grants of Bellare.

[§] Microsoft Research, One Microsoft Way, Redmond, WA 98052, USA. E-Mail: `venkie@microsoft.com`

Contents

1 Introduction

A block cipher is a map $F : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$. Here κ is the key size and n is the block size. Each κ-bit key k induces a map $F_k(\cdot) \stackrel{\text{def}}{=} F(k,\cdot) : \{0,1\}^n \to \{0,1\}^n$ which is a *permutation* on $\{0,1\}^n$. Let F^{-1} denote the inverse cipher, meaning $F^{-1}(k,\cdot) \stackrel{\text{def}}{=} F_k^{-1}$ is the inverse map of $F_k(\cdot)$. For example, DES is such a cipher with $\kappa = 56$ and $n = 64$.

It is common practice to compose ciphers in attempts to increase security. The result of composition is a new cipher, with a larger key size but the same block size. Here are the two most popular mechanisms, corresponding, respectively, to double DES and (two-key) triple DES:

- *Double F, or the 2-cascade cipher:* $\mathsf{Dbl}\text{-}F : \{0,1\}^{2\kappa} \times \{0,1\}^n \to \{0,1\}^n$ is defined by
$$\mathsf{Dbl}\text{-}F_{k_1,k_2}(x) \;=\; F_{k_1}(F_{k_2}(x)) \;.$$
- *Two-key triple F:* $\mathsf{Trp}^2\text{-}F : \{0,1\}^{2\kappa} \times \{0,1\}^n \to \{0,1\}^n$ is defined by
$$\mathsf{Trp}^2\text{-}F_{k_1,k_2}(x) \;=\; F_{k_1}(F_{k_2}^{-1}(F_{k_1}(x))) \;.$$

Let $\mathsf{Op}\text{-}F : \{0,1\}^{\kappa^*} \times \{0,1\}^n \to \{0,1\}^n$ denote one of these, where $\kappa^* = 2\kappa$ and $\mathsf{Op} \in \{\mathsf{Dbl}, \mathsf{Trp}^2\}$. What we want to know is: How good a cipher is $\mathsf{Op}\text{-}F$? Has the composition and the increased key length actually bought us anything?

Generic versus cryptanalytic attacks. There are several possible approaches to this question, depending on what kinds of attacks one wants to take into account. There are two main classes of attacks:

- *Cryptanalytic attacks:* Like differential [3, 4] and linear [9] cryptanalysis
- *Generic attacks:* Like exhaustive key search and meet-in-the-middle attacks.

Generic attacks are, roughly, those that don't exploit the structure of the cipher, but work against any cipher, even an ideal one. More precisely, we define generic attacks as those that succeed in the Shannon model of an ideal cipher discussed below.

The strength of specific composed ciphers like double DES against cryptanalytic attacks is not known; certainly, one does not expect a proof of such strength. The strength of the composed cipher against generic attacks, in contrast, can at least in principle be determined, by an analysis in the Shannon model, since it is a purely information theoretic question. However, the technical problems here are quite challenging; in particular, it is not even known that composition increases the strength of a cipher at all in this model.

In this paper we tackle this question, analyzing, in the Shannon model, two-key based compositions such as the above. We will prove upper bounds on the probability of "breaking" the composed cipher as a function of the "effort" invested by the adversary, with both terms in quotes to be properly defined. Our results are the first to show that cipher composition in the Shannon model actually increases security: the success probability of an adversary, as a function of her resources, is significantly lower than in the case of a single key cipher. For the double cipher our results are actually tight (optimal) and show that meet in the middle is the best possible generic attack on this cipher. We now define the model, and state our results, more precisely.

1.1 The model

We model F as an *ideal* block cipher in the sense of Shannon. This means $F(k,\cdot)$ is a *random* permutation on $\{0,1\}^n$, for each k. More precisely, let $\mathrm{PERM}(n)$ be the set of all permutations on

$\{0,1\}^n$. Then, for each κ-bit key k, select, uniformly and independently, a map from PERM(n), and assign F_k this value. So F consists of 2^κ maps, each a random permutation.

Now, we want to ask how good is Op as a composition operator. How can we measure this? We do so in a strong adversarial model, which allows the adversary chosen plaintext attacks on Op-F. Furthermore, success for the adversary A does not mean she has to find the key: it suffices that A identify some "weakness" in the cipher. This means A should be able to detect any deviation in Op-$F_{k^*}(\cdot)$ from a truly random permutation, when k^* is a random and hidden key for Op-F.

Formally, give the adversary oracles for F, F^{-1}. (This models her ability to compute the original cipher at any points she likes.) Also give her an oracle we call $E : \{0,1\}^n \to \{0,1\}^n$, which can take one of two forms:

- *World 1:* Set $E = \mathsf{Op}\text{-}F_{k^*}(\cdot)$ where $k^* \in \{0,1\}^{\kappa^*}$ is a randomly chosen key for cipher Op-F
- *World 2:* Set $E = \pi$ where π is a permutation chosen randomly from PERM(n).

Put the adversary A in one of these worlds, and ask her which one she is in. If she can't tell then Op-$F_{k^*}(\cdot)$ is behaving like a random permutation, meaning it is good. Formally, define the *advantage* of A as $P_1 - P_2$, where P_i is the probability that A outputs 1 in world $i \in \{1,2\}$. (The probability is over the choice of the oracles in each case.) Call A a (q,t)*-adversary* if it makes at most t queries to the F, F^{-1} oracles and at most q queries to the E oracle. (Note in practice t is likely to be much larger than q since F, F^{-1} queries are just DES computations and E queries are plaintexts in a chosen plaintext attack. We always assume $q \geq 1$ since otherwise the advantage of the adversary is zero no matter what the construction.) Define

$$\mathbf{Sec}(\mathsf{Op}, \kappa, n, q, t)$$

as the maximum advantage attainable by any (q,t)-adversary. This is the key quantity; it is a function we call the *security* of the operator Op. The question is to determine this function as accurately as possible. In particular we want to upper bound it as a function of the adversary resources q, t and the block cipher parameters κ, n.

Before stating the results we stress the power of the model. It allows chosen plaintext attacks on the composite cipher Op-F. Note it certainly captures common attacks like birthday attacks and meet-in-the-middle attacks, but also more sophisticated attacks which could be adaptive.

Notice that the advantage of a (q,t) adversary in attacking the single key cipher F itself in this model (namely $E = F_k$ for a random κ bit string k in world 1) will be (at most) $t/2^\kappa$. This is the mark we have to beat if we want to show that the composed cipher is stronger than the original one.

1.2 The results

It is known that the strength of the composed cipher is at least that of the first [10], but prior to this work it was not known whether the advantage of a (q,t) adversary versus Dbl-F was any lower than its advantage versus the single key cipher F itself. Here we are able to show that composition actually increases security, in the ideal cipher model described above.

THE DOUBLE KEY CIPHER. Recall that the double F cipher Dbl-F has 2κ bits of key. Our main result is Theorem 3.1, which says that $\mathbf{Sec}(\mathsf{Op}, \kappa, n, q, t)$ is at most $t^2/2^{2\kappa}$. Namely, no (q,t)-adversary attacking the double cipher can achieve an advantage greater than $t^2/2^{2\kappa}$.

We also show this bound is essentially tight, due to (a variant of) the meet in the middle attack. Theorem A.2 presents an adversary who runs this attack, and analyzes it to show that its advantage is within a small factor of $t^2/2^{2\kappa}$.

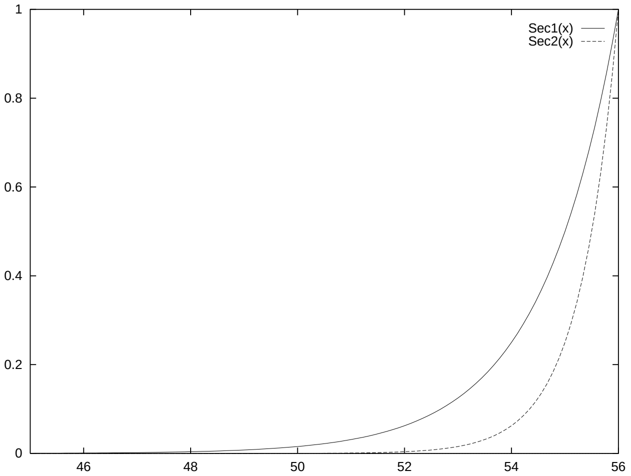

Figure 1: $\mathrm{Sec1}(x)$ *(the upper curve) and* $\mathrm{Sec2}(x)$ *(the lower curve) are, respectively, the maximal possible advantage obtainable by an adversary in breaking the single and double key ideal ciphers, respectively, as a function of* $x = \log_2(t)$*, the logarithm of the number of cipher computations made. We are using a key length of* $\kappa = 56$*. We see that* Sec2 *lies below* Sec1 *but they meet at* 1*. The text provides the exact formulas for these quantities.*

Note that the maximum possible advantage of an adversary attacking the double cipher case is the square of the maximum possible advantage of an adversary of the same resources attacking the original single key cipher. Thus, it is considerably smaller in most cases. (For example if $\kappa = 56$ and $t = 2^{45}$ then the former is 2^{-22} and the latter is 2^{-11}. Or, looking at it another way, to achieve an advantage of 2^{-11} against the double cipher you need at least 2^{50} queries, while to get the same advantage against the single cipher you need only 2^{45} queries.) To see the relation better, we plot in Figure 1 the maximal advantage $t/2^{\kappa}$ of an adversary in breaking the original single key cipher, and the maximal advantage $t^2/2^{2\kappa}$ of an adversary in breaking the double cipher, as a function of $x = \log_2(t)$.

