Cryptogr. Commun. (2018) 10:959–1012
https://doi.org/10.1007/s12095-018-0294-5

On stream ciphers with provable beyond-the-birthday-bound security against time-memory-data tradeoff attacks

Matthias Hamann[1] · **Matthias Krause**[1]

Received: 3 August 2017 / Accepted: 12 March 2018 / Published online: 8 May 2018

Abstract Most stream ciphers are vulnerable against generic time-memory-data tradeoff (TMD-TO) attacks, which reduce their effective key length to the birthday bound $n/2$, where n denotes the inner state length of the underlying keystream generator. This implies the necessity of a comparatively large inner state length for practical stream ciphers (e.g., $n = 288$ and $n = 160$ for the eSTREAM portfolio members Trivium and Grain v1, respectively). In this paper, we propose and analyze the LIZARD-construction, a new way to build stream ciphers. We prove a tight $2n/3$ bound on its security against TMD-TO key recovery attacks, where the security lower bound refers to chosen-IV attacks. The security against TMD-TO distinguishing attacks remains at the birthday-bound level $n/2$. The lower bound refers to a random oracle model which allows to derive formal security results w.r.t. generic TMD-TO attacks. While similar frameworks have already been widely used for analyzing the security of block cipher, MAC, and hash function constructions, to the best of our knowledge this is the first time that such a model is considered in the context of stream ciphers. The security analysis presented in this paper is also of immediate practical relevance as, with the stream cipher LIZARD, a first instantiation of our new design principle (which we hence named LIZARD-construction) was introduced at FSE 2017. LIZARD has an inner state length of only 121 bits and surpasses Grain v1, the most hardware efficient member of the eSTREAM portfolio, in important metrics for lightweight ciphers such as chip area and power consumption.

This article is part of the Topical Collection on *Special Issue on Statistics in Design and Analysis of Symmetric Ciphers*

✉ Matthias Hamann
hamann@uni-mannheim.de

Matthias Krause
krause@uni-mannheim.de

[1] Lehrstuhl für Theoretische Informatik, Universität Mannheim, 68131 Mannheim, Germany

Keywords Stream ciphers · Time-Memory-Data tradeoff attacks · Random oracle models · Provable security

Mathematics Subject Classification (2010) 94A60

1 Introduction

Stream ciphers are, besides block ciphers, the most popular family of modern symmetric encryption algorithms. They are intended for encrypting, in an online manner, plaintext bitstreams X which have to pass an insecure channel. The encryption is performed via bitwise addition of a keystream S to X, which depends on a secret symmetric key k and a public initial value IV. The legal recipient, who also knows k, decrypts the encrypted bitstream $Y = X \oplus S$ by generating S and computing $X = Y \oplus S$. An important use case for stream ciphers is the encryption of over-the-air communication for mobile devices, which implies that lightweight aspects play a dominant role in the design of stream ciphers.

In our framework, we suppose that the communication between legal users is organized in *sessions*, where in the first phase of each session, the secret session key k is generated by executing a key establishment protocol. This session key generation phase will not be considered in this paper. In practice, a session can, e.g., be a phone call, where at the beginning of the call, a key establishment protocol between the mobile phone and the nearest base station is performed.

Following [8], each stream cipher is associated with a well-defined set of inner states and its keystream generation process can be divided into the following two phases: (A) The **key and IV setup phase**, where an initial state is derived from the secret session key k and an initial value IV, and (B) the **keystream generation phase**, in which the keystream is generated based on the initial state derived in phase (A).

In this paper, we consider keystream generator-based stream ciphers, for which the main algorithmic component for performing phases (A) and (B) is a so-called keystream generator (KSG). KSGs are clock-controlled devices which can be formally specified by finite automata, defined by an inner state length n, the set of inner states $\{0,1\}^n$, a state update function $\pi : \{0,1\}^n \longrightarrow \{0,1\}^n$, and an output function $outbit : \{0,1\}^n \longrightarrow \{0,1\}$. Starting from an initial state q_0, in each clock cycle $t \geq 0$, the KSG produces an output bit $z_t = outbit(q_t)$ and changes the inner state according to $q_{t+1} = \pi(q_t)$. The keystream $S(q_0)$ corresponding to the initial state q_0 is defined by concatenating all the output bits $z_0 z_1 z_2 \cdots$.

The key and IV setup phase (A) of a KSG-based stream cipher is typically performed by a KSG-based **state initialization algorithm**, which computes the initial state q_{init} from the session key k and the initial value IV. It always contains the following two subphases:

(A.1) The **loading phase** defines how the session key k and the initial value IV are loaded into the inner state registers and results in a load state $q_{\text{load}} = q_{\text{load}}(k, IV)$.

(A.2) The **mixing phase** runs an appropriate KSG-based mixing algorithm $MIX : \{0,1\}^n \longrightarrow \{0,1\}^n$ on q_{load} and yields a state $q_{\text{mixed}} = MIX(q_{\text{load}})$.

The aim of the mixing phase (A.2) is to ensure that each initial state bit depends on many session key bits and IV bits and that this dependency, expressed as a multivariate $GF(2)$-polynomial over the session key bits and IV bits, has large degree. In many cases, an essential part of the mixing algorithm consists in running the KSG a certain number of

times without producing keystream bits. Moreover, for many ciphers (Grain, Trivium, E_0, A5/1 etc.) it holds that $q_{\text{init}} = q_{\text{mixed}}$.

In this paper, with the LIZARD-construction, we propose a state initialization algorithm of type

$$q_{\text{init}} = MIX\,(q_{\text{load}}) \oplus k, \tag{1}$$

where $q_{\text{load}} = k \oplus IV$, (see Fig. 1) and show that this state initialization algorithm (together with a certain mode of operation) guarantees a beyond-the-birthday-bound security against time-memory-data tradeoff (TMD-TO) attacks.

As in [8, 9], and many other papers, we consider the keystream generation phase (B) of KSG-based stream ciphers as being defined by the output block function $OUTBLOCK : \{0,1\}^n \longrightarrow \{0,1\}^n$, which assigns each inner state $q \in \{0,1\}^n$ to the first n keystream bits produced on q. We give now the exact definition of $OUTBLOCK$:

Definition 1 We consider a KSG with output bit function $outbit : \{0,1\}^n \longrightarrow \{0,1\}$ and state transition function $\pi : \{0,1\}^n \longrightarrow \{0,1\}^n$. For all $q \in \{0,1\}^n$, let

$$OUTBLOCK(q) = (\tilde{z}_0, \cdots, \tilde{z}_{n-1}),$$

where for all r, $0 \le r \le n-1$,

$$\tilde{z}_r = outbit(\pi^r(q)).$$

Here, π^r means applying the state transition function r times, so $\pi^r(q)$ denotes the state obtained from q after clocking the cipher r times.

Note that the keystream $S(q_{\text{init}}(k, IV)) = (z_0, z_1, \cdots)$ generated on a key-IV pair (k, IV) can now be expressed as follows (see also Fig. 2). For each n-bit subblock $(z_r, \cdots, z_{r+n-1})$, it holds

$$(z_r, \cdots, z_{r+n-1}) = OUTBLOCK\left(\pi^r\,(q_{\text{init}})\right).$$

One can distinguish the following two operation modes of stream ciphers.

In the *one-stream mode*, the key and IV setup phase (A) is performed only once at the beginning of the session and produces an initial state $q_{\text{init}} = q_{\text{init}}(k, IV)$. The corresponding keystream $S = S(q_{\text{init}})$ is used for the whole session. Note that due to their extremely large limits (e.g., 2^{64} bits for Trivium) on the amount of keystream generated under a single key-IV pair, Trivium [12] and Grain [25] can be considered to work in one-stream mode.

In contrast to this, in the *packet mode*, the communication and encryption process during a session is divided into packet steps $i = 1, 2, \cdots$, where in each packet step, a piece of message of a certain maximal packet length R is encrypted and sent. Corresponding to this, the keystream of a session is the concatenation of the keystream packets $S^1 S^2 S^3 \cdots$, where for all $i \ge 1$, S^i denotes the keystream packet generated in packet step i. The stream cipher is equipped with an additional mechanism which generates for each packet step i an initial value IV^i. Each packet step i starts with performing the key and IV setup phase

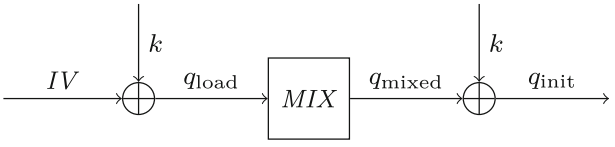

Fig. 1 The key and IV setup phase (A) of the LIZARD-construction. The XOR symbol denotes the addition of the corresponding n-bit vectors over $GF(2)$

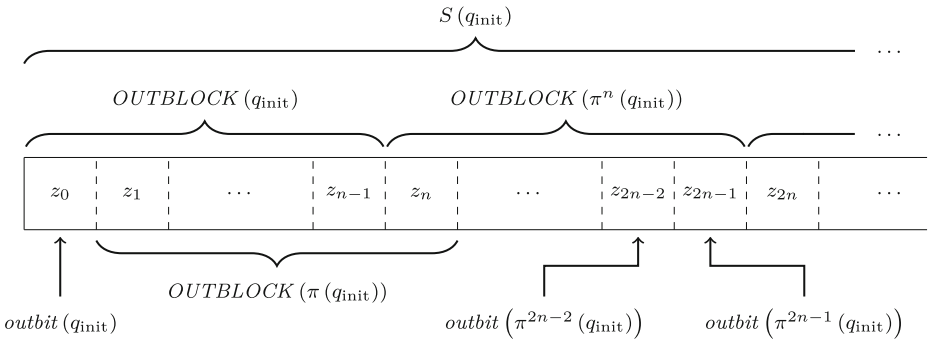

Fig. 2 The keystream generation phase (B) in terms of our model

(A), which computes a packet initial state $q^i_{\text{init}} = q^i_{\text{init}}(k, IV^i)$, followed by the generation of the keystream packet S^i, which is defined to be the prefix of length R of $S(q^i_{\text{init}})$. As in many communication scenarios data streams are encrypted and transmitted packet-wise (Bluetooth, WLAN, cellular networks etc.), it seems natural to run a stream cipher in packet mode. Typical examples are the GSM cipher A5/1 and the Bluetooth cipher E_0 (see [22] for more practical examples of stream ciphers running in packet mode and more information about the practical relevance of such ciphers). Clearly, Trivium and Grain could also be used in packet mode but, in contrast to, e.g., LIZARD [22], their design is not specifically optimized for such scenarios. More precisely, all current *small-state stream ciphers* (i.e., stream ciphers targeting beyond-the-birthday-bound security; see [23] for an overview) impose some additional restriction about their application context in order to reach this improved security level. For example, Sprout-like small-state stream ciphers (see below) assume that it is feasible to continuously access the secret key not only during initialization but also during keystream generation. LIZARD, on the other hand, uses the secret key only during initialization but assumes that per IV, at most 2^{18} keystream bits need to be generated. As explained in [22], this limit fits well for many prominent communication scenarios and allows, due to LIZARD's beyond-the-birthday-bound security, for a more efficient hardware implementation (w.r.t. important cost factors such as chip area and power consumptions) than, e.g., the general-purpose stream ciphers Trivium and Grain. This is particularly relevant in the context of ultra-constrained devices like low-cost radio-frequency identification (RFID) tags, where virtually every hardware gate matters and corresponding restrictions often hinder cryptographic schemes from being used in practice (see, e.g., [3]).

During the last decades, many stream ciphers have been suggested and many different techniques for cryptanalyzing stream ciphers have been developed (correlation attacks, fast correlation attacks, guess-and-verify attacks, BDD attacks, cube attacks etc.). Attacks on stream ciphers typically suppose that the attacker knows a piece S' of keystream which was generated under a secret session key k and a set of known or actively chosen initial values. Standard goals of attacks are to distinguish S' from a truly random bitstream, to recover the inner states responsible for S', to predict a new keystream packet on the basis of S', or to recover the secret session key.

In this paper, we focus on time-memory-data tradeoff (TMD-TO) attacks, which are for many stream ciphers the most powerful known attacks. TMD-TO attacks have a generic nature in the sense that they access the security-relevant components *MIX* and *OUTBLOCK*

only in a black-box manner. This implies that from the attacker's point of view, these components are *ideally designed* in the sense of [19]. Hence, the aim of TMD-TO attacks is to analyze the way how the components *MIX* and *OUTBLOCK* interact in computing the keystream from the secret session key k and an initial value IV and to check if this way opens the door for nontrivial attacks.

In consequence, such TMD-TO attacks can usually be formulated for variable inner state length n. Correspondingly, we express upper and lower security bounds for stream cipher constructions against TMD-TO attacks in an asymptotic manner. For instance, we say that a stream cipher construction has, for some number a, $0 \leq a \leq 1$, the security level $a \cdot n$ w.r.t. TMD-TO attacks if there is a TMD-TO attack of cost behavior $O(2^{a \cdot n})$ with significant success probability and, for all $\alpha < a$, all TMD-TO attacks of cost behavior $O(2^{\alpha \cdot n})$ have only negligibly small success probability. We will discuss the cost behavior of TMD-TO attacks in more detail at the beginning of Section 3.

The vulnerability against generic TMD-TO attacks such as those of Babbage [5] or Biryukov and Shamir [9] represents an inherent weakness of KSG-based stream ciphers. This vulnerability implies that for KSG-based stream ciphers working in one-stream mode, the effective key length is bounded by $\frac{n}{2}$, where n denotes the inner state length of the underlying KSG. As a consequence, modern practical stream ciphers have comparatively large inner state lengths (e.g., 288 bits for the eSTREAM portfolio member Trivium [12] or 160 bits for the eSTREAM portfolio member Grain v1 [25]).

In this paper, we propose a construction principle to design KSG-based stream ciphers with a provable beyond-the-birthday-bound security of $\frac{2}{3}n$ against generic TMD-TO key recovery attacks: taking a stream cipher with a state initialization algorithm as described in Relation (1) and using it in packet mode. We give this construction principle the name LIZARD-construction, as it underlies the stream cipher LIZARD [22] introduced at FSE 2017.

The LIZARD-construction can be motivated as follows. Babbage's TMD-TO attack [5] implies that if a KSG-based stream cipher runs in one-stream mode, then it is possible to predict the keystream of the whole session with a TMD-TO attack of cost behavior $O(2^{n/2})$ (see Theorem 1 in Section 3). Moreover, if the state initialization algorithm is efficiently invertible (as it is the case, e.g., with Trivium, Grain v1, A5/1), then this attack even yields the secret session key.

The question is if KSG-based stream ciphers running in packet mode can have beyond-the-birthday-bound resistance against TMD-TO attacks. The following observation shows that packet mode alone is not enough for reaching this goal. Many stream ciphers (e.g., A5/1, E_0, Trivium, Grain v1) employ a state initialization algorithm of type

$$q_{\text{init}} = MIX\,(q_{\text{load}}) \tag{2}$$

with $q_{\text{load}} = q_{\text{load}}(k, IV)$, instead of

$$q_{\text{init}} = MIX\,(q_{\text{load}}) \oplus k$$

with $q_{\text{load}} = q_{\text{load}}(k, IV) = k \oplus IV$, as used by the LIZARD-construction (cf. Relation (1)). We show in Theorem 3 that even if a stream cipher runs in packet mode and even if the state initialization algorithm is not efficiently invertible (as, e.g., that of E_0), a state initialization algorithm of the type in Relation (2) provides only a security level of $\frac{n}{2}$ w.r.t. session key recovery attacks.

In contrast to this, we show a tight $\frac{2}{3}n$ bound on the security of the LIZARD-construction against TMD-TO attacks. More precisely, in Theorem 4 we describe a TMD-TO session key recovery attack of TMD-cost $\tilde{O}(2^{(2/3)n})$ against the LIZARD-construction, which is

based on the Slidex attack of Dunkelman, Keller, and Shamir [16] against the one-key Even-Mansour cipher [18].

The main contribution of this paper is to show that for the LIZARD-construction, this $\frac{2}{3}n$ security bound is sharp.

The proof of the matching security lower bound result is done, in the spirit of [19], in a random oracle model corresponding to the components *MIX* and *OUTBLOCK* of the LIZARD-construction. We prove an information-theoretic lower bound on the security of the LIZARD-construction against generic chosen-IV attackers, who have black-box access to the component primitives *MIX* and *OUTBLOCK*, and to the stream cipher construction itself. Due to their generic nature, all known TMD-TO attacks against stream ciphers can be formulated as attacks in this model in a straightforward way.