Notice that the upper bound on the advantage in the double key case hits one (meaning, the scheme can be broken) when $t = 2^{\kappa}$. This is expected: that's the meet in the middle attack. Of course, that's the same point at which the advantage hits one for the original single key cipher. (In this case due to an exhaustive key search attack.) Thus, the "effective key length" of the double cipher is not more than that of the single one. That does not mean that security has not increased. Security is not a number, but a function of the resources invested, and our analysis and Figure 1 show that for values of t below 2^{κ} the chance of breaking the double cipher is smaller than that of breaking the original one.

The two-key triple cipher. We show that the same bound holds for the two-key triple cipher, meaning the advantage of a (q, t) adversary is bounded by $t^2/2^{2\kappa}$. This shows that here too there is an improvement in the security curve as a function of t. In this case our bound is tight for the case $t \approx q$ but not tight in general.

The m-fold cascade. The m-fold composition of cipher F is the cipher with key $k_1, \ldots, k_m$ defined by $F_{k_1,\ldots,k_m} = F_{k_1} \circ F_{k_2} \circ \cdots \circ F_{k_m}$. The techniques above extend to show that the advantage of an (q, t) adversary is at most $t^m/2^{m\kappa}$. This shows that the advantage grows more and more slowly as m increases. However, for $m \geq 3$ the result is not tight; we expect the 3-fold composed cipher to have an even greater strength than this indicates. Thus, we won't discuss this result any more in this paper.

THE FUTURE. The analysis of the two key ciphers we present here is a start on a problem that appears to be quite technically challenging. In the future we would like to see tight bounds on the advantage for the m-fold composition for $m \geq 3$ and also for the two-key triple cipher in the case $q << t$, but the distance needed to get there seems quite large at this time.

1.3 Related work

The model used here is that of Kilian and Rogaway [8], who in turn built on Even and Mansour [7], although the basic idea of course goes back to Shannon [13].

Kilian and Rogaway [8] analyze Rivest's DESX cipher in this model and show it has a large effective key length. If generic (or, as they call them, key search) attacks are the only concern, DESX is cheaper than Double or Triple DES, but DESX is just as vulnerable as DES to differential and linear cryptanalysis. The (apparent) strength of Double and two-key triple DES against cryptanalysis coupled with the proven strength against generic attacks seem to make a strong combination that is absent for DESX.

The basic meet in the middle attacks are due to [5, 12]. Even and Goldreich provide some time-space tradeoffs for meet-in-the-middle attacks [6], and Van Oorschot and Wiener [14] reduce the space requirements.

Even and Goldreich [6] had shown that the cascade of m ciphers is at least as strong as its strongest component. Maurer and Massey [10] argued that this result required restrictions in the model, and also showed that the cascade is at least as strong as its first component. Our work is the first to show that the cascade can be *stronger* than the original cipher.

Our analysis builds on techniques of [8] and [2]. Applications aside, we feel that we are looking at a basic information theoretic question, namely the power of cascaded ciphers.

A preliminary version of our paper appeard as [1]. Material omitted there due to space restrictions is included here.

1.4 Discussion on Implications of our result

What implications do these results have for the security of real ciphers like DES? This is a question that needs to be addressed with some care. After all, DES is not an ideal cipher.

We are not claiming to have "proven Double DES" secure; that obviously is not a realistic possibility. Our results might be interpreted as saying that the existence of a generic attack against DES that is substantially better than the meet in the middle attack would imply that there are serious weaknesses in the random behavior of DES that so far has empirical support.

The class of generic attacks is broad enough to be interesting, including meet-in-the-middle attacks and variants of it. But it does not include cryptanalytic attacks like differential or linear cryptanalysis, which exploit the structure of the cipher. However, one should note that at the moment the best attacks against Double and Triple DES are not the cryptanalytic ones, but the generic meet-in-the-middle attacks. And our results can be interpreted as ruling out improvements along those lines.

The adversary resources we consider here are the number of cipher computations t and the number of available plaintext-ciphertext pairs of the attacked cipher available, q. These are the most basic resources, and also the natural ones to consider in an information theoretic setting. One might attempt to consider other resources like space (e.g. when it is small compared to the number of queries), or make a distinction between parallelizable and sequential computations. Addressing these issues would change the nature of the problem to the point where it is difficult to see how it might be treated by techniques similar to the ones we use.

1.5 Organization

The double cipher analysis is in Section 3. There we state and prove the upper bound. In Appendix A we present the meet in the middle attack analysis that shows the upper bound is tight. The analysis of the two-key triple cipher is in Appendix B.

2 Definitions

General. We use standard notation for expressing probabilistic experiments and algorithms. Namely if S is a probability space then $x \leftarrow S$ denotes the operation of drawing x at random according to distribution S. If S is a set we use the same notation with the understanding that S is imbued with the uniform distribution. If S is not a set or probability space (in particular if x is a string or function) then $x \leftarrow S$ is simply an assignment statement.

Block ciphers. For an integer $n \geq 1$ let $\mathrm{PERM}(n)$ denote the set of all maps $\pi : \{0,1\}^n \to \{0,1\}^n$ that are permutations, meaning both one-to-one and onto. A function $F : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$ is a *block cipher* if for each *key* $k \in \{0,1\}^\kappa$, the function $F(k,\cdot) : \{0,1\}^n \to \{0,1\}^n$ is a permutation on $\{0,1\}^n$, meaning a member of $\mathrm{PERM}(n)$. Here, n is the block length of the cipher and κ is the key length of the cipher. Think of F as a 2^κ by 2^n table, with entry (k,x) containing $F(k,x)$. Each row is a permutation of $\{0,1\}^n$. For convenience, define $F_k : \{0,1\}^n \to \{0,1\}^n$, for each $k \in \{0,1\}^\kappa$, by $F_k(x) = F(k,x)$. This is the permutation in the k-th row. Although the function F does not have an inverse function, it does have a well defined inverse block cipher. When it is clear from context that F is a block cipher then we will let $F^{-1} : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$ denote the block cipher inverse of F, defined as follows: $F^{-1}(k,y) = F_k^{-1}(y)$. That is, $F^{-1}(k,y) = x$ iff $F(k,x) = y$.

Let $\mathrm{BC}(\kappa,n)$ denote the set of all block ciphers with key length κ and block length n. This is viewed as a probability space under the uniform distribution. Thus $F \leftarrow \mathrm{BC}(\kappa,n)$ means that F is selected according to the following experiment:

```
for all $k \in \{0,1\}^\kappa$ do $F(k,\cdot) \leftarrow \mathrm{PERM}(n)$.
```

Operators: Double and triple. We are interested in transformations, or operators, which map one block cipher to another. In general such an operator is a map Op taking a block cipher $F \in \mathrm{BC}(\kappa,n)$ and returning another block cipher, which we denote by $\mathsf{Op}\text{-}F$, and which belongs to $\mathrm{BC}(\kappa^*,n^*)$ for some values of κ^*, n^* that depend on κ, n and Op. (In this paper it will always be the case that $n^* = n$.) We now define the two central operators for this paper.

The *double composition operator* $\mathsf{Dbl} : \mathrm{BC}(\kappa,n) \to \mathrm{BC}(2\kappa,n)$ is defined by $\mathsf{Dbl}\text{-}F_{k_1k_2} = F_{k_1} \circ F_{k_2}$. In other words, $\mathsf{Dbl}\text{-}F(k_1k_2,x) = F(k_1,F(k_2,x))$ for every $k_1,k_2 \in \{0,1\}^\kappa$ and every $x \in \{0,1\}^n$. The *two key, triple composition operator* $\mathsf{Trp}^2 : \mathrm{BC}(\kappa,n) \to \mathrm{BC}(2\kappa,n)$ is defined by $\mathsf{Trp}^2\text{-}F_{k_1k_2} = F_{k_1} \circ F_{k_2}^{-1} \circ F_{k_1}$. In other words, $\mathsf{Trp}^2\text{-}F(k_1k_2,x) = F(k_1,F^{-1}(k_2,F(k_1,x)))$ for every $k_1,k_2 \in \{0,1\}^\kappa$ and every $x \in \{0,1\}^n$. Note both these ciphers have key length twice that of the original cipher.