The proof of our security lower bound follows the typical structure of similar information-theoretic proofs in the context of iterated Even-Mansour ciphers (see, e.g., [2, 11, 13, 14, 18, 26]). In particular, it is inspired by the (much shorter) security lower bound proof in [18]. As in [18], the lower bound against key recovery attacks follows from a lower bound against the weaker type of attack in which the goal of the attacker is to predict a new keystream packet, and which we call *packet prediction attack* in the following. The rough idea consists in proving that if Eve poses significantly less than at most $2^{(2/3)n}$ component and construction queries, then, with high probability, the entropy of the secret session key will still be at least $n - 1$. This immediately implies an exponentially small success probability for recovering the session key and we will show that this is also the case for predicting a correct new packet.

Note that the $\frac{2}{3}n$ security bound for the LIZARD-construction cannot hold against distinguishing attacks. We show in Theorem 2 that there is a TMD-TO distinguishing attack of TMD-cost $\tilde{O}(2^{(1/2)n})$ against any KSG-based stream cipher working in packet mode if the packet length exceeds the inner state length n.

To the best of our knowledge, this is the first time that a formal random oracle model for the security of stream ciphers against generic TMD-TO attacks is considered. So far, similar models were used, e.g., for analyzing the security of operation modes of key-alternating block cipher constructions (see the framework of iterated Even-Mansour ciphers), or of cryptographic hash functions, or of MAC algorithms, but not for stream ciphers.

Note that in [8], another way of formally analyzing the security of stream cipher constructions was proposed, namely in the complexity-theoretic framework of pseudorandom number generators and pseudorandom function generators.

In 2015, Armknecht and Mikhalev suggested with Sprout [4] another construction method for stream ciphers with beyond-the-birthday-bound security against TMD-TO attacks. In Sprout, the symmetric secret key is not only accessed during the state initialization but also continuously used as part of the state update during the subsequent keystream generation phase. The hope here was to obtain stream ciphers with the maximal possible resistance against TMD-TO attacks.

Although Sprout was broken soon after publication via non-generic attacks (see, e.g., [6, 17, 27, 33]), it has raised interest in the design principle and a number of related ciphers have been suggested since, including Fruit [20] and Plantlet [30]. Very recently, however, it has been shown in [23] that this whole class of Sprout-like ciphers is susceptible to generic TMD-TO distinguishing attacks (with complexity about $2^{n/2}$) and, hence, does not meet the original expectation of providing *full* TMD-TO security. This emphasizes the importance of *provable* resistance against TMD-TO attacks as a design criterion for new stream cipher constructions.

As already mentioned, the LIZARD-construction, offering provable beyond-the-birthday-bound security against TMD-TO key recovery attacks, has inspired the design of the recently published lightweight stream cipher LIZARD [22]. LIZARD works in packet mode with a packet length of $R \leq 2^{18}$ bits, has a state initialization algorithm of type (1), and an inner state length of 121 bits. The design features of LIZARD presented in [22] show that the LIZARD-construction allows for practical instantiations which are competitive w.r.t. important hardware metrics for lightweight devices such as chip area and power consumption. The maximum packet length of $R \leq 2^{18}$ bits was chosen by the designers *as large as necessary, but as small as possible*. More precisely, in [22], an overview over the most prominent packet-based (encrypted) communication scenarios (such as cellular networks, Bluetooth, WLAN, HTTPS etc.) is given. While Bluetooth packets contain at most 2^{12} bits for the so-called basic rate and in WLAN connections, at most 2^{16} bits are encrypted under the same key/IV pair, the current TLS version 1.2 [15] (underlying HTTPS) requires to encrypt at most $2^{17} + 2^{13} < 2^{18}$ bits per packet, hence the limit of $R \leq 2^{18}$ for LIZARD. Based on the principle *as small as possible*, the limit was not chosen larger in order to keep the impact of distinguishing attacks to a tolerable level. For an in-depth discussion of this, we refer the reader to Section 4.2 of [22], where a concrete example based on the maximum packet length of 2^{18} for LIZARD is given.

Recently, interesting cryptanalytic results for LIZARD have been published in [7, 29, 31]. Please note, however, that none of these papers violates the security claims made in the LIZARD specification and, hence, breaks the ciphers. In particular, the analysis provided in [7, 29, 31] does not indicate any weakness of the general LIZARD-construction design principle. To avoid potential misconceptions, it is especially important to realize that the algorithms in [7] for computing key-IV pairs which produce identical initial states (and, hence, identical keystreams) do not lead to actual attacks. This is due to the fact that in these algorithms, the attacker chooses the keys himself. This way, he is able to invert Phase 3 (the second key addition) of LIZARD's state initialization and generate **some** key-IV pair that leads to a given initial state. However, the algorithms in [7] do not provide any indication on how to efficiently find the **actual** secret key if the attacker is only given an initial state together with the IV that was used to generate it (under this secret key).

Structure of the paper In Section 2, we discuss security-relevant properties of the stream cipher components *MIX* and *OUTBLOCK* and describe the structure of some existing stream ciphers in terms of these components. In Section 3, we describe three TMD-TO attacks against KSG-based stream ciphers, including one against the LIZARD-construction. In Section 4, we introduce the random oracle model for stream ciphers. Section 5 contains the corresponding lower bound results. Section 6 concludes the paper by summarizing our results and showing some directions of further research.

2 More on stream ciphers

In this section, we first discuss some concrete security-relevant issues of the components *MIX* and *OUTBLOCK* of KSG-based stream ciphers. After this, we describe the state initialization algorithms of some concrete stream ciphers in terms of our formalism.

Let us fix a KSG of inner state length n, and let π and *outbit* define its state transition and output bit function, respectively. Observe first that the corresponding function *OUTBLOCK* is π-iterative in the following sense:

Definition 2 A function $F : \{0,1\}^n \longrightarrow \{0,1\}^n$ is called π-iterative if for all inputs $y \in \{0,1\}^n$ it holds that the suffix of length $n-1$ of $F(y)$ equals the prefix of length $n-1$ of $F(\pi(y))$.

Observe next that *OUTBLOCK* should be *preimage resistant* in the sense that it is infeasible to compute, for given $z \in \{0,1\}^n$, a value $y \in \{0,1\}^n$ fulfilling $OUTBLOCK(y) = z$. Otherwise, it would be feasible to predict, on the basis of the first n keystream bits of a packet, all remaining keystream bits of this packet.

Concerning *MIX*, observe that standard efficiency and security assumptions on KSGs imply that *MIX* should be an efficiently computable function which behaves like a random function with respect to important properties such as correlation immunity, algebraic degree, or resistance against conditional differential cryptanalysis. Moreover, in most stream ciphers, *MIX* is bijective and can be inverted efficiently (see the examples below).

These properties will be reflected in our security analysis in the way that *OUTBLOCK* is assumed to be a randomly chosen π-iterative function and that *MIX* is assumed to be a randomly chosen permutation which can be inverted efficiently.

Note here that for many practical KSG-based stream ciphers, the component *OUTBLOCK* may deviate from behaving like a random function in the following respect. For efficiency reasons, the state transition function π and the output bit function *outbit* are oftenly defined to be of low degree. According to Definition 1, this implies that a small number of output bits of *OUTBLOCK* (namely the leftmost ones) can have a lower degree than expected for a random function, and consequently that the stream cipher can have a lowered sampling resistance in the sense of [9] and [10]. As described there, this could be used for reducing the cost of certain TMD-TO attacks in a non-generic manner. This allowed, e.g., a very efficient attack on the A5/1-cipher (see [10]). As for the asymptotic behavior of TMD-TO attacks the cost reductions of techniques like BSW-sampling (see [10]) are rather negligible, we do not consider this effect in our security model. However, the possibility of a lowered sampling resistance has to be considered in the design of concrete stream ciphers.

We conclude this section by describing the state initialization algorithm of some relevant stream ciphers and expressing them by our formalism.

Trivium The stream cipher Trivium has an inner state of length 288 bits, distributed over three nonlinear feedback shift registers (NFSRs) of lengths 93 bits, 84 bits, and 111 bits. The state update function consists of the corresponding three feedback functions, which in each case are quadratic and take their inputs from two of the three NFSRs. The linear output function XORs six inner state bits, two from each NFSR. The loading state $q_{\text{load}}(k, IV) = (k||IV||CONST)$ is defined to be the concatenation of the 80-bit session key k, the 80-bit IV IV and a predefined 128-bit constant $CONST$. In the mixing phase, the KSG is clocked $4 \cdot 288$ times without producing output (see [12] for more details). Consequently,

$$q_{\text{init}} = q_{\text{mixed}} = MIX(k||IV||CONST). \tag{3}$$

Grain v1 The stream cipher Grain v1 has an inner state of length 160 bits, distributed over one NFSR and one linear feedback shift register (LFSR), both of length 80 bits. The state update function consists of the corresponding two feedback functions, where the NFSR feedback function depends also on one of the LFSR bits. The output function produces one keystream bit per clock cycle and depends nonlinearly on five LFSR bits and one NFSR bit and linearly on further seven NFSR bits. The loading state $q_{\text{load}}(k, IV, CONST)$ is defined to be the concatenation of the 80-bit session key k, a 64-bit IV IV and a predefined 16-bit

constant *CONST*. In the mixing phase, the Grain-KSG is clocked 160 times, where, in each clock cycle, the corresponding output keystream bit is XORed to the result of each of the two feedback functions (see [25] for more details). Consequently, we have again

$$q_{\text{init}} = q_{\text{mixed}} = MIX(k||IV||CONST). \tag{4}$$

E_0 (Bluetooth) Bluetooth works in packet mode with a packet length $R \leq 2745$ bits.[1] The inner state length of E_0 is 132 bits, distributed over four LFSRs of overall length 128 bits and an extra finite state machine of inner state length four bits. The state update function updates all LFSRs separately. The state transition of the 4-bit finite state machine additionally depends on four bits from the LFSRs. The output function XORs the output bits of the LFSRs with the nonlinear 1-bit output of the finite state machine.

For each packet i, the initial value IV^i is composed of the 48-bit Bluetooth address of the master device, 26 bits of the master's clock (to which both devices are synchronized) at the time of the first transmission slot of this packet, and two 3-bit constants. The E_0 cipher loads k and IV^i stepwise to the register cells of the KSG, resulting in the inner state $q^i_{\text{load}} = L(k) \oplus \tilde{L}(IV^i)$, where $L, \tilde{L}$ denote linear functions defined by the four linear feedback shift registers of the E_0-KSG. Subsequently, the generator is clocked 56 times and the output is discarded. Based on the resulting inner state of the E_0-KSG, 128 keystream bits are then computed without outputting them. Instead, they are copied into the LFSR register cells, overwriting the old inner state (see [32] for more details). Consequently, the state initialization algorithm of E_0 can be modeled as

$$q^i_{\text{init}} = q^i_{\text{mixed}} = MIX\left(L\left(k\right) \oplus \tilde{L}\left(IV^i\right)\right). \tag{5}$$

For a more precise description of the rather involved structure and key/IV loading of E_0, we refer the reader to, e.g., Section 3.1 of [21]. Note that E_0 has to be considered broken as, e.g., in [28], a key recovery attack which only requires the first 24 bits of $2^{23.8}$ frames and 2^{38} computations is presented.

LIZARD The definition of LIZARD is inspired by the Grain family [24] of stream ciphers. It employs 120-bit keys and 64-bit IVs, targeting a security level of 80 bits against key recovery and 60 bits against distinguishing. In opposite to Grain, LIZARD is designed for working in packet mode with a packet length of 2^{18} bits. It has an inner state length of 121 bits, distributed over two NFSRs of lengths 90 bits and 31 bits, and a nonlinear output function (see [22] for more details). The prominent innovation of LIZARD is that the state initialization algorithm is designed according to the scheme in Relation (1), i.e.,

$$q^i_{\text{init}} = q^i_{\text{mixed}} \oplus k = MIX\left(k \oplus IV^i\right) \oplus k. \tag{6}$$

Obviously, the above numbers w.r.t. key size, IV size, and inner state size do not directly match Relation 6. In particular, for efficiency reasons, in LIZARD the IV is only of size 64 bits and corresponds to the 120-bit string $IV^i_0, \ldots, IV^i_{63}, 0, \ldots, 0$ in terms of the general LIZARD-construction. In Section 4, we explain why the general security promise of the LIZARD-construction still carries over to its concrete instantiation LIZARD. For further details, e.g., relating the fact that in LIZARD, the state size is one bit larger than the key size, we refer the reader to the Section 3.5 of [22].

[1] More precisely, if the so-called *basic rate* is used, Bluetooth data packets contain at most 2745 bits of payload, which are encrypted using the E_0 cipher.

In Table 1, an overview of the above stream ciphers in terms of our model is presented. For E_0 (used in Bluetooth), we give the IV size as 74 bits based on the contained 48-bit Bluetooth address of the master device and the 26 bits of the master's clock at the time of the first transmission slot of the respective packet (as described above). Note, however, that an attacker targeting a specific connection between two devices will only have to deal with the changing 26 bits of the master's clock as part of the IV, as, in this situation, the 48-bit Bluetooth address of the master device is constant. Furthermore, for E_0, we do not list any security claims in the column *Security Level* of Table 1, as the Bluetooth specification [32] does not provide any such specific commitments. However, despite the fact that already non-generic attacks (such as [28]) exists for E_0, through Theorem 3 in Section 3 we emphasize that, in general, such a construction can only reach a security level of $n/2$, where $n = 132$ bits in the case of E_0.

3 Time-Memory-Data tradeoff attacks

In this section, we first make some general remarks on TMD-TO attacks against KSG-based stream ciphers and then describe four such attacks. As already mentioned, we consider KSG-based stream ciphers to be defined by the inner state length n, the key length KL, the IV length IVL, and the algorithmic components $LOAD : \{0,1\}^{KL} \times \{0,1\}^{IVL} \longrightarrow \{0,1\}^n$, $MIX : \{0,1\}^n \longrightarrow \{0,1\}^n$, and $OUTBLOCK : \{0,1\}^n \longrightarrow \{0,1\}^n$. A TMD-TO attacker is supposed to know a certain amount of keystream (usually called *data*), which has its origin in one session, i.e., it was generated under one secret session key k. TMD-TO attacks are generic in the sense that they assume that the security-relevant components *MIX* and *OUTBLOCK* are ideally designed, which is reflected by the assumption that the attacker has only black-box access to *MIX* and *OUTBLOCK*. One distinguishes passive TMD-TO attacks, in which the attacker knows a certain amount of keystream generated w.r.t. to one or several IVs known to her, and active TMD-TO attacks, in which the attacker gets keystream packets generated w.r.t. to IVs of her choice. Note that the TMD-TO attacks discussed in this section are passive, while our security lower bound in Section 5 refers to active TMD-TO attacks.

Depending on the goal of the attack, we distinguish four types of TMD-TO attacks, namely 1.) session key recovery attacks, 2.) inner state recovery attacks, 3.) packet prediction attacks, and 4.) distinguishing attacks. The goal of attacks of type 1.) is to recover the secret session key under which the given data was generated, where the goal of attacks

Table 1 An overview of the some prominent stream ciphers in terms of our model. *Key Rec.* stands for key recovery and *Dist.* for distinguishing

Cipher	Size State/Key/IV [bits]	Security Level Key Rec. / Dist. [bits]	State Initialization	Broken?
Trivium	288/80/80	80/80	$q_{\text{init}} = q_{\text{mixed}}$	No
Grain v1	160/80/64	80/80	$q_{\text{init}} = q_{\text{mixed}}$	No
E_0	132/128/74	?/?	$q^i_{\text{init}} = q^i_{\text{mixed}}$	Yes
LIZARD	121/120/64	80/60	$q^i_{\text{init}} = q^i_{\text{mixed}} \oplus k$	No

of type 2.) is to recover at least one inner state q for which the corresponding piece of keystream $OUTBLOCK(q)$ belongs to the given data. Attacks of type 3.) refer to ciphers working in packet mode. Here the goal is to compute a pair $(IV, z) \in \{0,1\}^{IVL} \times \{0,1\}^n$, where IV denotes a new initial value which is not associated with the given data, and $z = OUTBLOCK(q_{\text{init}}(IV, k))$ is the prefix of length n of the packet generated on initial value IV under the given secret session key k. The goal of a distinguishing attack against a given cipher is to decide if the given data is generated in a pseudorandom scenario or in a random scenario. In the pseudorandom scenario, the data is generated by the cipher on the basis of a randomly chosen secret session key, while in the random scenario the data is generated by a random bit generator. Besides the running time, a relevant cost parameter of distinguishing attacks is the advantage, which is defined to be the difference of two conditional probabilities, namely that the attacker outputs *pseudorandom* in the pseudorandom scenario, and that the attacker outputs *pseudorandom* in the random scenario. TMD-TO attacks are usually randomized algorithms. The success probability of an attack is the probability of the event that the attacker reaches her goal, where the underlying probability space is defined by the random choice of the secret information and the internal randomization of the attacker.