Security. We will be considering the security of these operators. The setting for security is the following. Consider an adversary algorithm A which has access to three oracles, E, F, F^{-1}, where $F \in \mathrm{BC}(\kappa,n)$ and $E : \{0,1\}^n \to \{0,1\}^n$. It computes with them and eventually outputs a bit. This computation is *adaptive.* This means that it makes queries to oracles as it pleases, choosing these queries as a function of answers to previous queries. We represent A's output when interacting with these oracles by $A^{E,F,F^{-1}}$. (Since we will not restrict the computational power of the adversary A, it is without loss of generality deterministic, and hence this output is uniquely defined once A, F, E are fixed.) If the oracles that A interacts with are chosen according to some distribution then A's output will be a random variable over $\{0,1\}$. We let

$$\mathbf{Succ}_A(\kappa, n) \;=\; \Pr\left[\, A^{E,F,F^{-1}} = 1 \;:\; F \leftarrow \mathrm{BC}(\kappa, n) \;;\; E \leftarrow \mathrm{PERM}(n) \,\right]$$

denote the success probability of A in the "ideal world" (called world 2 in the Introduction) where E is a random permutation independent of the cipher F. On the other hand, if $\mathsf{Op} : \mathrm{BC}(\kappa, n) \to \mathrm{BC}(\kappa^*, n^*)$ is an operator then we let

$$\mathbf{Succ}_A(\mathsf{Op}, \kappa, n) \;=\; \Pr\left[\, A^{E,F,F^{-1}} = 1 \;:\; F \leftarrow \mathrm{BC}(\kappa, n) \;;\; k^* \leftarrow \{0,1\}^{\kappa^*} \;;\; E \leftarrow \mathsf{Op}\text{-}F_{k^*} \,\right] .$$

In other words, having selected F, apply the operator to it to get a new cipher $F^* = \mathsf{Op}\text{-}F$. Now, choose at random a permutation E of this cipher, by choosing a key k^* and setting E to $F^*_{k^*}$. (This was called world 1 in the Introduction.) Now let

$$\mathbf{Adv}_A(\mathsf{Op}, \kappa, n) \;=\; \mathbf{Succ}_A(\mathsf{Op}, \kappa, n) \;-\; \mathbf{Succ}_A(\kappa, n) .$$

This is the the advantage of A in breaking the Op induced cipher. To measure the quality of a particular operator Op (eg. Dbl or Trp^2) we want to upper bound the advantage in terms of the resources used by the adversary, meaning the number of queries it makes to its oracles. We call a query to the E oracle an E-query; a query to the F oracle an F query; a query to the F^{-1} oracle an F^{-1} query. Typically the number of E-queries is denoted q, while the sum of the number of F and F^{-1} queries is denoted t. The security of the operator Op is then given by

$$\mathbf{Sec}(\mathsf{Op}, \kappa, n, q, t) \;=\; \max_A \mathbf{Adv}_A(\mathsf{Op}, \kappa, n) ,$$

where the maximum is taken over all adversaries A who make at most q E-queries and at most t F/F^{-1} queries. Thus our goal will be to bound $\mathbf{Sec}(\mathsf{Op}, \kappa, n, q, t)$ in terms of q, t, κ, n for the two ciphers we are investigating, namely $\mathsf{Op} = \mathsf{Dbl}$ and $\mathsf{Op} = \mathsf{Trp}^2$.

We stress that this bound will apply to any adversary. No assumptions are made about the strategy followed by this adversary other than that it is limited to the specified number of queries.

3 Security analysis of the double cipher

In this section our goal will be to determine the security of the doubly iterated ideal cipher. In other words, we want to estimate, as accurately as possible, the value of $\mathbf{Sec}(\mathsf{Dbl}, \kappa, n, q, t)$, as a function of the cipher parameters κ, n and the adversary resource bounds q, t. The following is the main theorem, which provides an upper bound on the security. It says that the advantage of any adversary A attacking the doubly iterated ideal cipher is at most $t^2/2^{2\kappa}$, regardless of the strategy used by this adversary.

Theorem 3.1 *For any $\kappa, n, q, t \geq 1$ it is the case that*

$$\mathbf{Sec}(\mathsf{Dbl}, \kappa, n, q, t) \;\leq\; \frac{t^2}{2^{2\kappa}} .$$

Notice that the bound depends only on the number t of F/F^{-1} queries made by A, and the key length κ of the cipher; it does not depend on the number q of E-queries made by A or the block length n of the cipher. This reflects the reality. In fact our result is essentially tight; more precisely, the bound above is tight up to constant factors as long as q is not too tiny. This is established by Theorem A.2 where we show that an appropriate adaptation of the standard meet in the middle attack enables an adversary to obtain an advantage close to that of the upper bound.

The rest of this section will be devoted to a proof of Theorem 3.1. We fix an adversary A who makes at most q E queries and at most t F/F^{-1} queries. We want to show that $\mathbf{Adv}_A(\mathsf{Dbl}, \kappa, n) \leq t^2/2^{2\kappa}$. We will first introduce some terminology.

3.1 Preliminaries

The probability spaces. We consider two "games." Each consists of running the adversary with its oracles chosen according to some probability space. Probability Space 1 is that of the experiment defining $\mathbf{Succ}_A(\mathsf{Dbl}, \kappa, n)$. Namely, the underlying experiment is:

$$F \leftarrow \mathrm{BC}(\kappa, n) \ ; \ k_1^* \leftarrow \{0,1\}^\kappa \ ; \ k_2^* \leftarrow \{0,1\}^\kappa \ ; \ E \leftarrow F_{k_1^*} \circ F_{k_2^*} \ ,$$

and Game 1 is to just run $A^{E,F,F^{-1}}$ and reply to its oracle queries according to the functions E, F, F^{-1} chosen by the experiment. Now, the experiment defining Probability Space 2 is

$$F \leftarrow \mathrm{BC}(\kappa, n) \ ; \ k_1^* \leftarrow \{0,1\}^\kappa \ ; \ k_2^* \leftarrow \{0,1\}^\kappa \ ; \ E \leftarrow \mathrm{PERM}(n) \ .$$

In Game 2, we just run $A^{E,F,F^{-1}}$ and reply to its oracle queries according to the functions E, F, F^{-1} chosen by the experiment. Notice that in so doing, we completely ignore the two keys k_1^*, k_2^*; the responses to oracle queries do not depend on these at all. Thus, the output of A in Game 2 is exactly that in the experiment defining $\mathbf{Succ}_A(\kappa, n)$. The extra keys we have created will be used only in the analysis. We let $\Pr_1[\cdot]$ denote the probability under Probability Space 1, and $\Pr_2[\cdot]$ that under Probability Space 2.

Quantities involved. Since we are not limiting the computing power of the adversary, we may, without loss of generality, regard it as deterministic. We may also assume it makes exactly q E queries and exactly t F/F^{-1} queries, and that no query is ever repeated. When the oracles E, F, F^{-1} are fixed, the sequence of queries by A and responses by the oracles is determined. We view it as a game in which the adversary and the oracles alternate moves; one query followed by a response is a round, so each round has two moves, the first by the adversary, the second by the oracles. There are $q + t$ rounds. We will be referring to the following quantities:

Mvs = The set $\{0, 1, \ldots, 2(q+t)\}$ whose members will be used to index moves of the game.

OdMvs = The set of odd numbers in Mvs, corresponding to question moves.

EvMvs = The set of even numbers in Mvs, corresponding to reply moves.

It is technically convenient to include 0 in these sets even though there is no 0-th round or move. Furthermore we use the following notation:

q_i : For $i \in$ OdMvs, the query in the i-th move. It is of the form $(x, *)$, $(k, x, *)$, or $(k, *, y)$ which are queries to E, F, and F^{-1}, respectively.

r_i : For $i \in$ EvMvs, the reply in the i-th move. For $i > 0$ it is $(x, E(x))$, $(k, x, F_k(x))$, or $(k, F_k^{-1}(y), y)$, corresponding, respectively, to the query q_{i-1}; for $i = 0$ it is the empty string.

$\mathrm{View}_i(A^{E,F,F^{-1}})$: For $i \in$ Mvs, the view of the adversary after i moves; this is $q_1 r_2 \ldots q_{i-1} r_i$ if $i > 0$ is even; $q_1 r_1 \ldots r_{i-1} q_i$ if i is odd; and the empty string if $i = 0$

$\mathrm{View}(A^{E,F,F^{-1}})$: $\mathrm{View}_{2(q+t)}(A^{E,F,F^{-1}})$.

Note the adversary's output bit is some deterministic function of the last view. We call the keys (k_1^*, k_2^*) chosen in the games the *crucial key pair*. Our analysis will focus on whether or not this key pair is "eliminated" by a current view, and what is its distribution from the point of view of A

if not. So let v_i represent a possible view after i moves of the game. We consider two sets of key pairs, the "seen key pairs" (SKP) and the "remaining key pair" (RKP):

$\mathrm{SKP}(v_i)$: A key pair k_1, k_2 is in $\mathrm{SKP}(v_i)$ if there are two queries q and q' in v_i such that q is an F-query or F^{-1} query with key k_1 (i.e., a query of the form $(k_1, x, *)$ or $(k_1, *, y)$, respectively), and q' is an F-query or F^{-1} query with key k_2 (i.e., a query of the form $(k_2, x, *)$ or $(k_2, *, y)$, respectively).

$\mathrm{RKP}(v_i) = (\{0,1\}^\kappa \times \{0,1\}^\kappa) - \mathrm{SKP}(v_i)$

Note that $\mathrm{SKP}(v_i)$ depends only on the queries in v_i and not on the replies. That is, $\mathrm{SKP}(v_i) = \mathrm{SKP}(v_{i+1})$ for $i \in \mathrm{OdMvs}$. If A knows that $F_{k_2}(x) = y$ and $F_{k_1}(y) = z$ and has also made the E query x then it can with high probability eliminate (k_1, k_2) as a candidate for the crucial key pair. Intuitively, we might think of the key pairs $(k_1, k_2) \in \mathrm{SKP}(v)$ as being "eliminated". (Of course, they might not be eliminated, but we can't be sure, so we count them out.) Thus $\mathrm{RKP}(v_i)$ captures the set of remaining key pairs associated to any view. These are the key pairs (k_1, k_2) so that at least one of them has not been in either an F or an F^{-1} query. Note the key pair is *not* considered "eliminated" if one of its components has been in a F/F^{-1} query: *both* have to have been in such queries to "eliminate" the pair.

The current view v_i contains some number of F or F^{-1} queries on a particular key k. This effectively "opens up" the corresponding spots in row k of the F table, in the sense that in the randomly chosen F table, these entries become known to the adversary. Similarly for E-queries. We let

$\text{F-Qrs}(v_i, k)$ = The set of all y such that there are responses in v_i of the form (k, x, y).