Following [9] and [10], TMD-TO attacks are considered to be divided into a (key-independent) precomputation phase, in which, based on the components of the cipher, some search data structure is constructed, and the online phase, in which the now given data (e.g., passively or actively obtained keystream) is used for reaching the attack goal based on the previously computed search data structure. Corresponding to this, TMD-TO attacks are associated with the cost metrics P (time of the precomputation phase), M (memory), T (time of the online phase), and D (data), which are scalable in the sense that a smaller amount of one resource (e.g., data) can be compensated by a larger amount of other resources (e.g., memory and time). This implies that the cost behavior of TMD-TO attacks is usually expressed as a so-called tradeoff curve of type $f(P, M, T, D) = B$, where f is some real function depending on P, M, T, D and B is some real number. The interpretation is that if one invests precomputation time P, online time T, memory M, and data D such that $f(P, M, T, D) = B$, then the attack reaches its goal with significant success probability. For instance, the tradeoff curve of Babbage's attack is $T \cdot D = 2^n$.

In our context, we concentrate on the overall time $P + T$ needed by both phases. We understand under the minimum TMD-cost of a TMD-TO attack the minimal number C for which the relation $f(P, M, T, D) = B$ can be fulfilled under the condition that $P + T$ does not exceed C. Note that $P + T \leq C$ implies $D \leq C$ and $M \leq C$. This is due to the fact that the relations $P + T \geq M$ and $T \geq D$ always hold. Note that the tradeoff curve $T \cdot D = 2^n$ of Babbage's attack implies a minimum TMD-cost of $C = 2^{n/2}$.

The typical basic operations of TMD-TO attacks are operations over n-bit blocks (inner states or n-bit blocks of keystream bits) such as comparing them or searching for them in a data base, where n denotes the inner state length of the underlying cipher. This is the reason why usually the $\tilde{O}(\cdot)$-notation is used for expressing the asymptotic cost behavior of TMD-TO attack algorithms. Running time $\tilde{O}(f(n))$ means that the attack performs $O(f(n))$ basic operations over n-bit blocks.

At some places, we use the relation $\lim_{N\to\infty}(1 - \frac{1}{N})^N = e^{-1}$. This relation implies that the probability that at least one of N independent trials with success probability $\frac{1}{N}$ is successful, is greater than $\frac{1}{4}$ if $N > 2$ and around $1 - e^{-1}$ if N is large enough.

Let us start with the the traditional inner state recovery attack of Babbage in [5]:

Theorem 1 *We consider a KSG-based stream cipher of inner state length n working in packet mode or in one-stream mode. Suppose that attacker Eve knows D different keystream sequences (which may be from different packets) of length at least n. Then Eve can compute the initial state of at least one packet in time $\tilde{O}(2^n/D)$ with success probability greater than $\frac{1}{4}$ if $n > 1$ (and close to $1 - e^{-1}$ if n is large enough), which implies a tradeoff curve $T \cdot D = 2^n$ and minimum TMD-cost $2^{n/2}$.*

Proof of Theorem 1 Eve generates $T = 2^n/D$ times a pair $(y, OUTBLOCK(y))$ for randomly and independently chosen inner states $y \in \{0,1\}^n$. As $T \cdot D = 2^n$, with probability around $1 - e^{-1}$ there is some pair $(y, OUTBLOCK(y))$ such that y equals one of the inner states behind the D keystream subsequences of length n known to Eve. As the state transition function π is efficiently invertible, this allows to efficiently compute the secret initial state q^i_{init} of the corresponding packet, respectively of the only initial state q_{init} if the cipher runs in one-stream mode. Setting $T = D = 2^{n/2}$ implies an attack of time and data $\tilde{O}(2^{n/2})$. □

In one-stream mode, the attack in Theorem 1 discovers the only initial state q_{init} and, thus, the whole keystream of this session. In packet mode, the initial state of at least one packet is discovered and, thus, the whole packet can be computed.

In the following, we show that the attack in Theorem 1 can be converted into a distinguishing attack of the same minimal TMD-cost against ciphers working in packet mode if the packet length is greater than the inner state length.

Theorem 2 *We consider a KSG-based stream cipher of inner state length n working in packet mode with packet length $R = n + 1$. Suppose that Eve knows data consisting of D different keystream packets. Then Eve can distinguish the pseudorandom scenario from the random scenario in time $\tilde{O}(2^n/D)$ with advantage around $\sqrt{e^{-1}} - e^{-1}$.*

Proof of Theorem 2 For each inner state $y \in \{0,1\}^n$, we denote by $Z(y) \in \{0,1\}^{n+1}$ the sequence of the first $n+1$ keystream bits generated on initial state y. Moreover, let $\mathcal{D}^*$ denote the set of all those packets $Z \in \{0,1\}^{n+1}$ contained in the data for which there is some inner state $y \in \{0,1\}^n$ with $Z(y) = Z$. We know that in the pseudorandom case all D packets contained in the data belong to $\mathcal{D}^*$, while in the random case the probability that $|\mathcal{D}^*|$ deviates significantly from $D/2$ is negligibly small.

Eve now generates at most $T = 2^n/D$ times a pair $(y, Z(y))$ for randomly and independently chosen inner states $y \in \{0,1\}^n$ and stops with output `pseudorandom` if she gets a collision, i.e., if she generated some y for which $Z(y)$ coincides with one of the D packets contained in the data. If after $T = 2^n/D$ rounds no collision happened, she stops and outputs `random`.

By the same arguments as in the proof of Theorem 1 it follows that in the pseudorandom case the probability that Eve outputs `pseudorandom` is around $1 - e^{-1}$. In the random case, the probability that Eve outputs `pseudorandom` is around $1 - (1 - \frac{D/2}{2^n})^{2^n/D} \approx 1 - \sqrt{e^{-1}}$. This implies an advantage of

$$\left|\left(1 - e^{-1}\right) - \left(1 - \sqrt{e^{-1}}\right)\right| = \sqrt{e^{-1}} - e^{-1}.$$

□

Theorem 2 shows that with respect to distinguishing attacks, KSG-based stream ciphers, even if they run in packet mode, cannot reach beyond-the-birthday-bound security. But what about the security against attacks with more complex goals such as packet prediction attacks and session key recovery attacks? Theorem 1 shows that for achieving a higher resistance than $\tilde{O}(2^{n/2})$ it must be hard to compute the secret session key from a given packet initial state. Note that this is **not true** for Trivium, Grain, and A5/1, as for all these ciphers *MIX* is efficiently invertible and its result is taken directly as the initial state. The following theorem shows that even if the mixing algorithm is presumably preimage resistant (as in the case of the Bluetooth cipher E_0), the security against session key recovery attacks will be only $n/2$ if the state initialization algorithm implies that the packet initial states are equal to the packet mixing states (as it is the case for E_0).

Theorem 3 *We consider a KSG-based stream cipher working in packet mode with a state initialization for which the packet initial states are equal to the packet mixing states. Then, with probability around* $1 - e^{-1}$*, Eve can compute the secret session key in time* $\tilde{O}(2^n/D)$ *if she knows* D *different* n*-bit keystream packet prefixes (which implies a tradeoff curve* $T \cdot D = 2^n$ *and minimum TMD-cost* $2^{n/2}$*).*

Proof of Theorem 3 By the assumption it holds that

$$q^i_{\text{init}} = MIX\left(q_{\text{load}}\left(k, IV^i\right)\right)$$

for all packets i.

Eve generates $T = 2^n/D$ times a pair $(q, OUTBLOCK(MIX(q)))$ for randomly and independently chosen inner states $q \in \{0,1\}^n$. As $T \cdot D = 2^n$, with probability around $1 - e^{-1}$, for some q, $MIX(q)$ equals the initial state of one of the D prefixes known to Eve, which implies $q = q_{\text{load}}(k, IV^i)$. This allows to compute k from q as IV^i is public. Here we assumed that k can be efficiently computed from $q_{\text{load}}(k, IV^i)$ and IV^i, which is true for all KSG-based stream ciphers which are known to us. □

Theorem 3 shows that for achieving beyond-the-birthday-bound security against generic TMD-TO attacks, the state initialization algorithm has to provide

$$q^i_{\text{init}} \neq MIX(q_{\text{load}}(k, IV^i)).$$

The main result of this paper is to show that beyond-the-birthday-bound security against packet prediction and session key recovery TMD-TO attacks can be achieved if the packet initial states are computed according to the LIZARD-construction (see Relation (1)), i.e., as

$$q^i_{\text{init}} = MIX\left(q^i_{\text{load}}\right) \oplus k,$$

where $q^i_{\text{load}} = k \oplus IV^i$. But before we go into the details of the corresponding lower bound proof, the next theorem will first show a respective upper bound, namely a session key recovery TMD-TO attack with minimal TMD-cost $\tilde{O}(2^{(2/3)n})$ against the LIZARD-construction.

Theorem 4 *We consider a KSG-based stream cipher working in packet mode for which the inner state length* n *equals the initial value length and the session key length, and we assume that for all* $i \geq 1$ *the packet initial states* q^i_{init} *are generated according to Relation (1), see above. Suppose that Eve knows a set of* D *packet prefixes of length* n*, i.e., a set of pairs*

$\{(IV^i, PREFIX(IV^i)); i \in I^*\}$ *for a set* $I^* \subseteq \mathbb{N}$, *where* $D = |I^*| \geq 2^{n/2}$ *and, for each* $i \in I^*$,

$$PREFIX\left(IV^i\right) = OUTBLOCK\left(MIX\left(IV^i \oplus k\right) \oplus k\right).$$

Then, with constant positive probability, Eve can compute the secret session key in time $\tilde{O}(D + \frac{2^{2n}}{D^2})$, *which implies TMD-cost of* $\tilde{O}(2^{(2/3)n})$.

Proof of Theorem 4 Our attack uses the idea of the Slidex attack of Dunkelman, Keller, and Shamir [16] against the one-key Even-Mansour cipher. We describe the attack but only sketch the analysis of the success probability. For an exact specification of the success probabilities we refer to [16]. Note that our attack does not have a preprocessing phase.

Let Q^* denote the set of all initial states corresponding to the indices in I^*, i.e.,

$$Q^* = \left\{MIX\left(IV^i \oplus k\right) \oplus k; i \in I^*\right\}.$$

Observe that before starting the attack, Q^* is unknown to Eve. The attack now consists of two phases.

In the first phase, Eve generates D times a pair $(q, OUTBLOCK(q))$ for randomly and independently chosen inner states $q \in \{0,1\}^n$. Whenever q falls into Q^*, which happens with probability $\frac{D}{2^n}$, Eve sees a collision of $OUTBLOCK(q)$ with $PREFIX(IV^i)$ for some $i \in I^*$ and it holds $MIX(IV^i \oplus k) \oplus k = q$. Consequently, after the first phase, a standard Chernoff bound argument yields that Eve knows with constant positive probability a set of pairs $\{(IV^i, q_{\text{init}}(IV^i)); i \in I^{**}\}$ for some $I^{**} \subseteq I^*$ with $|I^{**}| \geq \frac{D^2}{2^n}$.

In a second phase, Eve generates $\frac{2^n}{D^2/2^n} = \frac{2^{2n}}{D^2}$ times a pair $(u, MIX(u))$ for randomly and independently chosen inner states $u \in \{0,1\}^n$. Eve stops with u if

$$u \oplus IV^i = MIX(u) \oplus q_{\text{init}}(IV^i) \tag{7}$$

for some $i \in I^{**}$ and publishes the hypothesis that $u \oplus IV^i$ equals the secret session key k.

In [16], it is shown that the event that Relation (7) holds implies the event $k = u \oplus IV^i$ with positive constant probability if *MIX* is supposed to behave like a random permutation. Note further that, as $\frac{2^{2n}}{D^2} \cdot |I^{**}| \geq \frac{2^{2n}}{D^2} \cdot \frac{D^2}{2^n} = 2^n$, the event that Relation (7) is fulfilled during the second phase happens with probability around $1 - e^{-1}$. □

4 A random oracle model for the LIZARD-Construction

In this section, we introduce a random oracle model which allows to prove information-theoretic lower bounds on the security of KSG-based stream ciphers against TMD-TO attacks. In this model we suppose that the components *MIX* and *OUTBLOCK* of a given KSG-based stream cipher are ideally designed and that an attacker has only black-box oracle access to these components. In this sense, random oracle models allow to analyze the power of generic attacks, which do not exploit possible cryptographic weaknesses of the components *MIX* and *OUTBLOCK*, but concentrate on the way how these components interact in computing the keystream from the secret session key and public initial values. Note that all TMD-TO attacks presented in Section 3 have this generic nature and can be formulated in our random oracle model in a straightforward way.

We start with a formal definition which gives an exact specification of the notion LIZARD-construction.

Definition 3 A KSG-based stream cipher is called to be designed according to the LIZARD-construction if it fulfills the following criteria:

(L1) The construction refers to three auxiliary parameters n, π, R, where n denotes the inner state length, $\pi : \{0,1\}^n \longrightarrow \{0,1\}^n$ denotes the state transition function of the underlying KSG, and R denotes the packet length. It is required that $R \geq n$, that π is bijective, and that for all inner states $q \in \{0,1\}^n$ the period of the sequence $(\pi^r(q))_{r=0}^{\infty}$ is greater than R.

(L2) The construction refers to a set of secret session keys and a set of public initial values, which are both defined to be $\{0,1\}^n$. Moreover, the construction refers to an *ideal* bijective state mixing function $MIX : \{0,1\}^n \longrightarrow \{0,1\}^n$, which behaves like a random permutation over $\{0,1\}^n$, and an *ideal* π-iterative output block function $OUTBLOCK : \{0,1\}^n \longrightarrow \{0,1\}^n$, which behaves like a random π-iterative function over $\{0,1\}^n$ (see Definition 2 for the definition of π-iterativeness). *MIX* and *OUTBLOCK* are considered to be the main components of the cipher.

(L3) The construction consists in the following rules how to generate from a secret key $k \in \{0,1\}^n$ and a packet initial value $IV \in \{0,1\}^n$ the packet initial state $q_{\text{init}}(k, IV) \in \{0,1\}^n$ and the corresponding keystream packet $PACKET(k, IV) = PACKET(q_{\text{init}}(k, IV)) \in \{0,1\}^R$. Let

$$q_{\text{init}}(k, IV) = MIX(k \oplus IV) \oplus k \tag{8}$$

and

$$PACKET(q_{\text{init}}(k, IV)) = (z_0, z_1, \cdots, z_{R-1}),$$

where for all r, $0 \leq r \leq R - n$, it holds

$$(z_r, z_{r+1}, \cdots, z_{r+n-1}) = OUTBLOCK\left(\pi^r(q_{\text{init}}(k, IV))\right). \tag{9}$$

Note that Relation (9) corresponds to the usual keystream generation definition (see Section 2, especially Definition 1). Note further that the stream cipher LIZARD, as defined in [22], differs from the design features of the LIZARD-construction in some minor points, which do not harm our security bounds. For instance, in contrast to condition (L2), the IV length of LIZARD is smaller than the inner state length. Observe that a smaller IV length means that an attacker can use only a smaller set of possible IVs as inputs of oracle queries in his attack. Thus, a smaller IV length lowers the power of a chosen-IV attacker, i.e., our security lower bounds also hold for a modified LIZARD-construction of IV length smaller than n.

We model the security of the LIZARD-construction against generic TMD-TO attacks by the adversary Eve's success probability to win the following *packet prediction game* with a limited number of oracle queries against Alice, who holds a secret session key k. Eve has black-box access to the ideal components *MIX* and *OUTBLOCK*, and is allowed to ask for keystream packets $PACKET(k, IV^i)$ generated w.r.t. the secret session key k held by Alice and IVs IV^i of Eve's choice. Eve wins the game if, after asking a certain number of oracle queries, she is able to predict the keystream packet w.r.t. to a new IV, which has not been asked before.

From now on, we denote the component functions *MIX* and *OUTBLOCK* by P and F, respectively, the construction function *PACKET* by E, and initial values by $x \in \{0,1\}^n$ (respectively x', x^* etc.).

Definition 4 (The Packet Prediction Game)

(i) The game depends on the global parameters n, π, R, which satisfy the rules in Definition 3, and a parameter M, which bounds the number of oracle queries. The game is divided into a query phase and a prediction phase.

(ii) At the beginning, Alice chooses randomly and w.r.t. the uniform distribution a secret triple $\omega = (k_\omega, P_\omega, F_\omega)$, where

- $k_\omega \in \{0,1\}^n$ denotes the secret session key,
- $P_\omega : \{0,1\}^n \longrightarrow \{0,1\}^n$ denotes a random permutation (corresponding to *MIX*),
- $F_\omega : \{0,1\}^n \longrightarrow \{0,1\}^n$ denotes a random π-iterative function (corresponding to *OUTBLOCK*).

We denote by Ω the corresponding probability space of all such triples together with the uniform distribution.

(iii) The adversary Eve is a randomized oracle algorithm of potentially unbounded computational power, who is allowed to pose *component oracle queries* of type $P(u) =?$, or $P^{-1}(v) =?$, or $F(y) =?$ for inputs $u, v \in \{0,1\}^n$ and $y \in \{0,1\}^n$, which are correctly answered by Alice by $P_\omega(u)$, $(P_\omega)^{-1}(v)$, or $F_\omega(y)$, respectively.

(iv) Moreover, Eve is allowed to pose *construction queries* of the form $E(x) =?$, where $x \in \{0,1\}^n$, which will be answered by Alice with the keystream packet $E_\omega(x)$ corresponding to the initial state

$$y := P_\omega(x \oplus k_\omega) \oplus k_\omega$$

induced by the session key k_ω and the initial value x (see Relation (8)). Note that this keystream packet $E_\omega(x)$ is the concatenation of R/n F-values. In particular,

$$E_\omega(x) = F_\omega(y) \Big|\Big| F_\omega\left(\pi^n(y)\right) \Big|\Big| F_\omega\left(\pi^{2n}(y)\right) \Big|\Big| \cdots \Big|\Big| F_\omega\left(\pi^{(R/n-1)n}(y)\right).$$

W.l.o.g., for the sake of simplicity we assume in our proof that n divides R.

(v) In the query phase, Eve poses exactly M oracle queries. In the prediction phase, Eve has to submit a pair $(x^*, z^*) \in \{0,1\}^n \times \{0,1\}^n$, where x^* does not occur as input of an E-query in the query phase. Eve wins if $z^* = F_\omega(P_\omega(x^* \oplus k_\omega) \oplus k_\omega)$, i.e., if z^* equals the block of the first n bits of the keystream packet $E_\omega(x^*)$ corresponding to the initial state $P_\omega(x^* \oplus k_\omega) \oplus k_\omega$, i.e., the keystream packet corresponding to session key k_ω and initial value x^*.

(vi) Besides the number M of oracle queries, the essential cost parameter is the winning probability of Eve, which is measured with respect to the uniform distribution on Ω and the internal randomization of Eve.

Observe that generic TMD-TO attacks (as described in Section 3) against the LIZARD-construction can be formulated in a straightforward way as packet prediction or key recovery games in the sense of Definition 4. Here, the cost metric *data* (i.e., D) corresponds to the number of E-queries (possibly multiplied by the packet length), while each evaluation of the cipher components *MIX* and *OUTBLOCK* corresponds to a P-, P^{-1}-, or F-query in the sense of Definition 4. Note here that in a possible precomputation phase of a TMD-TO attack, only P-, P^{-1}-, or F-queries will be posed. Hence, the overall number of oracle queries in our game is a lower bound for the cost metric *overall time* (i.e., $T + P$; cf. Section 3) of the online phase and a possible precomputation phase of a corresponding generic TMD-TO attack against the LIZARD-construction. Consequently, a lower bound

on the number of oracle queries also lower bounds the minimum TMD-cost of TMD-TO attacks against the LIZARD-construction.

Note here that evaluations of the state transition function π by Eve do not count in our security analysis; it is supposed that π is completely known to Eve. The function π has to satisfy rule (L1) of Definition 3, but apart from that, no further security-relevant properties are required. Our lower bound arguments even work in the case that π is linear.

We conclude this section by describing how a random uniformly distributed π-iterative function can be generated.

Generating a random π-iterative function F Note that, as π is bijective, the strongly connected components of the directed graph $G_\pi = (\{0,1\}^n, E_\pi)$, where $E_\pi = \{(v, \pi(v)); v \in \{0,1\}^n\}$, are simple cycles $C^1, \cdots, C^s$ of sizes $d_1, \cdots, d_s$, which we call π-cycles.

For each π-cycle C^j, $1 \le j \le s$, fix a starting point $v_0^j \in C^j$. Note that $C^j = \{v_0^j, \cdots, v_{d_j-1}^j\}$, where for all i, $1 \le i \le d_j - 1$, it holds $v_i^j = \pi^i(v_0^j)$.

A uniformly distributed π-iterative function F can be defined by choosing for all j, $1 \le j \le s$, randomly and independently a uniformly distributed bitstring

$$b^j = \left(b_0^j, \cdots, b_{d_j-1}^j\right) \in \{0,1\}^{d_j}$$

and defining $F(v_i^j)$ for all i, $0 \le i \le d_j - 1$, by

$$F\left(v_i^j\right) = \left(b_i^j, b_{(i+1) \bmod d_j}^j, \cdots, b_{(i+n-1) \bmod d_j}^j\right).$$

Here we took into account that, by Definition 3, the sizes of the cycles are each larger than $R \ge n$. Note that the entropy of a random π-iterative function is 2^n.

5 The security lower bound proof

5.1 Preliminaries

In this section, we show the main result of this paper, a sharp security lower bound for the LIZARD-construction. At one essential point, our lower bound proof uses a combinatorial result proved by Chen, Lampe, Lee, Seurin, Steinberger in [13], namely Theorem 1 in Section 3, which is known as *Sum-Capture Theorem*.

For motivating the use of this result, let us consider the situation that Alice holds a secret triple $\omega = (k_\omega, P_\omega, F_\omega)$ and that Eve asked a number of queries, where $U, X, Y \subseteq \{0,1\}^n$ denote the sets of inputs of the P-queries, E-queries and F-queries asked by Eve so far, respectively.

Definition 5 (Critical Triples) A triple $(u, x, y) \in U \times X \times Y$ is called

- ω-critical if $x \oplus u = P_\omega(u) \oplus y$,
- ω-dangerous if (u, x, y) is ω-critical and (x, y) form an ω-collision in the sense that $F_\omega(y)$ equals the prefix of length n of the packet $E_\omega(x)$, and
- ω-sudden death if $x \oplus u = P_\omega(u) \oplus y = k_\omega$.

It can be easily derived from Definition 4 that from (u, x, y) is ω-sudden death it follows that (u, x, y) is ω-dangerous and ω-critical. Note that Eve can immediately check if a given

triple $(u, x, y) \in U \times X \times Y$ is ω-critical or even ω-dangerous. Note further that in our lower bound proof we will use a more general definition of collision than in Definition 5.

The following lemma shows that from Eve's point of view it is desirable that the choice of U, X, Y implies a sufficiently large set of ω-critical triples.

Lemma 1 *For all ω-critical triples $(u, x, y) \in U \times X \times Y$, it holds that if (u, x, y) is not ω-dangerous, then $x \oplus u \neq k_\omega$. If (u, x, y) is ω-dangerous, then $x \oplus u = k_\omega$ holds with significant probability.*

Consequently, if Eve manages to construct an ω-dangerous triple, then she wins with significant success probability. Note that the aim of the attack in Theorem 4 against the LIZARD-construction is to construct an ω-dangerous triple.

Proof of Lemma 1 If $x \oplus u = P_\omega(u) \oplus y$ and $x \oplus u = k_\omega$, then $y = P_\omega(x \oplus k_\omega) \oplus k_\omega$, which implies that, by definition, $F_\omega(y)$ equals the prefix of length n of $E_\omega(x)$. Consequently, if $F_\omega(y)$ does not equal the prefix of length n of $E_\omega(x)$, then $x \oplus u \neq k_\omega$.

For the other direction, we sketch only the proof; the complete proof can be found in [16]. We suppose that $x \oplus u = P_\omega(u) \oplus y$ and that $F_\omega(y)$ equals the prefix of length n of $E_\omega(x)$. As F_ω is randomly chosen, it follows with high probability that $y = P_\omega(x \oplus k_\omega) \oplus k_\omega$. Now let $M_\omega(u)$ denote the set of all $u' \in \{0,1\}^n$ for which $u' \oplus P_\omega(u') = u \oplus P_\omega(u)$. As P_ω is randomly chosen, it can be shown that $|M_\omega(u)| \leq n$ with high probability. Now observe that the events $x \oplus u = P_\omega(u) \oplus y$ and $y = P_\omega(x \oplus k_\omega) \oplus k_\omega$ imply that $x \oplus k_\omega \in M_\omega(u)$. Consequently, $x \oplus k_\omega = u$ with probability $|M_\omega(u)|^{-1}$. □

Note that as Eve knows $P_\omega(u)$ for all $u \in U$ and as Eve has unbounded computational power, she is able to compute from U, X, Y the set of all ω-critical and all ω-dangerous triples without further oracle queries.

Lemma 1 shows that Eve wins the game with high probability if she manages to pose the queries in such a way that for almost all keys $k \in \{0,1\}^n$ there is some ω-critical triple $(u, x, y) \in U \times X \times Y$ fulfilling $x \oplus u = P_\omega(u) \oplus y = k$. We mention here that there is an algorithm which reaches this goal with high probability with $\tilde{O}(2^{(2/3)n})$ oracle queries. As the time cost of the corresponding TMD-TO attack is much worse than that of the attack described in Theorem 4, we omit the description of this algorithm.

The Sum-Capture Theorem from [13], which we present in a slightly modified form, shows that reaching this goal with significantly less than $\tilde{O}(2^{(2/3)n})$ oracle queries succeeds only with exponentially small success probability.

Theorem 5 *Let P denote a uniformly random permutation over $\{0,1\}^n$, let $N = 2^n$ and fix an arbitrary number M, $9n \leq M \leq N/2$. Suppose that Eve (who is supposed to be a probabilistic algorithm) poses a sequence $U = \{u_1, \cdots, u_M\}$ of M P-queries. For any subsets $X, Y \subseteq \{0,1\}^n$ let*

$$\mu(P, U, X, Y) = |\{(u, x, y) \in U \times X \times Y; x \oplus u = y \oplus P(u)\}| .$$

Then the probability for the event that there are subsets $X, Y \subseteq \{0,1\}^n$ such that

$$\mu(P, U, X, Y) \geq \frac{M \cdot |X| \cdot |Y|}{N} + \frac{2M^2 \cdot \sqrt{|X| \cdot |Y|}}{N} + 3\sqrt{n \cdot M \cdot |X| \cdot |Y|} \qquad (10)$$

is at most $\frac{2}{N}$, where the probability is taken over the random choice of P and the internal randomization of Eve. □

5.2 The main theorem

In this subsection, we formulate our main technical result (Theorem 6) and start with a technical definition:

Definition 6 (The Number $B(M, R, n)$) For natural numbers M, R, n, with $R \geq n$, let

$$B(M, R, n) = 2^{-n} \cdot M^3 \cdot \left(R + n - 1 + 2\sqrt{R + n - 1}\right) + 3 \cdot \sqrt{n \cdot M^3 \cdot (R + n - 1)}.$$

Note that $B(M, R, n)$ equals the term on the right-hand side of Relation (10) for $|X| = M$ and $|Y| = (R + n - 1)M$.

Theorem 6 (Main Theorem) *Suppose that the parameters M, n, R satisfy the following rules for some number Δ:*

(1) $B(M, R, n) + 2 \cdot \Delta \cdot M + \frac{(R+n)\cdot M^2}{\Delta} \leq \left(1 - \frac{1}{\sqrt{2}}\right) \cdot 2^n$,

(2) $22 \cdot 2^{-(n-1)} \cdot R \cdot M^2 + \sqrt{\frac{n \cdot M}{2}} \leq \frac{\Delta - (R+n-1)}{R+n-1}$,

(3) $\Delta \cdot ((n + R) \cdot M) \leq \ln 2 \cdot 2^{n-3}$.

Then Eve's success probability to win the packet prediction game with parameters Δ, R, n with M oracle queries is bounded by

$$34 \cdot 2^{-n} + M \cdot e^{-n} + M \cdot (\Delta + 2) \cdot 2^{-(n-1)} + 11 \cdot (R + 4n) \cdot M \cdot 2^{-(n-1)}.$$

This implies the following asymptotic lower bound result, which can be derived straightforwardly from Theorem 6.

Corollary 1 (Main Corollary) *Let $\epsilon > 0$ and $a > 1$ be constants and suppose that $M \leq 2^{(\frac{2}{3}-\epsilon)n}$, $R \leq n^a$ and $\Delta = \lfloor 2^{\frac{1}{3}n} \rfloor$. Then M, R, n, Δ satisfy all rules in Theorem 6 and Eve's success probability to win the packet prediction game with parameters Δ, R, n with M oracle queries is bounded by $3 \cdot 2^{-\epsilon \cdot n}$, if n is large enough.* □

The remaining part of this paper is devoted to the proof of Theorem 6.

A first overview over the idea and the structure of the proof We prove the security bound of Theorem 6 for deterministic adversaries. This is justified by the fact that each upper bound on the success probability of deterministic adversaries posing M queries also holds for randomized adversaries posing M queries. We give a proof of this folklore result at the beginning of Section 5.4. The fact that Eve is assumed to be deterministic allows to assign to each elementary event $\omega = (k_\omega, P_\omega, F_\omega)$ a computation τ_ω, which is performed by Eve on ω, and which is either successful (i.e., Eve wins), or not. Let us denote by $\Omega^{\text{succ}} \subseteq \Omega$ the set of all elementary events for which τ_ω is successful. For showing that $\Pr_\Omega[\Omega^{\text{succ}}]$ fulfills the relation claimed in Theorem 6, we partition Ω into a set Ω^{bad} of bad events and a set Ω^{good} of good events, where bad events ω will have the property that the corresponding computation τ_ω yields some nontrivial information on the secret key k_ω, which helps Eve to win the game. An elementary event is called to be good if it is not bad.

This partition into bad and good elementary events allows to express the success probability of Eve as

$$\begin{aligned}\Pr_{\Omega}[\Omega^{\text{succ}}] &= \Pr_{\Omega}[\Omega^{\text{succ}}|\Omega^{\text{bad}}] \cdot \Pr_{\Omega}[\Omega^{\text{bad}}] + \Pr_{\Omega}[\Omega^{\text{succ}}|\Omega^{\text{good}}] \cdot \Pr_{\Omega}[\Omega^{\text{good}}] \\ &\leq \Pr_{\Omega}[\Omega^{\text{bad}}] + \Pr_{\Omega}[\Omega^{\text{succ}}|\Omega^{\text{good}}]. \end{aligned} \tag{11}$$

We prove Theorem 6 by deriving bounds for $\Pr_{\Omega}[\Omega^{\text{bad}}]$ and $\Pr_{\Omega}[\Omega^{\text{succ}}|\Omega^{\text{good}}]$. In particular, we will distinguish four sorts of bad events, namely sudden death events, black events, red events, and blue events, and denote the remaining sort of good elementary events to be green events. Corresponding to this, the proof Theorem 6 rests upon four pillars consisting in proving small upper bounds for the probability of 1.) the set of sudden death events, 2.) the set of black events, 3.) the event that a green elementary event determines a successful computation, and 4.) the set of red and blue events.

The proof of Theorem 6 is divided into three phases. In the first phase, which comprises Sections 5.3, 5.4, and 5.5, we slightly modify the operation mode of Alice, formalize the computational behavior of Eve, and discuss a number of basic properties of elementary events and computations of Eve. This enables us to give a more detailed overview about the remaining two phases of the proof, and to formulate our main lemma (Lemma 2). This lemma consists of four items corresponding to the formulation of the concrete bounds determining the four pillars of the proof mentioned above.

In the second phase, we develop the mathematical tools which are necessary for proving the items of Lemma 2 (Section 5.6 and Section 5.8). The intention of these tools, which we call basic methods, is to analyze and formalize Eve's knowledge about the secret elementary event held by Alice under the condition that a certain computation τ has already happened, i.e., that a certain number of oracle queries has been posed and answered. This knowledge corresponds to the probability space $\Omega(\tau)$ formed by all elementary events which are consistent with the computation τ. In the Consistency Lemma (Lemma 3 in Section 5.6) and the Smoothness Lemma (Lemma 5 in Section 5.8) the structure of this probability space $\Omega(\tau)$ is analyzed, especially that of the induced probability space $K(\tau)$ of all keys $k \in \{0,1\}^n$ which are consistent with τ. The six items forming Corollary 2 in Section 5.8, which result from this analysis, can be seen as a toolbox of methods for handling all the different situations occurring in the proofs of the items of Lemma 2.