$\text{E-Qrs}(v_i)$ = The set of all y such that there are responses in v_i of the form (x, y).

THE RANDOM VARIABLES. Under the random choice of E, F, F^{-1} made in the probability spaces 1 and 2, the above discussed quantities become random variables. Here are some random variables we will need to refer to explicitly:

Q_i : Takes value q_i, the i-th query, for $i \in \mathrm{OdMvs}$.

R_i : Takes value r_i, the i-th reply, for $i \in \mathrm{EvMvs}$.

T_i : Equals Q_i if i is odd and R_i if i is even.

View_i : Takes value $\mathrm{View}_i(A^{E,F,F^{-1}})$, for $i \in \mathrm{Mvs}$.

View : Takes value $\mathrm{View}(A^{E,F,F^{-1}})$.

$\mathsf{U}_{i,j}$: Equals $\mathsf{T}_i \ldots \mathsf{T}_j$

THE BAD EVENT. We also define a central event:

$\textsc{bad}_i$: For $i \in \mathrm{Mvs}$, event $\textsc{bad}_i$ is said to happen if the crucial key pair $(\mathsf{k}_1^*, \mathsf{k}_2^*)$ is seen, that is, $(\mathsf{k}_1^*, \mathsf{k}_2^*) \in \mathrm{SKP}(\mathsf{View}_i)$.

In other words, the crucial key pair is "eliminated". Whether a particular key pair has been seen only depends on the queries of A and thus $\textsc{bad}_i = \textsc{bad}_{i+1}$ for $i \in \mathrm{OdMvs}$. We let $\textsc{bad}$ be $\textsc{bad}_{2(q+t)}$, meaning it captures whether the bad event happened at the end of the game.

3.2 Proof outline

A very rough cut at the idea of the analysis is that as long as BAD has not happened in probability space 1, the answers coming back to oracle queries there "look random" and so probability space 1 looks like probability space 2. We can then bound the advantage by the probability of the bad event.

This is overly simplistic. It is also incorrect. One should first note that even if the bad event fails to happen in game 1, that game will not look like game 2; there are events that have probability one in the latter and zero in the former. In fact, we need to condition on the bad event not happening in *both* probability spaces.

We will show that the conditional probability of a particular view given that BAD has not occurred is the same in the two games. To show this we will be forced to show something stronger as stated in the lemma below.

Lemma 3.2 Let $i \in \text{Mvs}$ and let v_i be a possible view of the adversary after the i-th move. Then for all $0 \leq s \leq 2(q+t) - i$,

$$\Pr_1\left[\,\mathsf{View}_i = v_i \mid \overline{\textsc{bad}}_{i+s}\,\right] \quad = \quad \Pr_2\left[\,\mathsf{View}_i = v_i \mid \overline{\textsc{bad}}_{i+s}\,\right].$$

The proof of this lemma is postponed until later. Since the final decision of the adversary depends only on its view, the distribution of the adversary's decision is the same in the two games as long as the bad event has not happened. Thus, a corollary to the above lemma is

$$\Pr_1\left[A^{E,F,F^{-1}} = 1 \mid \overline{\textsc{bad}}\right] \;=\; \Pr_2\left[A^{E,F,F^{-1}} = 1 \mid \overline{\textsc{bad}}\right]. \tag{1}$$

Less obvious is that Lemma 3.2 will also be needed to show that the probability of the bad event is the same in both games. To show this we need to prove something a bit stronger: we need to show that the equality holds at any stage. This is stated in the lemma stated below.

Lemma 3.3 For all $i = 0, \ldots, 2(q+t)$,

$$\Pr_1\left[\,\textsc{bad}_i\,\right] \;=\; \Pr_2\left[\,\textsc{bad}_i\,\right]. \tag{2}$$

The proof of this lemma is also postponed until later. Lemmas 3.2 and 3.3 can be used to bound the advantage of the adversary by the probability of the bad event.

Lemma 3.4 $\mathbf{Adv}_A(\mathsf{Dbl}, \kappa, n) \leq \Pr_2[\,\textsc{bad}\,]$.

Proof of Lemma 3.4: The lemma is shown using the following straightforward calculation. We suppress the superscripts of $A^{E,F,F^{-1}}$ for clarity.

$$\begin{aligned}
&\Pr_1[\,A=1\,] - \Pr_2[\,A=1\,] \\
&\quad= \quad \Pr_1[\,A=1 \mid \overline{\textsc{bad}}\,]\cdot\Pr_1[\overline{\textsc{bad}}\,] \;-\; \Pr_2[\,A=1 \mid \overline{\textsc{bad}}\,]\cdot\Pr_2[\overline{\textsc{bad}}\,] \\
&\qquad\quad + \Pr_1[\,A=1 \mid \textsc{bad}\,]\cdot\Pr_1[\,\textsc{bad}\,] \;-\; \Pr_2[\,A=1 \mid \textsc{bad}\,]\cdot\Pr_2[\,\textsc{bad}\,] \\
&\quad= \quad (\Pr_1[\,A=1 \mid \overline{\textsc{bad}}\,] - \Pr_2[\,A=1 \mid \overline{\textsc{bad}}\,])\cdot\Pr_2[\overline{\textsc{bad}}\,]) \\
&\qquad\quad + \;(\Pr_1[\,A=1 \mid \textsc{bad}\,] - \Pr_2[\,A=1 \mid \textsc{bad}\,])\cdot\Pr_2[\,\textsc{bad}\,] \\
&\quad= \;(\Pr_1[\,A=1 \mid \textsc{bad}\,] - \Pr_2[\,A=1 \mid \textsc{bad}\,])\cdot\Pr_2[\,\textsc{bad}\,].
\end{aligned}$$

The second equality follows by Lemma 3.3. The last equality follows by Equation (1). $\blacksquare$

Of course, since the probability of the bad event is the same in both probability spaces we could have bounded the advantage by the probability of the bad event in probability space 1. However, calculating the probability of the bad event is very easy in probability space 2 as can be seen below.

Lemma 3.5 $\Pr_2[\,\text{BAD}\,] \;\leq\; t^2/2^{2\kappa}$.

Proof of Lemma 3.5: This is straightforward, since in Game 2, no information about the keys $(\mathsf{k}_1^*, \mathsf{k}_2^*)$ is given to the adversary. The bad event depends only on the number of F and F^{-1} queries, and in the worst case all the t such queries are made to different keys. Then the chance that k_1^* is in any query is $t/2^\kappa$, and the same, independently, for k_2^*, so the bound holds. ▌

Clearly, Lemmas 3.4 and 3.5 imply Theorem 3.1. This completes the outline of the proof of Theorem 3.1. To complete the proof we must prove Lemmas 3.2 and 3.3.

To do so we will first need a sequence of three lemmas, Lemmas 3.6, 3.7, and 3.8. The last of these will be used in the proof of Lemma 3.2. Lemma 3.6 will again be used to prove Lemma 3.9 on the conditional probability of the crucial key pair. Lemma 3.9 will then be used with Lemma 3.2 to prove Lemma 3.3.

3.3 Distribution of replies in the next round

In Game 2, given the view v_i at any point, the distribution of the answer to the next oracle query is, clearly, uniform, over the remaining range; for example, the answer to an E-query is uniform over $\{0,1\}^n - \text{E-Qrs}(v_i)$.

The first lemma will say this is true for Game 1 too, as long as the bad event does not happen. However, we will need to say this in a strong sense. Namely, fix any key pair that has still not been "eliminated". Conditioned on this being the crucial key pair, as well as on the current view, the distribution of the answer to the next oracle query is still "as it should be," meaning uniform over whatever possibilities remain. Note we must show this for all types of queries: E, F and F^{-1}.

Lemma 3.6 Let $j \in \{1,2\}$ and $i \in \text{OdMvs}$. Let $v_i = q_1 r_2 \ldots q_{i-2} r_{i-1} q_i$ be a possible view of the adversary just before the answer to query q_i is obtained. For any string $r_{i+1} \in \{0,1\}^n$ and all $(k_1, k_2) \in \text{RKP}(v_i \| r_{i+1})$,

$$\Pr_j[\,\mathsf{R}_{i+1} = r_{i+1} \mid (\mathsf{k}_1^*, \mathsf{k}_2^*) = (k_1, k_2) \wedge \mathsf{View}_i = v_i\,] = \begin{cases} \dfrac{1}{2^n - |\text{E-Qrs}(v_i)|} & \text{if } q_i \text{ is an } E\text{-query and } r_{i+1} \notin \text{E-Qrs}(v_i) \\ \dfrac{1}{2^n - |\text{F-Qrs}(k, v_i)|} & \text{if } q_i \text{ is an } F \text{ or } F^{-1} \text{ query with key } k \text{ and } r_{i+1} \notin \text{F-Qrs}(k, v_i) \\ 0 & \text{otherwise.} \end{cases}$$

In particular, the value depends neither on j nor on (k_1, k_2).

Proof of Lemma 3.6: This is clear for Game 2, ie. for $j = 2$. The proof is devoted to showing it also for Game 1, ie. for $j = 1$.

Let $v_{i+1} = v_i r_{i+1}$. We fix a particular key pair $(k_1, k_2) \in \text{RKP}(v_{i+1})$. Assume $\mathsf{View}_i = v_i$, and assume $(\mathsf{k}_1^*, \mathsf{k}_2^*) = (k_1, k_2)$. Note this implies that $\overline{\text{BAD}}_{i+1}$ holds. Now consider three cases.