Moreover, in the second phase we give the definitions of sudden death, black, red, blue, and green elementary events and discuss basic properties of these definitions (Section 5.7). Informally, an elementary event ω is defined to fulfill the sudden death rule if during the computation τ_ω an ω-sudden death triple will be generated (see Definition 5 and Lemma 1). We have seen already in Lemma 1 that a sudden death event allows to determine the secret key with high probability.

The elementary event ω is defined to be black if during τ_ω too many ω-critical triples are generated. Lemma 1 showed that black events are dangerous for Alice as a large number of ω-critical triples implies an ω-dangerous triple (i.e., a sudden death event) with high probability.

The elementary event ω is called red if during τ_ω too many ω-collisions are generated. The danger of red events follows by Theorem 4, where it is shown how to recover the secret key on the basis of a large set of ω-collisions.

Finally, ω is defined to be blue if the probability distribution corresponding to $K(\tau)$ differs too much from the uniform distribution. We have to exclude this case as our techniques for bounding the probability of certain bad events rest on the assumption that the keys in $K(\tau)$ are nearly uniformly distributed.

As the proofs of one part of Lemma 5 and some parts of Corollary 2 are quite long and tedious, we shifted them into Appendix C and B, respectively. In Appendix A, we derive a further basic method, a modified Chernoff bound technique, which is necessary for bounding the probability of red and blue elementary events.

In the third phase of the proof of Theorem 6, consisting of the Sections 5.9, 5.10, 5.11, and 5.12, we prove the four items of Lemma 2. As mentioned above, all these four proofs use the methods contained in Corollary 2.

We illustrate the modular structure of the proof of Theorem 6 in Fig. 3.

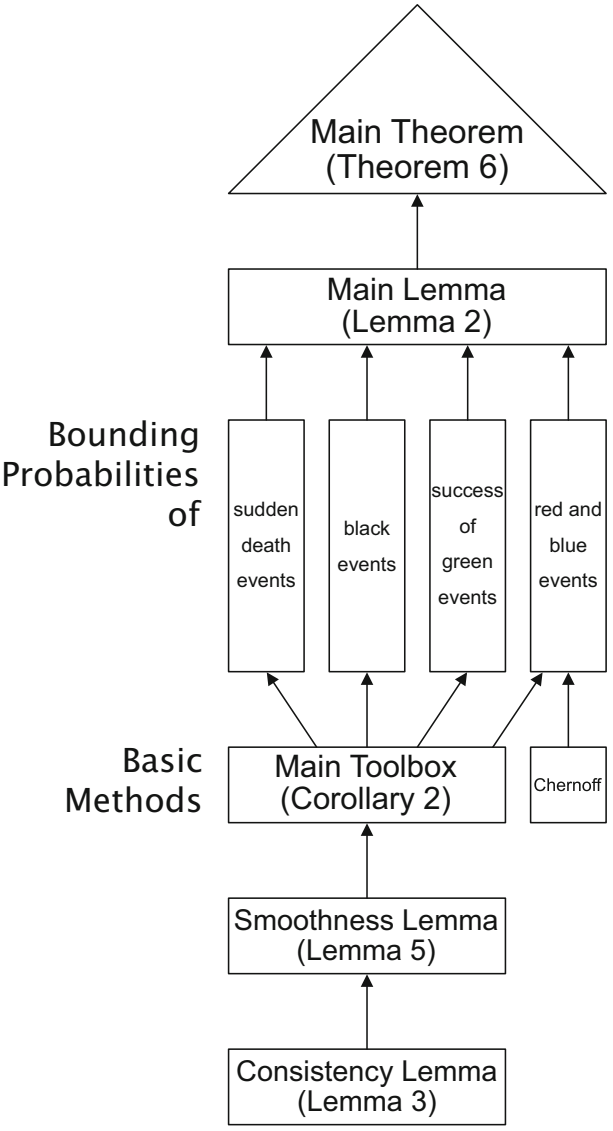

Fig. 3 The modular structure of the proof of the Main Theorem (Theorem 6)

5.3 Structural collisions, the friendly Alice, and sudden death

We will prove our security bound for a modified game, in which Alice is supposed to be *friendly* to Eve in the sense that in certain situations Alice provides some additional information to Eve. These situations will have to do with collisions, which can occur, e.g., between E-queries and F-queries, i.e., the E-query with an input x yields the same answer as the F-query with an input y. The reason for such a collision can be twofold. The first possible reason is that $y = P_\omega(x \oplus k_\omega) \oplus k_\omega$. We will call this type of collision a structural EF-collision. Another reason can be that the values y and $P_\omega(x \oplus k_\omega) \oplus k_\omega$ are a collision of the function F_ω. Note that in the case of a structural collision, Eve obtains the valuable information that $y = P_\omega(x \oplus k_\omega) \oplus k_\omega$. We have seen in Theorem 4 that the secret key can be computed by Eve on the basis of $2^{n/2}$ such pairs (x, y), which highlights the importance of structural collisions. Another type of collisions are EE-collisions (x, x') of E-queries with inputs $x \neq x'$. Here, too, we distinguish between structural collisions, where $P_\omega(x \oplus k_\omega) \oplus k_\omega = P_\omega(x' \oplus k_\omega) \oplus k_\omega$, and non-structural collisions, where this is not the case but $P_\omega(x \oplus k_\omega) \oplus k_\omega$ and $P_\omega(x' \oplus k_\omega) \oplus k_\omega$ are a collision of F_ω. The friendly Alice will inform Eve if Eve managed to discover a structural collision. Moreover, she follows a *sudden-death rule* (see Definition 9), which has to do with structural collisions.

Definition 7 (Structural Collisions)

- A pair (x, y), where $x, y \in \{0,1\}^n$, is called a structural EF-collision w.r.t. to an elementary event $\omega = (k_\omega, P_\omega, F_\omega)$ if
$$y = \pi^r(P_\omega(x \oplus k_\omega) \oplus k_\omega), \tag{12}$$
for some r, $-(n-1) \leq r \leq R-1$. Note that this implies that the n-bit block $F_\omega(y)$ is a subblock of packet $E_\omega(x)$ or has at least a nonempty intersection with packet $E_\omega(x)$.
- If (x, y) is a structural EF-collision w.r.t. ω, then the point $\bar{y} = P_\omega(x \oplus k_\omega) \oplus k_\omega$ is called the reference point of this collision.
- A pair (x, x'), where $x \neq x' \in \{0,1\}^n$, is called a structural EE-collision w.r.t. to ω if the initial states of the packets $E_\omega(x)$ and $E_\omega(x')$ come so close that these packets have a nonempty intersection, i.e., there is some number r, $1 \leq r \leq R-1$, such that
$$\begin{aligned} \pi^r(P_\omega(x \oplus k_\omega) \oplus k_\omega) &= P_\omega(x' \oplus k_\omega) \oplus k_\omega \text{ or} \\ \pi^r(P_\omega(x' \oplus k_\omega) \oplus k_\omega) &= P_\omega(x \oplus k_\omega) \oplus k_\omega. \end{aligned} \tag{13}$$
Note that this implies that the suffix of packet $E_\omega(x)$ starting at position r equals the prefix of packet $E_\omega(x')$ ending at position $R-(r-1)$, or that the suffix of packet $E_\omega(x')$ starting at position r equals the prefix of packet $E_\omega(x)$ ending at position $R-(r-1)$.

Note here again that structural EF-collisions are exactly those collisions which are exploited in the TMD tradeoff attacks against the LIZARD-construction described in Theorem 4.

Suppose that Alice holds the elementary event $\omega = (k_\omega, P_\omega, F_\omega)$ and communicates with Eve. Note that if Eve detects a collision in the answers to her questions, she is not able to decide if it is a structural collision, and, in the case of a structural collision, she is not able to compute the reference point. The friendly Alice helps her in such situations in the following way:

Definition 8 (The Friendly Alice)

- Whenever Eve poses an F-query with some input $y \in \{0,1\}^n$ which forms a structural EF-collision (x, y) w.r.t. ω for some $x \in \{0,1\}^n$ which already occurred as input of an E-query posed before, then, besides publishing $F_\omega(y)$, Alice confirms a structural collision, publishes a pointer to the input x and publishes the reference point $P_\omega(x \oplus k_\omega) \oplus k_\omega$ of this collision.
- Whenever Eve poses an E-query with some input $x \in \{0,1\}^n$ which forms a structural EF-collision (x, y) w.r.t. ω for some y which already occurred as input of an F-query posed before, then, besides publishing $E_\omega(x)$, Alice confirms a structural collision, publishes a pointer to y and publishes the reference point $P_\omega(x \oplus k_\omega) \oplus k_\omega$ of this collision.
- Suppose that Eve poses an E-query with some input $x \in \{0,1\}^n$ which forms a structural EE-collision (x, x') w.r.t. ω for some x' which already occurred as input of another E-query posed before. Suppose w.l.o.g. that $\pi^r(P_\omega(x \oplus k_\omega) \oplus k_\omega) = P_\omega(x' \oplus k_\omega) \oplus k_\omega$ for some r, $1 \le r \le R-1$. Then, besides publishing $E_\omega(x)$, Alice confirms a structural EE-collision and publishes a pointer to x'. Moreover, Alice publishes the value $y = \pi^r(P_\omega(x \oplus k_\omega) \oplus k_\omega) = P_\omega(x' \oplus k_\omega) \oplus k_\omega$, the value $F_\omega(y)$, and the reference points $\bar{y} = P_\omega(x \oplus k_\omega) \oplus k_\omega$ and y of the resulting structural EF-collisions (x, y) and $(x', \bar{y})$.

Next we formulate the *sudden-death rule*.

Definition 9 (Sudden Death) Suppose that Alice holds an elementary event $\omega = (k_\omega, P_\omega, F_\omega)$ and consider a situation in which Eve already posed a number of queries. A pair (x, u), where $x, u \in \{0,1\}^n$, is called a sudden-death pair w.r.t. ω if the following conditions are fulfilled:

- Eve has already discovered a structural EF-collision (x, y) (which implies that Eve has asked an E-query with input x).
- Eve has already asked a P-query with input u or a P^{-1}-query with output u.
- It holds $x \oplus u = k_\omega$.

Whenever Eve asks a query which causes a sudden-death pair w.r.t. to the secret ω held by Alice, then Alice immediately gives up, the game stops and Eve wins.

The motivation for considering sudden death pairs is given with Lemma 1. There it is proved that for each sudden death pair (x, u) it holds with significant probability that $x \oplus u$ equals the secret session key k_ω. Note that the TMD-TO attack described in Theorem 4 consists in generating a sudden-death pair.

Looking at the remarks motivating the consideration of structural collisions and sudden death pairs, it should be clear the the additional information given by a friendly Alice helps Eve in winning the game. The friendliness of Alice increases Eve's chances to win the prediction game (any additional information which is provided for free does so). Consequently, it is sufficient to show the security lower bound of Theorem 6 for an adversary Eve who plays the packet prediction game with a friendly Alice. Note that the reason for considering a friendly Alice instead of an unrestricted one is to simplify the proof of Theorem 6.

5.4 Formalizing the computational behavior of Eve

First note the well-known fact, proved, e.g., in [13] and many other papers, that it is sufficient to prove our security lower bound for deterministic adversaries. For showing this suppose that Eve is randomized and that the randomization is organized by a number B of random bits. Then Eve's success probability can be written as

$$\Pr[\text{Eve successful}] = \sum_{b\in\{0,1\}^B} 2^{-B} \Pr[\text{Eve successful} \mid b], \tag{14}$$

where $\Pr[\text{Eve successful} \mid b]$ denotes the success probability of the deterministic algorithm obtained by assigning b to Eve's random bits.

Consequently, if we show an upper bound on the success probability of all deterministic adversaries then, by (14), this bound also holds for randomized adversaries.

Therefore, we assume from now on that Eve is deterministic.

Remember that, during each computation, Eve poses at most M oracle queries, where she either wins via sudden death of Alice or she stops after M queries with the publication of a pair consisting of an initial value $x^* \in \{0,1\}^n$ and a keystream prefix $z^* \in \{0,1\}^n$ as final output.

We identify such computations with transcripts

$$\tau = \left((type_1, input_1, output_1), \cdots, (type_j, input_j, output_j)\right),$$

$j \le M$, which are defined to be sequences of query triples corresponding to the oracle queries posed during the computation.

Here, $type_r \in \{F, P^{-1}, P, E\}$, $input_r$, and $output_r$ denote the type, the input, and the output of the r-th oracle query, $r = 1, \cdots, j$, respectively. Note that the output of an oracle query can, besides the output function values of P_ω, or P_ω^{-1}, or F_ω, or E_ω, contain additional information about structural collisions discovered by this query (see Definition 8).

If τ has length M, then $(x^*(\tau), z^*(\tau)) \in \{0,1\}^n \times \{0,1\}^n$ denotes the (initial value, keystream prefix) pair published after τ based on τ.

For transcripts τ of length j, $1 \le j \le M$, and numbers i, $1 \le i \le j$, we denote by $\tau^{\le i}$ the subtranscript corresponding to the first i queries along τ.

Each elementary event $\omega \in \Omega$ defines a unique transcript τ_ω corresponding to the computation of Eve on ω.

The length of τ_ω can be smaller than M. This is the case if and only if the next query after the last query of τ_ω produces a sudden-death pair w.r.t. ω (see Definition 9). In this case, this next query is not counted to be a part of τ_ω.

Let us denote by $\Omega^{\text{s.death}}$ the set of all elementary events ω for which τ_ω leads to the generation of a sudden-death pair w.r.t. ω, and note that this is equivalent to τ_ω having length smaller than M.

Eve's computation τ_ω on an elementary event ω is successful if and only if either the length of τ_ω is smaller than M (i.e., $\omega \in \Omega^{\text{s.death}}$) or the first n bits of the keystream packet corresponding to $x^*(\tau_\omega)$ via ω coincide with $z^*(\tau_\omega)$, i.e.,

$$F_\omega\left(P_\omega\left(x^*(\tau_\omega) \oplus k_\omega\right) \oplus k_\omega\right) = z^*(\tau_\omega).$$

We denote by $\Omega^{\text{succ}} \subseteq \Omega$ the set of all elementary events leading to a successful computation. Note that $\Omega^{\text{s.death}} \subseteq \Omega^{\text{succ}}$.

5.5 Further basic definitions and the idea of the proof of Theorem 6

For all j, $1 \leq j \leq M$, we denote by $\mathcal{T}^j$ the set of all transcripts τ of length j (i.e. consisting of j query triples) which occur with positive probability, i.e., for which there is some $\omega \in \Omega$ such that τ is the prefix of length j of τ_ω. With the following definition for each j, $1 \leq j \leq M$, and each transcript $\tau \in \mathcal{T}^j$, we define the following sets corresponding to the queries along τ. Moreover, we define the notions τ-consistent elementary event and τ-consistent key.

Definition 10 (Components of Transcripts)

- $X(\tau) = \{x \in \{0,1\}^n ;\ \tau$ contains an E-query with input $x\}$,
- $Y(\tau) = \{y \in \{0,1\}^n ;\ \tau$ contains an F-query with input $y\}$. Note that we put also those y to $Y(\tau)$ which occur at the right side of a structural EF-collision that was disclosed by Alice as additional information to an EE-collision, see Definition 8.
- $U(\tau) = \{u \in \{0,1\}^n ;\ \tau$ contains a P-query with input u, or a P^{-1}-query with output $u\}$,
- $V(\tau) = \{v \in \{0,1\}^n ;\ \tau$ contains a P-query with output v, or a P^{-1}-query with input $v\}$,
- $X^*(\tau) = \{x \in X(\tau);\ x$ there is a $y \in Y(\tau)$ such that (x, y) is a structural EF-collision discovered during $\tau\}$,
- $\bar{Y}^*(\tau) = \{\bar{y} \in \{0,1\}^n ;\ \bar{y}$ is the reference point of some structural EF-collision discovered during $\tau\}$,
- $Coll(\tau) = \{(x, \bar{y});$ where $x \in X^*(\tau)$, and $\bar{y} \in \bar{Y}^*(\tau)$ is the reference point of a structural EF-collision (x, y) discovered during $\tau\}$,
- $\bar{Y}^{(r)}(\tau) = \{\bar{y} \in \{0,1\}^n ;\ \pi^r(\bar{y}) \in Y(\tau)\}$,
- $\bar{Y}(\tau) = \bigcup_{r=-(n-1)}^{R-1} \bar{Y}^{(r)}(\tau)$.
- For all j, $1 \leq j \leq M$, and transcripts $\tau \in \mathcal{T}^j$, we denote by $\Omega(\tau)$ the set of all τ-consistent elementary events ω, i.e.,

$$\Omega(\tau) = \{\omega;\ \tau_\omega^{\leq j} = \tau\}.$$

- $\Omega(\tau)$ defines the set $K(\tau) \subseteq \{0,1\}^n$ of τ-consistent keys, i.e.,

$$K(\tau) = \{k_\omega;\ \omega \in \Omega(\tau)\}.$$

Important Note Remember that we suppose Alice to be friendly in the sense of Definition 8. This implies that the oracle answers of Alice along τ yield the complete knowledge about possible structural collisions, i.e., this knowledge is part of τ (see Section 5.4). This implies that if some $(x, y) \in X(\tau) \times Y(\tau)$ is a structural EF-collision for *some* elementary event in $\Omega(\tau)$, then it is a structural EF-collision with the same reference point for *all* elementary events in $\Omega(\tau)$. The same is true for structural EE-collisions. This justifies the definitions of $Coll(\tau)$ and $\bar{Y}^*(\tau)$ and $X^*(\tau)$.

Observe that $X(\tau)$ corresponds to the set of all initial values for which Eve gets the corresponding keystream packet from Alice during τ, and that $X^*(\tau)$ corresponds to the set of all those initial values for which Eve even knows the initial state of the corresponding packet. These known initial states are contained in the set $\bar{Y}^*(\tau)$.

The set $\bar{Y}(\tau)$ corresponds to the set of all initial states for which a part of the corresponding keystream packet has been discovered during τ.

Note further that $Coll(\tau)$ yields all information also about structural EE-collisions discovered during τ. This is because, due to Definition 8, for each structural EE-collisions

(x, x') discovered during τ, there is some $y \in Y(\tau)$ such that (x, y) and (x', y) are structural EF-collisions discovered during τ.

Moreover, $Coll(\tau)$ defines a one-to-one correspondence between $X^*(\tau)$ and $\bar{Y}^*(\tau)$, which is established by the bijection $P_\omega(x \oplus k_\omega) \oplus k_\omega$ for an $\omega \in \Omega(\tau)$. (Note that, by definition, this bijection is the same for all τ-consistent elements $\omega \in \Omega(\tau)$.)

Note that $\Omega(\tau)$ defines a probability distribution $\Pr_\tau$ on $K(\tau)$. For all $k \in K(\tau)$, it holds

$$\Pr_\tau[k] = \frac{|\{\omega \in \Omega(\tau); k_\omega = k\}|}{|\Omega(\tau)|}.$$

After the computation τ has happened, Eve's knowledge about the secret ω can be identified with the probability space $\Omega(\tau)$ with the uniform distribution. Note that the induced probability distribution $\Pr_\tau$ on $K(\tau)$ does not need to be uniform. Actually, the analysis of the distribution $\Pr_\tau$ on $K(\tau)$ will be the key ingredient for proving Theorem 6.

For describing the main idea of the proof, let us consider the situation that Alice holds a secret $\omega = (k_\omega, P_\omega, F_\omega)$ and that Eve performed M queries without generating a sudden-death pair. Let us denote by τ the corresponding transcript τ_ω.

Clearly, during τ the set of τ-consistent keys becomes smaller and smaller. For getting a first impression how key candidates $k \in \{0, 1\}^n$ are discarded during τ, suppose that τ contains query triples (P, u, v), (E, x, p), (F, y, z) for which $x \oplus u = k$ and $\pi^r(v \oplus k) = y$ for some r, $0 \le r \le R - 1$. Then there are three possibilities:

1.) z does not equal the substring of packet $p \in \{0, 1\}^R$ starting at position $r + 1$, or, if $r > R - n$, the prefix of z of length $R - r$ does not equal the suffix of length $R - r$ of packet p. Then k can not be the right key, i.e., $k \notin K(\tau)$.
2.) z equals the substring of packet $p \in \{0, 1\}^R$ starting at position $r + 1$, or, if $r > R - n$, the prefix of z of length $R - r$ equals the suffix of length $R - r$ of packet p, but $(x, v \oplus k)$ does not belong to $Coll(\tau)$. This implies that (x, y) does not form a structural EF-collision (which would be the case if k was the right key k_ω) and that the collision of z with p is caused by a (nonstructural) internal collision of F_ω. Consequently, $k \notin K(\tau)$.
3.) z equals the substring of packet $p \in \{0, 1\}^R$ starting at position $r + 1$, or, if $r > R - n$, the prefix of z of length $R - r$ equals the suffix of length $R - r$ of packet p and $(x, v \oplus k) \in Coll(\tau)$. Then it also holds that $k \notin K(\tau)$. Otherwise, if $k = k_\omega$, then (x, u) would form a sudden-death pair and the computation would have stopped before τ was completed.

After τ is completed, Eve has to choose a pair $(x^*(\tau), z^*(\tau))$. She is in a promising position if one of the following two conditions is fulfilled.

Condition-1 $K(\tau)$ contains only a small number of keys. In this case, Eve can choose one of the few keys in $K(\tau)$, say k', and check the hypothesis $k' = k_\omega$ as follows. She looks for some $y \in Y(\tau)$ and some r, $0 \le r \le R - n$, such that $v = \pi^{-1}(y) \oplus k' \in V(\tau)$. In this case let $u = P_\omega^{-1}(v)$. If $x = u \oplus k' \in X(\tau)$, then $k' \ne k_\omega$, as otherwise Alice would have indicated before that (x, u) is a sudden-death pair. If there is no such pair (y, r), then Eve fixes an arbitrary $y \in Y(\tau)$ and poses one additional P^{-1} query with input $y \oplus k'$ and obtains $u = P_\omega^{-1}(y \oplus k')$. If $k' = k_\omega$ and $u \oplus k' \in X(\tau)$, then (x, u) is a sudden-death pair and Eve wins immediately. If $x = u \oplus k' \notin X(\tau)$ and $k' = k_\omega$, then Eve wins with $x^*(\tau) = x$ and $z^*(\tau) = F_\omega(y)$.

Condition-2 The key k_ω belongs to a small set $K' \subseteq K(\tau)$ such that $\Pr_\tau[k']$ is nontrivially large for all $k' \in K'$. In this case, Eve tests for all $k' \in K'$ the hypothesis $k' = k_\omega$ as in Condition-1.

The structure of the remaining part of the proof of Theorem 6 Remember that we denote by Ω^{succ} the set of all elementary events $\omega \in \Omega$ for which the computation τ_ω is successful. We have to bound the probability $\Pr_\Omega[\Omega^{\text{succ}}]$ according to the relation in Theorem 6. Our proof starts with a combinatorial characterization of τ-consistency of elementary events and keys (see Section 5.6). This characterization will lead to the formulation of three properties of elementary events ω, for which the following holds. Event ω has one of these properties if and only if $K(\tau_\omega)$ satisfies Condition-1 or Condition-2 (see Lemma 5). We will identify these three properties with the colors black, red and blue and denote by Ω^{black}, resp. Ω^{red}, resp. Ω^{blue} the sets of elementary events having the corresponding property (see Section 5.7). We will further define an elementary event ω to be green, if it is neither red, nor black, nor blue, nor belongs to $\Omega^{\text{s.death}}$, and will denote the set of all green elementary events by Ω^{green}.

We prove Theorem 6 by using the following relation:

$$\begin{aligned}\Pr_\Omega\left[\Omega^{\text{succ}}\right] \leq\ & \Pr_\Omega\left[\Omega^{\text{black}} \cap \Omega^{\text{succ}}\right] + \Pr_\Omega\left[\left(\Omega^{\text{red}} \cup \Omega^{\text{blue}}\right) \cap \Omega^{\text{succ}}\right] \\ & + \Pr_\Omega\left[\left(\Omega^{\text{s.death}} \setminus \left(\Omega^{\text{black}} \cup \Omega^{\text{red}} \cup \Omega^{\text{blue}}\right)\right) \cap \Omega^{\text{succ}}\right] \\ & + \Pr_\Omega\left[\Omega^{\text{green}} \cap \Omega^{\text{succ}}\right].\end{aligned}$$

Consequently, $\Pr_\Omega[\Omega^{\text{succ}}]$ can be upper bounded by

$$\begin{aligned}\Pr_\Omega\left[\Omega^{\text{succ}}\right] \leq\ & \Pr_\Omega\left[\Omega^{\text{black}}\right] + \Pr_\Omega\left[\Omega^{\text{red}} \cup \Omega^{\text{blue}}\right] \\ & + \Pr_\Omega\left[\Omega^{\text{s.death}} \setminus \left(\Omega^{\text{black}} \cup \Omega^{\text{red}} \cup \Omega^{\text{blue}}\right)\right] \\ & + \Pr_\Omega\left[\Omega^{\text{succ}} \cap \Omega^{\text{green}}\right]. \end{aligned} \tag{15}$$

Black and red elementary events will have the important property that for all transcripts τ the following holds: if one event $\omega \in \Omega(\tau)$ is black (resp. red), then all events in $\Omega(\tau)$ are black (resp. red). This justifies to define transcripts τ to be black (resp. red) if at least one τ-consistent elementary event is black (resp. red). All transcripts which are neither red nor black are called to be green. Note that for green transcripts τ, the set $\Omega(\tau)$ can contain green elementary events and blue elementary events.

This allows to rewrite the probability $\Pr_\Omega[\Omega^{\text{succ}} \cap \Omega^{\text{green}}]$ as

$$\begin{aligned}\Pr_\Omega[\Omega^{\text{succ}} \cap \Omega^{\text{green}}] &= \Pr_\Omega[\Omega^{\text{green}}] \cdot \Pr_{\Omega^{\text{green}}}[\Omega^{\text{succ}}] \\ &\leq \Pr_{\Omega^{\text{green}}}[\Omega^{\text{succ}}], \end{aligned} \tag{16}$$

where $\Pr_{\Omega^{\text{green}}}[\Omega^{\text{succ}}]$ can be written as

$$\begin{aligned}\Pr_{\Omega^{\text{green}}}[\Omega^{\text{succ}}] &= \sum_{\tau \in \mathcal{T}^M_{\text{green}}} \Pr_{\Omega^{\text{green}}}[\Omega^{\text{succ}} \cap \Omega^{\text{green}}(\tau)] \\ &= \sum_{\tau \in \mathcal{T}^M_{\text{green}}} \Pr_{\Omega^{\text{green}}}[\tau] \cdot \Pr_{\Omega^{\text{green}}(\tau)}[\Omega^{\text{succ}}]. \end{aligned} \tag{17}$$

Here, $\mathcal{T}^M_{\text{green}}$ denotes the set of all green transcripts that have length M, and $\Omega^{\text{green}}(\tau)$ denotes the set of all green elementary events in $\Omega(\tau)$.

Note that we occasionally use the following denotation for conditional probabilities. Let A, B be subsets (events) of the probability space Ω. Then we write

$$\Pr_B[A] := \Pr_\Omega[A \mid B] = \frac{\Pr_\Omega[A \cap B]}{\Pr_\Omega[B]}.$$

By relations (15), (16) and (17), Theorem 6 follows directly from:

Lemma 2 (Main Lemma)

(i) *It holds that*

$$\Pr_\Omega\left[\Omega^{\text{s.death}} \setminus \left(\Omega^{\text{black}} \cup \Omega^{\text{red}} \cup \Omega^{\text{blue}}\right)\right] \leq 2^{-(n-1)} \cdot (\Delta + 2) \cdot M.$$

(ii) *It holds that* $\Pr_\Omega[\Omega^{\text{black}}] \leq 34 \cdot 2^{-n}$.

(iii) *For all* $\tau \in \mathcal{T}^M_{\text{green}}$, *it holds that*

$$\Pr_{\Omega^{\text{green}}(\tau)}[\Omega^{\text{succ}}] \leq 11 \cdot (R + 4n) \cdot M \cdot 2^{-(n-1)}.$$

(iv) *It holds that* $\Pr_\Omega[\Omega^{\text{red}} \cup \Omega^{\text{blue}}] \leq M \cdot e^{-n}$.

We will prove Lemma 2 in Section 5.8 and the sections following it.

5.6 Basic methods I: The characterization of τ-consistency

Definition 11 (Critical Points) Let $k \in \{0,1\}^n$. A point $u \in U(\tau)$ is called (τ, k)-critical if at least one of the following conditions is fulfilled.

C1: $u \oplus k \in X(\tau) \setminus X^*(\tau)$ and $P_\tau(u) \oplus k \in \bar{Y}(\tau) \setminus \bar{Y}^*(\tau)$.
C2: $u \oplus k \in X^*(\tau)$.
C3: $P_\tau(u) \oplus k \in \bar{Y}^*(\tau)$.

Here, $P_\tau(u)$ denotes the output of the P-query on input u along τ, resp. the input of the P^{-1}-query with output u.

The notion of (τ, k)-critical points allows to characterize τ-consistency.

Lemma 3 (Consistency Lemma) *A key* $k \in \{0,1\}^n$ *is not* τ*-consistent (i.e.,* $k \notin K(\tau)$*), if and only if there is a* (τ, k)*-critical point* $u \in U(\tau)$.

Proof of Lemma 3 We first prove the if-direction.

Let $k \in \{0,1\}^n$ and suppose that there is some $u \in U(\tau)$ which is (τ, k)-critical.

For deriving a contradiction we assume that $k \in K(\tau)$, i.e., that there is some $\omega \in \Omega(\tau)$ with $k_\omega = k$.

Suppose first that u is (τ, k)-critical via condition C1 of Definition 11.

By definition, $P_\tau(u) \oplus k = P_\omega(u) \oplus k_\omega \in \bar{Y}(\tau)$, which implies the existence of some r, $-(n-1) \leq r \leq R-1$ such that $\pi^r(P_\omega(u) \oplus k_\omega) \in Y(\tau)$. This implies, that $(u \oplus k_\omega, \pi^r(P_\omega(u) \oplus k_\omega))$ has to be classified as a structural EF-collision with reference point $P_\omega(u) \oplus k_\omega$ along τ. But this can not be true, as, by Definition 9, $(u \oplus k, u)$ would form a sudden-death pair w.r.t. ω, which implies that $\omega \notin \Omega(\tau)$.

Suppose now that u is (τ, k)-critical via condition C2 of Definition 11. If $u \oplus k = u \oplus k_\omega \in X^*(\tau)$, then $(u \oplus k, u)$ is again a sudden-death pair w.r.t. ω, which implies that $\omega \notin \Omega(\tau)$.

If $P_\tau(u) \oplus k \in \bar{Y}^*(\tau)$ (condition C3), then $(u \oplus k, P_\tau(u) \oplus k) \in \mathit{Coll}(\tau)$, which again implies that $(u \oplus k, u)$ is a sudden-death pair and that $\omega \notin \Omega(\tau)$.

Let us now show the only-if direction of Lemma 3.

We fix some j, $1 \leq j \leq M$, some transcript $\tau \in \mathcal{T}^j$ with $\Pr_\Omega[\tau] > 0$ and some key $k \in \{0,1\}^n$ for which there do not exist (τ, k)-critical points $u \in U(\tau)$ in the sense of Definition 11.

We have to show that k is τ-consistent.

We do this by constructing a permutation P' over $\{0,1\}^n$ and a π-iterative function $F' : \{0,1\}^n \longrightarrow \{0,1\}^n$ such that $\omega' = (k, P', F') \in \Omega(\tau)$.

For all inputs $x \in X(\tau)$, $u \in U(\tau)$, $v \in V(\tau)$, $y \in Y(\tau)$ of oracle queries posed during τ, we denote by $E_\tau(x)$, $P_\tau(u)$, $P_\tau^{-1}(v)$, and $F_\tau(y)$, resp., the corresponding oracle answers given by Alice during τ. P' and F' have to satisfy the condition that $P'(u) = P_\tau(u)$ and $F'(y) = F_\tau(u)$ for all $u \in U(\tau)$ and $y \in Y(\tau)$, respectively.

We now have to define P' and F' outside of $U(\tau)$ and $Y(\tau)$, respectively, in such a way that ω' is τ-consistent.

We do this by defining P' and F' along the (k, P')-paths $(u \oplus k, u, P'(u) \oplus k)$ for all $u \in \{0,1\}^n$, where we go with u through $\{0,1\}^n$ in a certain order.

Hereby, we dynamically maintain a set $\mathit{Target}(P')$, which is initially set to $\{0,1\}^n \setminus V(\tau)$. Whenever we define $P'(u)$ for a *new* u, we delete $P'(u)$ from $\mathit{Target}(P')$.

- **Phase 1:** Here we consider all $u \in \{0,1\}^n$ for which $u \oplus k \in X^*(\tau)$.