Case 1: q_i is an E-query.

We want to show that R_{i+1} is equally likely to be any string not yet returned as an answer to an E-query. The danger is that F or F^{-1} queries have been made to at least one of the crucial keys k_1, k_2, and this is giving some information about $F_{k_1} \circ F_{k_2}$ in addition to that from the E queries.

However, this won't happen. This can be seen as follows. We know that $\overline{\text{BAD}}_{i+1}$ holds, which means either k_1 or k_2 has never been in any F or F^{-1} query of the adversary. This means that $F_{k_1} \circ F_{k_2}$, being the composition of two permutations with one random, is random from the point of view of the adversary. (The probability here is over the choice of the cipher F, which assigns a random permutation to each key.) Of course the adversary has partial information about $F_{k_1} \circ F_{k_2}$ in the form of replies to previous E-queries, but this gives no information on the value of any remaining one except that it will not be one already seen.

Case 2: q_i is an F-query.

Let k be the key in the query. If $k \notin \{k_1, k_2\}$ it is clear that the response to the query is randomly distributed over $\{0,1\}^n - \text{F-Qrs}(k, v_i)$ just by the random choice of F in the experiment. So suppose $k = k_l$ where $l \in \{1,2\}$. Now, the danger is that E queries yielded some information about F_k in addition to the queries made directly to key F_k, so the adversary will have some advantage in predicting a new value on F_k.

However, this will not be true. This can be seen as follows. We know $(k_1, k_2) \in \text{RKP}(v_{i+1})$, which means that either k_1 or k_2 has not been in any F or F^{-1} query up to and including the query in q_i. Let $\pi = F_{k_1} \circ F_{k_2}$. As the composition of two permutations, one of which is random, it is random from the point of view of the adversary. Then $F_k = F_{k_l} = \pi \circ F_{k_2}^{-1}$ if $l = 1$ and $F_k = F_{k_l} = F_{k_1}^{-1} \circ \pi$ if $l = 2$. In either case, F_k is the composition of two permutations, one of which is random from the point of view of the adversary, and hence the response to an F query on key k will return a value distributed uniformly over $\{0,1\}^n - \text{F-Qrs}(k, v_i)$.

Case 3: q_i is an F^{-1}-query.

The proof that the response to the query is uniformly distributed over $\{0,1\}^n - \text{F-Qrs}(k, v_i)$ is similar to the case above. ▌

The above lemma shows that for a fixed partial conversation v_i where $i \in \text{OdMvs}$, and fixed pair of keys k_1, k_2 such that $\overline{\text{BAD}}_i$ is true (i.e., $(k_1, k_2) \in \text{RKP}(v_i)$), all the answers r_{i+1} which continue to keep the partial conversations from being "bad" (i.e., $(k_1, k_2) \in \text{RKP}(v_i r_{i+1})$), have the same probability in each probability space. We will use this lemma to prove an extension of this. Namely, for a fixed partial conversation v_i and fixed pair of keys k_1, k_2 such that $\overline{\text{BAD}}_i$ is true, all further move sequences which continue to keep the partial conversations from being "bad" have the same probability in each probability space. We state this formally below.

Lemma 3.7 Let $j \in \{1,2\}$. Let v_i be a possible view of the adversary after move $i \in \text{Mvs}$, and let $1 \leq \ell \leq 2(q+t) - i$. For any possible extension $u_{i+1,i+\ell}$ of v_i by ℓ moves, and for any key pair $(k_1, k_2) \in \text{RKP}(v_i \| u_{i+1,i+\ell})$,

$$\Pr_j [\, \mathsf{U}_{i+1,i+\ell} = u_{i+1,i+\ell} \mid (\mathsf{k}_1^*, \mathsf{k}_2^*) = (k_1, k_2) \wedge \mathsf{View}_i = v_i \,]$$

depends neither on j nor on (k_1, k_2). (That is, it depends only on v_i and $u_{i+1,i+\ell}$.)

Proof of Lemma 3.7: We will prove this by induction on ℓ. The base case is $\ell = 1$. In this case the lemma is clear when $i + 1 = i + \ell$ is odd, because in this case $u_{i+1,i+1}$ is a query, which is a function only of A and v_i. In the case of $i + 1 = i + \ell$ being even, $u_{i+1,i+1}$ is the response R_{i+1}, and we can apply Lemma 3.6.

Now assume that the lemma is true for $\ell = s$. We want to establish it for $\ell = s + 1$. Again, this is trivial if $i + s + 1$ is odd, because then the extension is a query, uniquely determined given $v_i u_{i+1,i+s}$ and A. So assume $i + s + 1$ is even. Let $u_{i+1,i+s+1} = u_{i+1,i+s} r_{i+s+1}$ and $v_{i+s} = v_i u_{i+1,i+s}$. We assume that $(k_1,k_2) \in \mathrm{RKP}(v_i u_{i+1,i+s+1})$. We can write

$$\begin{aligned}
&\Pr_j\,[\,\mathsf{U}_{i+1,i+s+1} = u_{i+1,i+s+1} \mid (\mathsf{k}_1^*,\mathsf{k}_2^*) = (k_1,k_2) \,\wedge\, \mathsf{View}_i = v_i\,] = \\
&\quad \Pr_j\,[\,\mathsf{R}_{i+s+1} = r_{i+s+1} \mid (\mathsf{k}_1^*,\mathsf{k}_2^*) = (k_1,k_2) \,\wedge\, \mathsf{View}_{i+s} = v_{i+s}\,] \\
&\qquad \cdot \Pr_j\,[\,\mathsf{U}_{i+1,i+s} = u_{i+1,i+s} \mid (\mathsf{k}_1^*,\mathsf{k}_2^*) = (k_1,k_2) \,\wedge\, \mathsf{View}_i = v_i\,]\,.
\end{aligned}$$

The first factor depends neither on j nor on (k_1,k_2) by Lemma 3.6. The second factor has the same property by induction. ▌

We now use the above lemma to prove a generalization of Lemma 3.6 which we will need subsequently.

Lemma 3.8 Let $j \in \{1,2\}$ and $i \in \mathrm{OdMvs}$. Let $v_i = q_1 r_2 \ldots q_{i-2} r_{i-1} q_i$ be a possible view of the adversary just before the answer to query q_i is obtained. For any string $r_{i+1} \in \{0,1\}^n$, all $(k_1,k_2) \in \mathrm{RKP}(v_i \| r_{i+1})$, and all $0 \le s \le 2(q+t) - i$,

$$\Pr_j\,[\,\mathsf{R}_{i+1} = r_{i+1} \mid (\mathsf{k}_1^*,\mathsf{k}_2^*) = (k_1,k_2) \,\wedge\, \mathsf{View}_i = v_i \,\wedge\, \overline{\mathrm{BAD}}_{i+s}\,]$$

depends neither on j nor on k_1,k_2. (That is, it depends only on v_i and r_{i+1} and s.)

Proof of Lemma 3.8: First suppose $s = 0$. The conditioning on $\overline{\mathrm{BAD}}_i$ is redundant; this event will be true because $(k_1,k_2) \in \mathrm{RKP}(v_i \| r_{i+1})$. Thus the claim is true from Lemma 3.6.

So assume $s \ge 1$. The probability in the statement of the lemma can be written as

$$\frac{\Pr_j\,[\,\mathsf{R}_{i+1} = r_{i+1} \,\wedge\, \overline{\mathrm{BAD}}_{i+s} \mid (\mathsf{k}_1^*,\mathsf{k}_2^*) = (k_1,k_2) \,\wedge\, \mathsf{View}_i = v_i\,]}{\Pr_j\,[\,\overline{\mathrm{BAD}}_{i+s} \mid (\mathsf{k}_1^*,\mathsf{k}_2^*) = (k_1,k_2) \,\wedge\, \mathsf{View}_i = v_i\,]}.$$

The denominator can be written as

$$\sum \Pr_j\,[\,\mathsf{U}_{i+1,i+s} = u_{i+1,i+s} \mid (\mathsf{k}_1^*,\mathsf{k}_2^*) = (k_1,k_2) \,\wedge\, \mathsf{View}_i = v_i\,]$$

where the sum is over $u_{i+1,i+s}$ such that $(k_1,k_2) \in \mathrm{RKP}(v_i u_{i+1,i+s})$. By Lemma 3.7 each term of this sum has a value that depends neither on j nor on (k_1,k_2). The numerator can be written as

$$\sum \Pr_j\,[\,\mathsf{R}_{i+1}\mathsf{U}_{i+2,i+s} = r_{i+1} u_{i+2,i+s} \mid (\mathsf{k}_1^*,\mathsf{k}_2^*) = (k_1,k_2) \,\wedge\, \mathsf{View}_i = v_i\,]$$

where the sum is over $u_{i+2,i+s}$ such that $(k_1,k_2) \in \mathrm{RKP}(v_i r_{i+1} u_{i+2,i+s})$. By Lemma 3.7 each term of this sum depends neither on j nor on (k_1,k_2). This completes the proof of the lemma. ▌

Proof of Lemma 3.2: The proof will be by induction on $i \in \mathrm{Mvs}$. The base case of the induction is when $i = 0$, and in this case the lemma is trivially true because the view is by definition the empty string. So assume the statement of the lemma up to move i. We will prove it for $i+1$. Fix an arbitrary $s \ge 0$.