  Then it holds $u \notin U(\tau)$ as otherwise u would be (τ, k)-critical via condition C2 of Definition 11.

  We define $P'(u) = \bar{y} \oplus k$, where $\bar{y}$ denotes the unique point in $\bar{Y}^*(\tau)$ for which $(u \oplus k, \bar{y}) \in \mathit{Coll}(\tau)$. Note that the point $\bar{y} \oplus k$ does not belong to $V(\tau)$. Otherwise, if $\bar{y} \oplus k$ would equal $P_\tau(u')$ for some $u' \neq u \in U(\tau)$, then u' would be (τ, k)-critical via condition C3.

  We define F' on the set $\{\pi^r(P'(u) \oplus k); r = -(n-1), \cdots, R-1\}$ according to the packet $E_\tau(u \oplus k)$. Note here that if $-(n-1) \leq r < 0$, then $E_\tau(u \oplus k)$ determines only a suffix of $F'(\pi^r(P'(u) \oplus k))$, and that if $R - n - 1 < r \leq R - 1$, then $E_\tau(u \oplus k)$ determines only a prefix of $F'(\pi^r(P'(u) \oplus k))$.
- **Phase 2** concerns the (k, P')-paths through those $u \in U(\tau)$ for which $u \oplus k \in X(\tau) \setminus X^*(\tau)$. Note that for these $u \in U(\tau)$, as they are not (τ, k)-critical, it holds $\bar{y} := P_\tau(u) \oplus k \notin \bar{Y}(\tau)$.

  This implies that for all r, $-(n-1) \leq r \leq R-1$, it holds that $\pi^r(\bar{y})$ is not in $Y(\tau)$, which allows us to define $F'(\pi^r(\bar{y}))$ according to the packet $E_\tau(u \oplus k)$.
- **Phase 3** considers all $u \notin U(\tau)$ for which $u \oplus k \in X(\tau) \setminus X^*(\tau)$.

  Here, $P'(u)$ has to be chosen in such a way that $P'(u) \oplus k \notin \bar{Y}(\tau)$. Otherwise, there would exist some r, $-(n-1) \leq r \leq R-1$, such that $(u \oplus k, \pi^r(P'(u) \oplus k))$ is an *EF*-collision discovered during τ, which would imply that $u \oplus k \in X^*(\tau)$ and contradict the assumption made for phase 3.

  Corresponding to this, we define a set

$$\mathit{Forbidden}(u) = \{v \in \{0,1\}^n ; v \oplus k \in \bar{Y}(\tau)\},$$

  and choose

$$P'(u) \in \mathit{Target}(P') \setminus \mathit{Forbidden}(u).$$

Note that for all remaining $u \in \{0,1\}^n$ the values of $P'(u)$ can be freely chosen in $Target(P')$. For all remaining $y \in \{0,1\}^n$, the values of $F'(y)$ can also be freely chosen $\{0,1\}^n$ in such a way that the π-iterativeness of F' is maintained. □

5.7 Assigning colors to elementary events, transcripts, and keys

We will now assign the colors red, black, blue and green to transcripts, elementary events, and keys. There will be three colors, namely black, red, and blue, which have to be considered as bad in the sense that if ω has a bad color, then τ_ω yields some nontrivial information which helps Eve to win the game.

Let us start with the definition of black elementary events, which is partly based on considering the following equivalence relation $\equiv_P$, induced by a permutation P over $\{0,1\}^n$.

Definition 12 (The Relation $\equiv_P$) Let P denote a permutation of $\{0,1\}^n$ and let U be an arbitrary subset of $\{0,1\}^n$.

- For all $u, u' \in U$ let $u \equiv_P u'$ if and only if $u \oplus P(u) = u' \oplus P(u')$.
- Let $Max(P, U)$ denote the maximal size of an equivalence class w.r.t. $\equiv_P$ in U.

Definition 13 (Critical Keys) A key $k \in \{0,1\}^n$ is called to be τ-critical if there is some $u \in U(\tau)$ such that $u \oplus k \in X(\tau)$ and $P_\tau(u) \oplus k \in \bar{Y}(\tau)$.

Note that $k \in \{0,1\}^n$ is called to be τ-critical if there is some $u \in U(\tau)$ such that u is (τ, k)-critical with regard to condition C1 in Definition 11.

Definition 14 (Alive Elementary Events) For all numbers j, $1 \le j \le M$, we call an elementary event $\omega \in \Omega$ to be j-alive if τ_ω contains at least j queries, i.e., the first $j-1$ queries and answers and the j-th query along τ_ω do not generate a sudden death pair with respect to ω.

Definition 15 (Black Transcripts and Elementary Events)

- For all j, $1 \le j \le M$, we call a transcript $\tau \in \mathcal{T}^j$ to be black if the number of τ-critical keys (see Definition 13) exceeds
$$B(M,R,n) = 2^{-n} \cdot M^3 \cdot \left(R + n - 1 + 2\sqrt{R+n-1}\right) + 3\sqrt{n \cdot M^3 \cdot (R+n-1)}$$
or if
$$Max(P_\tau, U(\tau)) > 5,$$
where $P_\tau : U(\tau) \longrightarrow \{0,1\}^n$ denotes the injective mapping corresponding to the P- and P^{-1}-queries in τ.
- For all j, $1 \le j \le M$, an elementary event ω is called j-black if ω is j-alive and the transcript $\tau_\omega^{\le j}$, corresponding to the first j queries along τ_ω, is black.
- Let Ω^j_{black} denote the set of all j-black elementary events and $\mathcal{T}^j_{\text{black}}$ the set of all black transcripts $\tau \in \mathcal{T}^j$.
- Let $\Omega^{\text{black}} = \bigcup_{j=1}^{M} \Omega^j_{\text{black}}$.

Let us next define red transcripts.

Definition 16 (Red Transcripts and Elementary Events)

- For all j, $1 \leq j \leq M$, we call a transcript $\tau \in \mathcal{T}^j$ to be red if it is not black and $|X^*(\tau)| > \Delta$. (Remember that Δ denotes some balancedness parameter, which was introduced in Theorem 6).
- An elementary event ω is called j-red if ω is j-alive and the transcript $\tau_\omega^{\leq j}$ is red.
- Let Ω_{red}^j denote the set of all j-red elementary events and $\mathcal{T}_{\text{red}}^j$ the set of all red transcripts $\tau \in \mathcal{T}^j$.
- Let $\Omega^{\text{red}} = \bigcup_{j=1}^M \Omega_{\text{red}}^j$.

Note that one strategy of Eve could be to pose queries in a first phase in such a way that for the resulting transcript τ it holds that the set $K(\tau)$ of τ-consistent keys is small, and then to try each key in $K(\tau)$ if it fits. Redness and blackness of transcripts τ cover exactly the case in which this strategy could be successful:

Lemma 4 *For all j, $1 \leq j \leq M$, and $\tau \in \mathcal{T}^j$ the following holds. If τ is neither red nor black, then*

$$|K(\tau)| \geq 2^n - B(M, R, n) - 2 \cdot \Delta \cdot j.$$

Proof of Lemma 4 From Definition 11 and Lemma 3 we know that $k \in \{0,1\}^n \setminus K(\tau)$ if and only if there is some $u \in U(\tau)$ such that u is (τ, k)-critical via condition C1 or via condition C2. Condition C1 implies that k is τ-critical in the sense of Definition 13. As τ is not black, the number of such keys is bounded by $B(M, R, n)$. Condition C2 implies that $k \in X^*(\tau) \oplus U(\tau)$ or $k \in \bar{Y}^*(\tau) \oplus V(\tau)$. [2]

As τ is not red, it holds that $|X^*(\tau) \oplus U(\tau)| \leq \Delta \cdot j$ and $|\bar{Y}^*(\tau) \oplus U(\tau)| \leq \Delta \cdot j$. □

The motivation for considering blue elementary events is as follows. We have seen above that $\Omega(\tau)$, the set of all possible events if Eve sees τ, defines a probability distribution on $K(\tau)$, the set of all keys which are consistent with τ. This distribution is known to Eve. This is due to the assumption that Eve has unbounded computational power, i.e., she knows the complete probability space $\Omega(\tau)$ and can compute the corresponding distribution on $K(\tau)$. Thus, Eve can test the hypothesis that the secret key belongs to the set of most probable keys in $K(\tau)$ in the sense of Condition 2 in Section 5.5.

Blue elementary events $\tilde{\omega} = (k_{\tilde{\omega}}, P_{\tilde{\omega}}, F_{\tilde{\omega}})$ will have the property that for $\tau = \tau_{\tilde{\omega}}$ it holds that $\Pr_{\Omega(\tau)}[k_{\tilde{\omega}}]$ is large, i.e., if Alice chooses a blue elementary event, then the success probability of Eve will be nontrivially high.

Definition 17 (Blue Elementary Events)

- For all numbers j, $1 \leq j \leq M$, we call an elementary event $\omega \in \Omega$ to be j-blue if ω is j-alive and not black and if

$$|(X(\tau_\omega^{\leq j}) \oplus k_\omega) \cap U(\tau_\omega^{\leq j})| > \Delta$$

or

$$|(\bar{Y}(\tau_\omega^{\leq j}) \oplus k_\omega) \cap V(\tau_\omega^{\leq j})| > \Delta.$$

- Let Ω_{blue}^j denote the set of all j-blue elementary events.

[2] For nonempty subsets $A, B \subseteq \{0,1\}^n$, we denote by $A \oplus B$ the set $\{a \oplus b; a \in A, b \in B\}$.

– Let $\Omega^{\text{blue}} = \bigcup_{j=1}^{M} \Omega^{j}_{\text{blue}}$.

Definition 18 (Green Elementary Events and Transcipts)

– For all numbers j, $1 \leq j \leq M$, an elementary event $\omega \in \Omega$ is called to be j-green if ω is j-alive and neither j-blue, nor j-red, nor j-black.
– Let $\Omega^{j}_{\text{green}}$ denote the set of all j-green elementary events.
– Let $\Omega^{\text{green}} = \Omega^{M}_{\text{green}}$.
– For all numbers j, $1 \leq j \leq M$, a transcript $\tau \in \mathcal{T}^{j}$ is called green, if it is neither red nor black.
– Let $\mathcal{T}^{j}_{\text{green}}$ denote the set of green transcripts in $\mathcal{T}^{j}$.

It is important to note the following difference between red and black events on the one side, and green and blue events on the other side. If, for a transcript τ, one elementary event in $\Omega(\tau)$ is black (resp. red), then all elementary events in $\Omega(\tau)$ are black (resp. red), which justifies to define τ to be black (resp. red).

On the other side, if a transcript τ is green, then the elementary events in $\Omega(\tau)$ are either blue or green. This is because blueness of an elementary event $\omega \in \Omega(\tau)$ does not only depend on τ but also on the key-component k_ω.

We will prove Theorem 6 by showing that the probabilities of black, red, and blue elementary events are exponentially small, that the probability of sudden-death events is exponentially small, and that for green transcripts $\tau \in \mathcal{T}^{M}_{\text{green}}$, the probability that Eve publishes a correct (initial value, keystream prefix) pair is exponentially small (see Lemma 2 in Section 5.8).

Therefore, let us take more insight into the structure of $\Omega(\tau)$ for green transcripts τ.

We know that for some green transcript τ the decision if an elementary event $\omega \in \Omega(\tau)$ is green or blue depends only on k_ω. This justifies the following definition.

Definition 19 (Green Keys)

– Let τ denote a green transcript. We call a τ-consistent key $k \in K(\tau)$ to be τ-green if $|(X(\tau) \oplus k) \cap U(\tau)| \leq \Delta$ and $|(\bar{Y}(\tau) \oplus k) \cap V(\tau)| \leq \Delta$, and τ-blue otherwise.
– We denote by $K^{\text{green}}(\tau)$ (resp. $K^{\text{blue}}(\tau)$) the set of all τ-consistent keys which are τ-green (resp. τ-blue).

Note that, by definition, for green transcripts τ it holds:

$$K(\tau) = K^{\text{green}}(\tau) \cup K^{\text{blue}}(\tau).$$

5.8 Basic methods II: Estimating probabilities

Remember that for proving Lemma 2 we have to show the following claims.

(i) It holds that

$$\Pr_{\Omega}\left[\Omega^{\text{s.death}} \setminus \left(\Omega^{\text{black}} \cup \Omega^{\text{red}} \cup \Omega^{\text{blue}}\right)\right] \leq 2^{-(n-1)} \cdot (\Delta + 2) \cdot M.$$

(ii) It holds that $\Pr_{\Omega}[\Omega^{\text{black}}] \leq 34 \cdot 2^{-n}$.

(iii) For all $\tau \in \mathcal{T}^{M}_{\text{green}}$, it holds that

$$\Pr_{\Omega^{\text{green}}(\tau)}[\Omega^{\text{succ}}] \leq 11 \cdot (R + 4n) \cdot M \cdot 2^{-(n-1)}.$$

(iv) It holds that $\Pr_\Omega[\Omega^{\text{red}} \cup \Omega^{\text{blue}}] \leq M \cdot e^{-n}$.

The proofs of parts (i), (ii), (iii), and (iv) will be given in Sections 5.10, 5.12, 5.9, and 5.11, respectively.

All these proofs use the following *Smoothness Lemma*, which shows that for all green transcripts τ, there is a sufficiently large number of green τ-consistent keys and that the probabilities of these green keys do not differ too much.

Lemma 5 (Smoothness Lemma) *For all green transcripts τ, the following is true if n is large enough:*

(I) $|K^{\text{green}}(\tau)| \geq \frac{1}{\sqrt{2}} \cdot 2^n$.

(II) *For all $k, k' \in K^{\text{green}}(\tau)$, it holds that*

$$\Pr_{\Omega^{\text{green}}(\tau)}[k] \leq \sqrt{2} \cdot \Pr_{\Omega^{\text{green}}(\tau)}[k'].$$

Lemma 5 implies the following corollary, which will be an important tool for proving Lemma 2.

Corollary 2 (Main Tool Box) *For all green transcripts τ the following is true:*

(a) *For all $k \in \{0,1\}^n$ it holds*

$$\Pr_{\Omega^{\text{green}}(\tau)}[k] \leq 2^{-(n-1)}.$$

(b) *For all $x, \bar{y} \in \{0,1\}^n$ the following holds:*

b.1 *If $(x, \bar{y}) \in Coll(\tau)$ then*

$$\Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y}] = 1.$$

b.2 *If $(x, \bar{y}) \notin Coll(\tau)$ but $x \in X^*(\tau)$ or $\bar{y} \in \bar{Y}^*(\tau)$ then*

$$\Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y}] = 0.$$

b.3 *If $x \in X(\tau) \setminus X^*(\tau)$ and $\bar{y} \in \bar{Y} \setminus \bar{Y}^*(\tau)$ then*

$$\Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y}] = 0.$$

b.4 *In all other cases, i.e., if $x \notin X^*(\tau)$ and $\bar{y} \notin \bar{Y}^*(\tau)$ and ($x \notin X(\tau)$ or $\bar{y} \notin \bar{Y}(\tau)$) it holds*

$$\Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y}] \leq 9 \cdot 2^{-(n-1)}.$$

(c) *For all $x, x' \in \{0,1\}^n$, where $x \notin X(\tau)$ and $x' \in X(\tau)$, and all r, $-(R+n) \leq r \leq R+n$, it holds*

$$\Pr_{\Omega^{\text{green}}(\tau)}\left[\pi^r\left(P_\omega(x \oplus k_\omega) \oplus k_\omega\right) = P_\omega(x' \oplus k_\omega) \oplus k_\omega\right] \leq 11 \cdot 2^{-(n-1)}.$$

Note here that part (a) of Corollary 2 follows directly from Lemma 5. Parts (b.1), (b.2), (b.3) follow directly from the definition of τ-consistent keys. Parts (b.4) and (c) will be proved in Appendix B.

In the following, we prove part (I) of Lemma 5. The proof of part (II) is quite technical and long, thus it was shifted to Appendix C.

Proof of Part (I) of Lemma 5 We fix some number j, $1 \le j \le M$, and some transcript $\tau \in \mathcal{T}^j_{\text{green}}$.