First consider the case where $i \in \mathrm{EvMvs}$, meaning the last move in v_i is a reply. Let q_{i+1} be arbitrary. Then:

$$\begin{aligned}
&\Pr_j\,[\,\mathsf{View}_{i+1} = v_i q_{i+1} \mid \overline{\mathrm{BAD}}_{i+1+s}\,] \\
&\quad = \Pr_j\,[\,\mathsf{View}_i = v_i \mid \overline{\mathrm{BAD}}_{i+1+s}\,] \cdot \Pr_j\,[\,\mathsf{Q}_{i+1} = q_{i+1} \mid \mathsf{View}_i = v_i \,\wedge\, \overline{\mathrm{BAD}}_{i+1+s}\,]\;.
\end{aligned}$$

First, look at the first factor. Since $s \geq 0$ by assumption, then $s+1 \geq 0$, and therefore the first term is the same for $j=1$ and 2 by induction. Next look at the second factor. A's query is just dependent on A and on v_i, the view so far. Thus, the probability is the same for both $j=1$ and $j=2$. (And is equal to 0 except possibly for one value of q_{i+1}.) Therefore, the product of the two probabilities is equal for $j=1$ and $j=2$, for all $s \geq 0$.

Next consider the case where $i \in \mathrm{OdMvs}$, meaning the last move in v_i is a query. Let $r_{i+1} \in \{0,1\}^n$ be arbitrary and let $v_{i+1} = v_i r_{i+1}$. Then:

$$\begin{aligned} & \Pr_j\left[\, \mathsf{View}_{i+1} = v_i r_{i+1} \;\middle|\; \overline{\mathrm{BAD}}_{i+1+s} \,\right] \\ & \quad = \; \Pr_j\left[\, \mathsf{View}_i = v_i \;\middle|\; \overline{\mathrm{BAD}}_{i+1+s} \,\right] \cdot \Pr_j\left[\, \mathsf{R}_{i+1} = r_{i+1} \;\middle|\; \mathsf{View}_i = v_i \;\wedge\; \overline{\mathrm{BAD}}_{i+1+s} \,\right] \,. \end{aligned}$$

Consider the first factor. Since $s \geq 0$ by assumption, then $s+1 \geq 0$, and therefore, by induction, the first term is the same for $j=1$ and 2. The second factor is equal to:

$$\sum_{(k_1,k_2)} p_j(k_1,k_2) \cdot q_j(k_1,k_2)$$

where the sum is over all $(k_1,k_2) \in \{0,1\}^\kappa \times \{0,1\}^\kappa$ and we have set

$$\begin{aligned} p_j(k_1,k_2) &= \Pr_j\left[\, \mathsf{R}_{i+1} = r_{i+1} \;\middle|\; (\mathsf{k}_1^*, \mathsf{k}_2^*) = (k_1,k_2) \;\wedge\; \mathsf{View}_i = v_i \;\wedge\; \overline{\mathrm{BAD}}_{i+1+s} \,\right] \\ q_j(k_1,k_2) &= \Pr_j\left[\, (\mathsf{k}_1^*, \mathsf{k}_2^*) = (k_1,k_2) \;\middle|\; \mathsf{View}_i = v_i \;\wedge\; \overline{\mathrm{BAD}}_{i+1+s} \,\right] \end{aligned}$$

We start by examining the first factor, namely $p_j(k_1,k_2)$. By Lemma 3.8, for all $(k_1,k_2) \notin \mathrm{SKP}(v_{i+1})$, this probability is the same for both $j=1$ and 2, and independent of (k_1,k_2). Call this value p. On the other hand for $(k_1,k_2) \in \mathrm{SKP}(v_{i+1})$ we have $p_j(k_1,k_2) = 0$ because of the conditioning on $\overline{\mathrm{BAD}}_{i+1+s}$. Thus the above sum reduces to

$$p \cdot \sum_{(k_1,k_2)} q_j(k_1,k_2)$$

where the sum is over all $(k_1,k_2) \in \mathrm{RKP}(v_{i+1})$. We claim that this range is over all the nonzero values of the probability and thus the sum is equal to 1. To see this, note that $q_j(k_1,k_2)$ is equal to 0 for $(k_1,k_2) \in \mathrm{SKP}(v_{i+1})$. This completes the induction and the proof of Lemma 3.2. ▌

The remaining task is to prove Lemma 3.3 which states that the probability that the bad event occurs is the same in both probability spaces. To do so we will first prove the following lemma about the distribution of keys. The proof of this lemma will use Lemma 3.2 which, recall, states that the probability of a given query and response (which are not bad) for a fixed partial view and a fixed pair of keys (which are not bad) is the same in both probability spaces.

3.4 Equi-probability of unseen keys

A crucial lemma is that in Game 1, as long as the bad event has not happened, if adversary has a particular view, then any "un-eliminated" key pair is equally likely to be the crucial key pair. Without this, it might be that the adversary's chance of hitting the crucial key is better in Game 1 (given the bad event fails) than in Game 2 (given the bad event fails). To simplify notation, for $j \in \{1,2\}$ and v_i let

$$\Pr_{j,v_i}[\,\cdot\,] \;=\; \Pr_j\left[\,\cdot \;\middle|\; \mathsf{View}_i = v_i \;\wedge\; \overline{\mathrm{BAD}}_i \,\right] \,.$$

Lemma 3.9 Let $j \in \{1,2\}$. Let v_i be a possible view of the adversary after move $i \in \text{Mvs}$. Let $(k_1, k_2) \in \text{RKP}(v_i)$. Then

$$\Pr_{j,v_i}\left[\, (\mathsf{k}_1^*, \mathsf{k}_2^*) = (k_1, k_2) \,\right] \;=\; \frac{1}{|\text{RKP}(v_i)|} \;.$$

Proof of Lemma 3.9: This is clear in Game 2, ie. for $j = 2$. The proof is devoted to showing the claim in Game 1, ie. for $j = 1$. The proof will be by induction on the move number $i \in \text{Mvs}$. The base case is $i = 0$. In this case no queries have been made so the adversary has no information about $(\mathsf{k}_1^*, \mathsf{k}_2^*)$, and all possible pairs of keys remain equally likely, so the claim is true. So, assume the lemma statement is true up to move $i \in \text{Mvs}$ where $i < 2(q+t)$. We will prove it for $i+1$.

Let $v_{i+1} = v_i\tau$ where $\tau = q_{i+1}$ is a query if i is even and $\tau = r_{i+1}$ is a reply if i is odd. Assume (k_1, k_2) is some key pair. Consider the quantity

$$\Pr_{1,v_{i+1}}\left[\, (\mathsf{k}_1^*, \mathsf{k}_2^*) = (k_1, k_2) \,\right] . \tag{3}$$

Claim 1: The quantity of Equation (3) is zero if $(k_1, k_2) \notin \text{RKP}(v_{i+1})$.

Proof of Claim 1: This is because $\Pr_{1,v_{i+1}}[\,\cdot\,]$ conditions on $\overline{\text{BAD}}_{i+1}$, meaning we know BAD_{i+1} did not happen. □

Claim 2: Let (k_1, k_2) be any key pair in $\text{RKP}(v_{i+1})$. Then the quantity of Equation (3) has a value that depends only on v_{i+1}, and not on (k_1, k_2).

We will prove Claim 2 below. The two claims together imply that the only possibility is that for all $(k_1, k_2) \in \text{RKP}(v_{i+1})$,

$$\Pr_{1,v_{i+1}}\left[\, (\mathsf{k}_1^*, \mathsf{k}_2^*) = (k_1, k_2) \,\right] \;=\; \frac{1}{|\text{RKP}(v_{i+1})|} \;.$$

Thus the induction would be completed.

Proof of Claim 2: Recall $\mathsf{T}_{i+1} = \mathsf{Q}_{i+1}$ if i is even and $\mathsf{T}_{i+1} = \mathsf{R}_{i+1}$ if i is odd. Expand the quantity of Equation (3):

$$\Pr_{1,v_{i+1}}\left[\, (\mathsf{k}_1^*, \mathsf{k}_2^*) = (k_1, k_2) \,\right] \;=\; \Pr_{1,v_i}\left[\, (\mathsf{k}_1^*, \mathsf{k}_2^*) = (k_1, k_2) \;\middle|\; \mathsf{T}_{i+1} = \tau \;\wedge\; \overline{\text{BAD}}_{i+1} \,\right]$$

and then apply Bayes rule to get:

$$\Pr_{1,v_i}\left[\, \mathsf{T}_{i+1} = \tau \;\wedge\; \overline{\text{BAD}}_{i+1} \;\middle|\; (\mathsf{k}_1^*, \mathsf{k}_2^*) = (k_1, k_2) \,\right] \cdot \frac{\Pr_{1,v_i}\left[\, (\mathsf{k}_1^*, \mathsf{k}_2^*) = (k_1, k_2) \,\right]}{\Pr_{1,v_i}\left[\, \mathsf{T}_{i+1} = \tau \;\wedge\; \overline{\text{BAD}}_{i+1} \,\right]} \;.$$

We want to argue this quantity does not depend on (k_1, k_2). Look first at the fraction. The value of the numerator is given by the induction hypothesis and in particular does not depend on (k_1, k_2). The value of the denominator obviously does not depend on (k_1, k_2) since that quantity appears nowhere in it. Thus, what is left is to show that

$$\Pr_{1,v_i}\left[\, \mathsf{T}_{i+1} = \tau \;\wedge\; \overline{\text{BAD}}_{i+1} \;\middle|\; (\mathsf{k}_1^*, \mathsf{k}_2^*) = (k_1, k_2) \,\right] \tag{4}$$

does not depend on (k_1, k_2).

Observe that the conjunction of the event $\overline{\text{BAD}}_{i+1}$ in the probability of Equation (4) is redundant: since we are conditioning on $(\mathsf{k}_1^*, \mathsf{k}_2^*) = (k_1, k_2)$, and we know that $(k_1, k_2) \in \text{RKP}(v_{i+1})$, the conditioning already tells us that $\overline{\text{BAD}}_{i+1}$ will hold. In other words,

$$\Pr_{1,v_i}\left[\, \mathsf{T}_{i+1} = \tau \;\wedge\; \overline{\text{BAD}}_{i+1} \;\middle|\; (\mathsf{k}_1^*, \mathsf{k}_2^*) = (k_1, k_2) \,\right] \;=\; \Pr_{1,v_i}\left[\, \mathsf{T}_{i+1} = \tau \;\middle|\; (\mathsf{k}_1^*, \mathsf{k}_2^*) = (k_1, k_2) \,\right] \;.$$

Now we consider separately the case where $i+1$ is odd (meaning $\mathsf{T}_{i+1} = \mathsf{Q}_{i+1}$ and $\tau = q_{i+1}$) and the case where $i+1$ is even (meaning $\mathsf{T}_{i+1} = \mathsf{R}_{i+1}$ and $\tau = r_{i+1}$). In the first case, note that the query made is determined only by (A and) the view v_i, so the probability in question does not depend on (k_1, k_2). In the second case, we can apply Lemma 3.6 which gives the value of the above quantity for each $(k_1, k_2) \in \mathrm{RKP}(v_{i+1})$, and, as we see from Lemma 3.6, that value does not depend on (k_1, k_2). This completes the proof of Claim 2. □ ▮

Using the above lemma we can now prove Lemma 3.3 which (recall) states that $\Pr_1[\,\mathrm{BAD}_i\,] = \Pr_2[\,\mathrm{BAD}_i\,]$ for all $i \in \mathrm{Mvs}$.

Proof of Lemma 3.3: The proof is by induction on $i \in \mathrm{Mvs}$. The base case is when $i = 0$. In this case, the current view v of the adversary, in either game, is empty, so that $\mathrm{SKP}(v) = \emptyset$. Thus, both probabilities are zero.

So, assume the lemma statement is true up to move $i \in \mathrm{Mvs}$ where $i < 2(q+t)$. We will prove it for $i+1$, namely we will show that

$$\Pr_1[\,\mathrm{BAD}_{i+1}\,] \;=\; \Pr_2[\,\mathrm{BAD}_{i+1}\,]\,. \tag{5}$$

We first consider the case where $i+1$ is even, meaning the last move in v_i is a query. We have

$$\Pr_j[\,\mathrm{BAD}_{i+i}\,] \;=\; \Pr_j[\,\mathrm{BAD}_i\,] \;+\; \Pr_j[\,\mathrm{BAD}_{i+1} \mid \overline{\mathrm{BAD}_i}\,]\;.$$

The first term is equal for $j = 1$ and 2 by induction, and $\Pr_j[\,\mathrm{BAD}_{i+1} \mid \overline{\mathrm{BAD}_i}\,] = 0$ because $i+1$ is even.

To complete the induction we need to prove Equation (5) for the case where $i+1$ is odd, meaning the last move in v_i is a reply. Let $j \in \{1, 2\}$. We can write

$$\Pr_j[\,\mathrm{BAD}_{i+1}\,] \;=\; \Pr_j[\,\mathrm{BAD}_i\,] \;+\; \Pr_j[\,\mathrm{BAD}_{i+1} \mid \overline{\mathrm{BAD}_i}\,]\;.$$

The first term is independent of j by the induction hypothesis. We will now argue that the second term is also independent of j. By conditioning we can write the second term as

$$\begin{aligned}\Pr_j[\,\mathrm{BAD}_{i+1} \mid \overline{\mathrm{BAD}_i}\,] &= \sum_{v_i \in V_j} \Pr_j[\,\mathrm{BAD}_{i+1} \mid \overline{\mathrm{BAD}_i} \wedge \mathsf{View}_i = v_i\,] \cdot \Pr_j[\,\mathsf{View}_i = v_i \mid \overline{\mathrm{BAD}_i}\,] \\ &= \sum_{v_i \in V_j} \underbrace{\underbrace{\Pr_{j,v_i}[\,\mathrm{BAD}_{i+1}\,]}_{\text{first term}} \cdot \underbrace{\Pr_j[\,\mathsf{View}_i = v_i \mid \overline{\mathrm{BAD}_i}\,]}_{\text{second term}}}_{\text{product term associated to } v_i}\,,\end{aligned}$$

where $V_j = \{\, v_i \,:\, \Pr_j[\,\mathsf{View}_i = v_i \mid \overline{\mathrm{BAD}_i}\,] > 0\,\}$ is the set of possible views after move i in Game j.

Let us first observe that $V_1 = V_2$, namely the set of views v_i for which the second term of the "product term associated to v_i" is positive is the same in both games. This is true by Lemma 3.2, which tells us that $\Pr_j[\,\mathsf{View}_i = v_i \mid \overline{\mathrm{BAD}_i}\,]$ does not depend on j and hence in particular the values of v for which it is zero are the same for $j = 1$ and $j = 2$.

Now let us set $V = V_1 = V_2$ and compare the sums, term by term, in the cases $j = 1$ and $j = 2$. Fix a particular string $v_i \in V$ and focus on the "product term associated to v_i." The second term in it is independent of j by Lemma 3.2. We will show the same is true for the first term, which will complete the proof. (One needs to be a little careful. The first term is not well defined for just any v, only for $v_i \in V_j$. That's why it was important, first, to restrict attention to these v_i values, and, second, to make sure that $V_1 = V_2$, since otherwise we would not be sure that we have shown equality for every term in the two sums.)

So the remaining task is to consider $\Pr_j[\,\mathrm{BAD}_{i+1} \mid \overline{\mathrm{BAD}_i} \wedge \mathsf{View}_i = v_i\,]$ for $v_i \in V$ and show it does not depend on j. First note that $\mathrm{RKP}(v_i) \neq \emptyset$, because, $\mathrm{RKP}(v_i) = \emptyset$ would imply $\Pr_j[\,\mathsf{View}_i = v_i \mid \overline{\mathrm{BAD}_i}\,] = 0$, and we have assumed the last to not be true.

Since the view v_i and the adversary are fixed, the next query q_{i+1} is uniquely determined. Let

$$\mathrm{NKP}(v_i, q_{i+1}) = \mathrm{RKP}(v_i) - \mathrm{RKP}(v_i \| q_{i+1})$$

be the set of "new key pairs" that are "seen" by the $(i+1)$-th query. (This set is empty if the latter is an E-query. It is also empty if it is an F or F^{-1} query with key with which A has already queried. If it is an F or F^{-1} query with key k with which A has not queried, then the set consists of pairs (k, k') and (k', k) where k' is any other key with which A has queried F or F^{-1}.) We claim that

$$\Pr_j\left[\, \mathrm{BAD}_{i+1} \;\middle|\; \overline{\mathrm{BAD}_i} \wedge \mathsf{View}_i = v_i \,\right] \;=\; \frac{|\mathrm{NKP}(v_i, q_{i+1})|}{|\mathrm{RKP}(v_i)|}\,, \tag{6}$$

for both $j = 1$ and $j = 2$. Note the fraction is well defined, in that the denominator is not zero, because $\mathrm{RKP}(v_i)$ is non-empty.

Equation (6) follows from Lemma 3.9. This tells us that from the point of view of the adversary, all remaining key pairs remain equally likely, in either game. ▌

References

[1] W. Aiello, M. Bellare, G. Di Crescenzo and R. Venkatesan, "Security amplification by composition: The case of doubly-iterated, ideal ciphers", Preliminary version of this paper, *Advances in Cryptology – Crypto* 98 *Proceedings*, Lecture Notes in Computer Science Vol. 1462, H. Krawczyk ed., Springer-Verlag, 1998.

[2] M. Bellare, J. Kilian and P. Rogaway, "The security of cipher block chaining", *Advances in Cryptology – Crypto* 94 *Proceedings*, Lecture Notes in Computer Science Vol. 839, Y. Desmedt ed., Springer-Verlag, 1994.

[3] E. Biham and A. Shamir, "Differential cryptanalysis of DES-like cryptosystems," *J. of Cryptology*, Vol. 4, No. 1, pp. 3–72, 1991.

[4] E. Biham and A. Shamir, "Differential cryptanalysis of the Full 16-round DES," *Advances in Cryptology – Crypto* 92 *Proceedings*, Lecture Notes in Computer Science Vol. 740, E. Brickell ed., Springer-Verlag, 1992.

[5] W. Diffie and M. Hellman, "Exhaustive cryptanalysis of the NBS data encryption standard," *Computer*, Vol. 10, No. 6, pp. 74–84, June 1977.

[6] S. Even and O. Goldreich, "On the power of cascade ciphers," *ACM Transactions on Computer Systems*, Vol. 3, No. 2, May 1985, pp. 108–116.

[7] S. Even and Y. Mansour, "A construction of a cipher from a single pseudorandom permutation," *Advances in Cryptology – ASIACRYPT* 91 *Proceedings*, Lecture Notes in Computer Science Vol. 739, H. Imai, R. Rivest and T. Matsumoto ed., Springer-Verlag, 1991.

[8] J. Kilian and P. Rogaway, "How to protect DES against exhaustive key search", *Advances in Cryptology – Crypto* 96 *Proceedings*, Lecture Notes in Computer Science Vol. 1109, N. Koblitz ed., Springer-Verlag, 1996.

[9] M. Matsui, "Linear cryptanalysis method for DES cipher," *Advances in Cryptology – Eurocrypt* 93 *Proceedings*, Lecture Notes in Computer Science Vol. 765, T. Helleseth ed., Springer-Verlag, 1993.

[10] U. Maurer and J. Massey, "Cascade ciphers: The importance of being first," *Journal of Cryptology*, Vol. 6, No. 1, 1993, pp. 55–61.

[11] R. Merkle, "Secrecy, authentication, and public key systems," UMI Research Press, Ann Arbor, Michigan, 1979.

[12] R. Merkle, and M. Hellman, “On the security of multiple encryption”, Communications of the ACM, vol. 24, n. 7, pp. 465–467, July 1981.

[13] C. Shannon, “Communication theory of secrecy systems,” *Bell Systems Technical Journal*, Vol. 28, No. 4, 1949, pp. 656–715.

[14] P. Van Oorschot and M. Wiener, “Improving meet in the middle attacks by orders of magnitude,” *Advances in Cryptology – Crypto* 96 *Proceedings*, Lecture Notes in Computer Science Vol. 1109, N. Koblitz ed., Springer-Verlag, 1996.

[15] P. Van Oorschot and M. Wiener. “A known plaintext attack on Two-Key Triple Encryption,” *Advances in Cryptology – Eurocrypt* 90 *Proceedings*, Lecture Notes in Computer Science Vol. 473, I. Damgård ed., Springer-Verlag, 1990.

A Best attack: Meet in the middle

In this section we will show the following:

Lemma A.1 For any $\kappa, n \geq 1$, any $1 \leq s \leq q \leq 2^{n-1}$, and any $t \geq 2s$, there is an adversary A such that

$$\mathbf{Adv}_A(\mathsf{Dbl}, \kappa, n) \;\geq\; \frac{t^2}{4s^2} \cdot \left(\frac{1}{2^{2\kappa}} - \frac{1}{2^{s(n-1)}} \right) \; .$$

We can now optimize the value of s and obtain the following theorem which says that the bound of Theorem 3.1 is essentially tight:

Theorem A.2 *For any $\kappa, n \geq 1$, let $s = \lceil (2\kappa+1)/(n-1) \rceil$. Then for any $t \geq 2s$ and $s \leq q \leq 2^{n-1}$ it is the case that*

$$\mathbf{Sec}(\mathsf{Dbl}, \kappa, n, q, t) \;\geq\; \frac{1}{8s^2} \frac{t^2}{2^{2\kappa}} \; .$$

Proof: The choice of s guarantees that $2^{2\kappa+1} \leq 2^{s(n-1)}$. This means that

$$\frac{1}{2^{2\kappa}} - \frac{1}{2^{s(n-1)}} \;\geq\; \frac{1}{2} \frac{1}{2^{2\kappa}} \; .$$

Now apply Lemma A.1. ■

Notice that for typical block cipher parameters κ, n, the value of s is very small. For example, for the DES parameters $\kappa = 56$ and $n = 64$ we have $s = \lceil 113/63 \rceil = 2$. Thus the above lower bound of Theorem A.2 is in practice close to the upper bound of Theorem 3.1.

Proof of Lemma A.1: The proof is by presenting an adversary A who achieves the claimed advantage. The adversary A plays a version of the meet-in-the-middle attack, but we need to adapt it slightly and then analyze it in our framework. It is convenient to let $[N] = \{1, 2, \ldots, N\}$ for any integer $N \geq 1$. The adversary proceeds as follows:

```
For $j = 1, \ldots, s$ do
    Let $x_j \in \{0,1\}^n$ be the $j$-th string in lexicographic order
    Compute $y_j = E(x_j)$
Endfor
Choose two disjoint sets $K_1 = \{ k_{1,i} : i \in [t/2s] \}$ and $K_2 = \{ k_{2,i} : i \in [t/2s] \}$ of $\kappa$-bit keys,
each set being of size $t/2s$. (These might be chosen at random, but not necessarily).
For $i = 1, \ldots, t/2s$ do
```

```
        For $j = 1, \ldots, s$ do Compute $u_{i,j} = F(k_{1,i}, x_j)$ and $v_{i,j} = F^{-1}(k_{2,i}, y_j)$ Endfor
        Let $u_i = (u_{i,1}, \ldots, u_{i,s})$ and $v_i = (v_{i,1}, \ldots, v_{i,s})$
    Endfor
    Let $C = \{ (a,b) \in [t/2s] \times [t/2s] \, : \, u_a = v_b \}$
    If $C \neq \emptyset$ then return 1 else return 0
```

We now analyze this attack. The first claim is that the cost is as claimed, meaning A makes at most q E-queries and at most t F/F^{-1} queries. The first is true because $s \leq q$ by assumption. The second is true because the number of calls to F/F^{-1} above is $2[(t/2s)s] = t$. We now want to lower bound

$$\mathbf{Adv}_A(\mathsf{Dbl}, \kappa, n) \;=\; \mathbf{Succ}_A(\mathsf{Dbl}, \kappa, n) - \mathbf{Succ}_A(\kappa, n) \; .$$

We will lower bound the first term and upper bound the second. Let $\Pr[\,\cdot\,]$ denote the probability in the experiment underlying the definition of $\mathbf{Succ}_A(\mathsf{Dbl}, \kappa, n)$, and let $k_1^* k_2^*$ denote the randomly chosen 2κ bit key in this experiment. Observe that if $k_1^* \in K_1$ and $k_2^* \in K_2$ then C is definitely non-empty. So

$$\mathbf{Succ}_A(\mathsf{Dbl}, \kappa, n) \;\geq\; \Pr[\, k_1^* \in K_1 \text{ and } k_2^* \in K_2 \,] \;=\; \left(\frac{t/2s}{2^{\kappa}}\right)^2 \;=\; \frac{1}{4s^2}\frac{t^2}{2^{2\kappa}} \; .$$

Now let $\Pr[\,\cdot\,]$ denote the probability in the experiment underlying the definition of $\mathbf{Succ}_A(\kappa, n)$, and observe that

$$\mathbf{Succ}_A(\kappa, n) \;=\; \Pr[\, C \neq \emptyset \,] \; .$$

For a fixed $a, b \in [t/2s]$ we have

$$\Pr[\, u_a = v_b \,] \;=\; \prod_{j=1}^{s} \frac{1}{N - j - 1} \;\leq\; \left(\frac{1}{2^{n-1}}\right)^s \; .$$

The last inequality here is by the assumption that $s \leq 2^{n-1}$. By the union bound we have

$$\Pr[\, C \neq \emptyset \,] \;\leq\; \frac{t^2}{4s^2} \cdot \frac{1}{2^{s(n-1)}} \; .$$

This completes the proof. ▌

B Analysis of the two-key triple cipher

The two-key triple cipher (namely, the construction underlying two-key triple DES) was defined in Section 2. The same upper bound on the advantage of any adversary A attacking this cipher can be shown as for the double cipher:

Theorem B.1 *For any $\kappa, n, q, t \geq 1$ it is the case that*

$$\mathbf{Sec}(\mathsf{Trp}^2, \kappa, n, q, t) \;\leq\; \frac{t^2}{2^{2\kappa}} \; .$$

Unlike the case of the double cipher, however, this bound is not tight, and we believe it can be improved by a better analysis.

The proof of the theorem is obtained by adapting the proof of Theorem 3.1. We will use essentially the same setup; we start by giving some new definitions and then continue by showing the necessary modifications for the proof in Section 3 so that it works also in the case of operator Trp^2. large

Games, setup, random variables and event $\textsc{bad}_i$. The experiment underlying Game 2 is the same as for the proof of Theorem 3.1. The experiment underlying Game 1 is now the following:

$$F \leftarrow \mathrm{BC}(\kappa, n) \; ; \; k_1^* \leftarrow \{0,1\}^\kappa \; ; \; k_2^* \leftarrow \{0,1\}^\kappa \; ; \; E \leftarrow F_{k_1^*} \circ F_{k_2^*}^{-1} \circ F_{k_1^*} \; ,$$

and the game is to just run $A^{E,F,F^{-1}}$ and reply to its oracle queries according to the functions E, F, F^{-1} chosen by the experiment. The setup and the random variables are defined exactly in the same way as before, with the understanding that when we mention E-queries, in Game 1, we refer to a query to the cipher $F_{k_1^*} \circ F_{k_2^*}^{-1} \circ F_{k_1^*}$. Event $\textsc{bad}_i$ is formally defined exactly as before.

Analysis. We observe that almost all lemmas in our previous analysis do not significantly depend on the construction we are analyzing. More precisely, we see that all lemmas but Lemma 3.6 require no modification for both the statement and the proof to hold also in the case of the construction Trp^2. So it remains to modify Lemma 3.6 so that it works also in the current case. Recall that such lemma is trying to show that the distribution of the next reply is independent of which game the adversary is in, and also of a fixed un-eliminated key pair. However, this can in fact be seen to still be true, because we are still looking at compositions of random permutations with one unknown. We omit the details.

Lower bound. The standard meet in the middle attack for triple DES [5, 12] can be put and analyzed in our model analogously to the way we did it above for the double cipher. The analysis indicates that our upper bound for the two-key triple cipher is tight (up to a constant factor) when $q \approx t$, but not tight in general. We do not include the details of this analysis.