By Lemma 4 it holds that

$$|K^{\text{green}}(\tau)| = |K(\tau)| - |K^{\text{blue}}(\tau)| \ge 2^n - B(M,R,n) - 2\cdot\Delta\cdot j - |K^{\text{blue}}(\tau)|.$$

We show that

$$|K^{\text{blue}}(\tau)| \le \frac{(R+n)\cdot j^2}{\Delta}.$$

This is because

$$\begin{aligned}\sum_{k\in\{0,1\}^n} |(X(\tau)\oplus k)\cap U(\tau)| &= \sum_{k\in\{0,1\}^n} |\{(x,u)\in X(\tau)\times U(\tau);\ x\oplus u = k\}| \\ &= |X(\tau)\times U(\tau)| = |X(\tau)|\cdot|U(\tau)|,\end{aligned}$$

which implies that

$$\left|\left\{k\in\{0,1\}^n;\ |(X(\tau)\oplus k)\cap U(\tau)| > \Delta\right\}\right| \le \frac{|X(\tau)|\cdot|U(\tau)|}{\Delta} \le \frac{j^2}{\Delta}.$$

In exactly the same way one can prove that

$$\begin{aligned}\left|\left\{k\in\{0,1\}^n;\ \left|\left(\bar{Y}(\tau)\oplus k\right)\cap V(\tau)\right| > \Delta\right\}\right| &\le \frac{|\bar{Y}(\tau)|\cdot|V(\tau)|}{\Delta} \\ &\le \frac{(R+n-1)\cdot j^2}{\Delta}.\end{aligned}$$

Consequently,

$$|K^{\text{green}}(\tau)| \ge 2^n - B(M,R,n) - 2\cdot\Delta\cdot j - \frac{(R+n)j^2}{\Delta} \ge \frac{1}{\sqrt{2}}\cdot 2^n$$

if n is large enough. The last inequation follows from Theorem 6, (1). □

5.9 Bounding the probability of sudden death (part (i) of Lemma 2)

In this subsection, we prove part (i) of Lemma 2, namely that

$$\Pr_{\Omega}\left[\Omega^{\text{s.death}} \setminus \left(\Omega^{\text{black}}\cup\Omega^{\text{red}}\cup\Omega^{\text{blue}}\right)\right] \le 2^{-(n-1)}\cdot(\Delta+2)\cdot M.$$

Let us denote by $\mathcal{T}^{\text{all}}_{\text{green}} = \bigcup_{j=1}^{M}\mathcal{T}^j_{\text{green}}$ the set of all green transcripts which occur with nonzero probability. Note that $\mathcal{T}^{\text{all}}_{\text{green}}$ has the structure of a partially ordered set, where a transcript τ is *smaller* than τ' if τ is a prefix of τ'.

We denote by $\mathcal{T}^*_{\text{green}}$ the set of maximal elements in this partially ordered set $\mathcal{T}^{\text{all}}_{\text{green}}$. Observe that

$$\mathcal{T}^*_{\text{green}} = \mathcal{T}^M_{\text{green}} \cup \mathcal{T}^{\text{strange}}_{\text{green}},$$

where $\mathcal{T}^{\text{strange}}_{\text{green}}$ contains all green transcripts τ of length smaller than M for which all transcripts τ' which contain τ as a prefix are black or red.

Let us denote by $\tilde{\Omega}$ the set

$$\tilde{\Omega} = \left(\Omega^{\text{s.death}} \setminus \left(\Omega^{\text{black}}\cup\Omega^{\text{red}}\cup\Omega^{\text{blue}}\right)\right) \cup \bigcup_{\tau\in\mathcal{T}^*_{\text{green}}} \Omega^{\text{green}}(\tau),$$

where for all j, $1 \leq j \leq M$, and all $\tau \in \mathcal{T}^{j}_{\text{green}}$ it holds

$$\Omega^{\text{green}}(\tau) = \{\omega \in \Omega^{j}_{\text{green}};\, \tau^{\leq j}_{\omega} = \tau\}.$$

For all $\tau \in \mathcal{T}^{\text{all}}_{\text{green}}$, we denote by $\Omega^{\text{s.death}}_{\text{green}}(\tau)$ the set of all elementary events $\omega \in \Omega^{\text{green}}(\tau)$ for which the next query after τ generates a sudden-death pair w.r.t. ω. Note that $\Omega^{\text{s.death}}_{\text{green}}(\tau) = \emptyset$ if the length of τ is M.

Observe that $\omega \in \Omega^{\text{s.death}} \setminus \left(\Omega^{\text{black}} \cup \Omega^{\text{red}} \cup \Omega^{\text{blue}}\right)$ if and only if there is some $\tau \in \mathcal{T}^{\text{all}}_{\text{green}}$ such that $\omega \in \Omega^{\text{s.death}}_{\text{green}}(\tau)$. Consequently,

$$\begin{aligned}&\Pr_{\Omega}\left[\Omega^{\text{s.death}} \setminus \left(\Omega^{\text{black}} \cup \Omega^{\text{red}} \cup \Omega^{\text{blue}}\right)\right]\\ \leq\ &\Pr_{\tilde{\Omega}}\left[\Omega^{\text{s.death}} \setminus \left(\Omega^{\text{black}} \cup \Omega^{\text{red}} \cup \Omega^{\text{blue}}\right)\right] = \sum_{\tau \in \mathcal{T}^{\text{all}}_{\text{green}}} \Pr_{\tilde{\Omega}}[\Omega^{\text{s.death}}_{\text{green}}(\tau)].\end{aligned}$$

We fix for all transcripts $\tau \in \mathcal{T}^{*}_{\text{green}}$ a natural number $i(\tau)$, $1 \leq i(\tau) \leq j$, where j denotes the length of τ. We do this in such a way that the sets

$$T(\tau) = \{\tau^{\leq i(\tau)}, \tau^{\leq i(\tau)+1}, \cdots, \tau^{\leq j}\}$$

form a partition of the set $\mathcal{T}^{\text{all}}_{\text{green}}$ into pairwise disjoints subsets.[3] Note that the sets $T(\tau)$ correspond to prefixes of the transcript τ.

Now define for all transcripts $\tau \in \mathcal{T}^{*}_{\text{green}}$ subsets $A(\tau)$ and $B(\tau)$ of $\tilde{\Omega}$:

$$A(\tau) = \bigcup_{\tilde{\tau} \in T(\tau)} \Omega^{\text{s.death}}_{\text{green}}(\tilde{\tau}),$$

$$B(\tau) = \Omega^{\text{green}}(\tau) \cup A(\tau).$$

Note that the set system $\{B(\tau);\, \tau \in \mathcal{T}^{*}_{\text{green}}\}$ defines a partition of $\tilde{\Omega}$ into pairwise disjoint subsets, that the set system $\{A(\tau);\, \tau \in \mathcal{T}^{*}_{\text{green}}\}$ defines a partition of the set $\Omega^{\text{s.death}} \setminus \left(\Omega^{\text{black}} \cup \Omega^{\text{red}} \cup \Omega^{\text{blue}}\right)$ into pairwise disjoint subsets, and that for all $\tau \in \mathcal{T}^{*}_{\text{green}}$ it holds $A(\tau) \subseteq B(\tau)$. Consequently,

$$\begin{aligned}\Pr_{\tilde{\Omega}}\left[\Omega^{\text{s.death}} \setminus \left(\Omega^{\text{black}} \cup \Omega^{\text{red}} \cup \Omega^{\text{blue}}\right)\right] &= \sum_{\tau \in \mathcal{T}^{*}_{\text{green}}} \Pr_{\tilde{\Omega}}[A(\tau) \cap B(\tau)]\\ &= \sum_{\tau \in \mathcal{T}^{*}_{\text{green}}} \Pr_{\tilde{\Omega}}[B(\tau)] \cdot \Pr_{B(\tau)}[A(\tau)]\\ &\leq \max_{\tau \in \mathcal{T}^{*}_{\text{green}}} \Pr_{B(\tau)}[A(\tau)]. \qquad (18)\end{aligned}$$

We fix some arbitrary $\tau \in \mathcal{T}^{*}_{\text{green}}$ and denote by j the length of τ. Note that for all transcripts $\tilde{\tau} \in T(\tau)$ it holds that $\omega \in \Omega^{\text{s.death}}_{\text{green}}(\tilde{\tau})$ if and only if $\omega \in \Omega^{\text{green}}(\tilde{\tau})$ and the key k_{ω} falls into the set $D(\tilde{\tau})$, which is defined as follows:

$$D(\tilde{\tau}) = \left(X^{*}_{\text{new}}(\tilde{\tau}) \oplus U_{\text{new}}(\tilde{\tau})\right) \setminus \left(X^{*}(\tilde{\tau}) \oplus U(\tilde{\tau})\right),$$

where $X^{*}_{\text{new}}(\tilde{\tau})$ and $U_{\text{new}}(\tilde{\tau})$ denote the new sets $X^{*}(\cdot)$ and $U(\cdot)$ after posing the uniquely determined next query after $\tilde{\tau}$.

[3] One way of constructing the numbers $i(\tau)$ is as follows. We enumerate the transcripts in $\mathcal{T}^{*}_{\text{green}}$, take the first transcript τ, set $i(\tau) = 1$, and label all transcripts $\tau^{\leq s}$, for $s = i(\tau), \cdots, j$, where j denotes the length of τ. For all other transcripts in $\tau \in \mathcal{T}^{*}_{\text{green}}$, define $i(\tau)$ to be the smallest number i for which $\tau^{\leq i}$ has not been labeled so far and label all transcripts in the corresponding set $T(\tau)$.

According to Corollary 2, part (a), the probability of this event is bounded by $2^{-(n-1)} \cdot |D(\tilde{\tau})|$.

Now observe that $\bigcup_{\tilde{\tau} \in T(\tau)} D(\tilde{\tau})$ is a subset of $X^*_{\text{new}}(\tau) \oplus U_{\text{new}}(\tau)$ if $j < M$ and of $X^*(\tau) \oplus U(\tau)$ if $j = M$.

If $j < M$, then $|X^*_{\text{new}}(\tau)| \leq |X^*(\tau)| + 2 \leq \Delta + 2$, as τ is green, and $|U_{\text{new}}(\tau)| \leq |U(\tau)| + 1 \leq M$. We obtain that

$$\Pr_{B(\tau)}[A(\tau)] \leq 2^{-(n-1)} \cdot |X^*_{\text{new}}(\tau) \oplus U_{\text{new}}(\tau)| \leq 2^{-(n-1)} \cdot (\Delta + 2) \cdot M,$$

which proves part (i) of Lemma 2 by Relation (18).

5.10 Bounding the probability of black transcripts (part (ii) of Lemma 2)

In this subsection, we prove part (ii) of Lemma 2, namely that

$$\Pr_{\Omega}\left[\Omega^{\text{black}}\right] \leq 34 \cdot 2^{-n}. \tag{19}$$

Proof of Relation (19) From Definition 15, it follows straightforwardly that for any elementary event $\omega \in \Omega$, it holds that the transcript τ_ω is black if and only if it has some black prefix (where τ_ω is considered to be its own prefix). This, in turn, implies that $\omega \in \Omega^{\text{black}}$ if and only if τ_ω is black. Consequently, it is sufficient here to assess the probability that for an $\omega \in \Omega$ chosen uniformly and at random (see Definition 4), the number of τ_ω-critical keys exceeds $B(M, R, n)$ or it holds $Max(P_{\tau_\omega}, U(\tau_\omega)) > 5$.

Remember from Definition 13 that a key $k \in \{0,1\}^n$ is called τ_ω-critical if there is some $u \in U(\tau_\omega)$ such that $x := u \oplus k \in X(\tau_\omega)$ and $y := P_{\tau_\omega}(u) \oplus k \in \bar{Y}(\tau_\omega)$, which implies that for the corresponding triple (u, x, y) it holds that $x \oplus u = y \oplus P_{\tau_\omega}(u)$. Moreover, remember from Theorem 5 the definition of $\mu(P, U, X, Y)$ for permutations P over $\{0,1\}^n$ and subsets U, X, Y of $\{0,1\}^n$:

$$\mu(P, U, X, Y) = |\{(u, x, y) \in U \times X \times Y; x \oplus u = y \oplus P(u)\}|. \tag{20}$$

Consequently, $\mu(P_{\tau_\omega}, U(\tau_\omega), X(\tau_\omega), \bar{Y}(\tau_\omega))$ is an upper bound for the number of τ_ω-critical keys.

Theorem 5 implies that the probability that for a randomly chosen $\omega \in \Omega$, it holds that

$$\mu(P_{\tau_\omega}, U(\tau_\omega), X(\tau_\omega), \bar{Y}(\tau_\omega)) \geq B(M, R, n),$$

is at most $2 \cdot 2^{-n}$. Here, we took into account that $|U(\tau_\omega)| \leq M$, $|X(\tau_\omega)| \leq M$, and $|\bar{Y}(\tau_\omega)| \leq M \cdot (R + n - 1)$.

So, the probability that for a randomly chosen $\omega \in \Omega$, ω falls into Ω^{black} because the number of τ_ω-critical keys exceeds $B(M, R, n)$, is bounded from above by $2 \cdot 2^{-n}$.

We complete the proof by showing that

$$\Pr_{\Omega}\left[Max\left(P_{\tau_\omega}, U(\tau_\omega)\right) \geq 6\right] \leq 32 \cdot 2^{-n}.$$

According to Definition 12, the event $Max(P_{\tau_\omega}, U(\tau_\omega)) \geq 6$ implies the existence of some $U' \subseteq U(\tau_\omega)$, $|U'| = 6$, such that $u'_1 \oplus P_{\tau_\omega}(u'_1) = u'_2 \oplus P_{\tau_\omega}(u'_2)$ for all $u'_1, u'_2 \in U'$. Given a subset $U' \subseteq U(\tau_\omega)$, $|U'| = 6$, the probability that $u'_1 \oplus P_{\tau_\omega}(u'_1) = u'_2 \oplus P_{\tau_\omega}(u'_2)$ holds for all $u'_1, u'_2 \in U'$, equals

$$\prod_{i=1}^{5} \frac{1}{2^n - i} \leq \left(\frac{1}{1/2 \cdot 2^n}\right)^5 = 2^5 \cdot 2^{-5 \cdot n}.$$

Consequently,

$$\Pr_{\Omega}\left[Max\left(P_{\tau_\omega}, U(\tau_\omega)\right) \geq 6\right] \leq |U(\tau_\omega)|^6 \cdot 2^5 \cdot 2^{-5\cdot n} \leq 2^{6\cdot(2/3)n} \cdot 2^5 \cdot 2^{-5\cdot n} = 32 \cdot 2^{-n}.$$

Here, for the sake of simplicity, we upper bounded the number of subsets with six elements of $U(\tau_\omega)$ by $|U(\tau_\omega)|^6$. $|U(\tau_\omega)|$, in turn, is upper bounded by $2^{(2/3)n}$ as the underlying transcript τ_ω consists of at most $2^{(2/3)n}$ queries. □

5.11 Bounding the success probability on green elementary events (part (iii) of Lemma 2)

Let τ be a green transcript of length M, i.e., $\tau \in \mathcal{T}^M_{\text{green}}$. We have to bound the probability that Eve is successful under the condition that Alice has chosen a green elementary event $\omega = (k_\omega, P_\omega, F_\omega) \in \Omega^{\text{green}}(\tau)$.

Depending on τ, Eve publishes a pair $(x^*(\tau), z^*(\tau)) \in \{0,1\}^n \times \{0,1\}^n$, where $x^*(\tau) \notin X(\tau)$. Eve wins if and only if $z^*(\tau)$ equals the block of the first n keystream bits of the packet generated on input $x^*(\tau)$ under ω, i.e.,

$$z^*(\tau) = F_\omega(P_\omega(x^*(\tau) \oplus k_\omega) \oplus k_\omega).$$

For all $\omega \in \Omega^{\text{green}}(\tau)$, let y_ω denote the value

$$y_\omega = P_\omega(x^*(\tau) \oplus k_\omega) \oplus k_\omega.$$

We have to bound the probability

$$\Pr_{\Omega^{\text{green}}(\tau)}[F_\omega(y_\omega) = z^*(\tau)].$$

We do this by dividing $\Omega^{\text{green}}(\tau)$ into two disjoint subsets *IND* and *DEP*, where *IND* contains all those elementary events $\omega \in \Omega^{\text{green}}(\tau)$ for which $F_\omega(y_\omega)$ is independent from the queries and answers contained in τ, and $DEP = \Omega^{\text{green}}(\tau) \setminus IND$.

Note that $\omega \in DEP$ if and only if

(I) there is some i, $-(n-1) \leq i \leq n-1$, such that $\pi^i(y_\omega) \in Y(\tau)$, or
(II) there is some i, $-(n-1) \leq i \leq n-1$, some $x \in X(\tau)$, and some r, $0 \leq r \leq R-1$, such that $\pi^i(y_\omega) = \pi^r(P_\omega(x \oplus k_\omega) \oplus k_\omega)$.

In case (I), $F_\omega(y_\omega)$ is not independent from the answer of the F-query with input $\pi^i(y_\omega)$; in case (II), $F_\omega(y_\omega)$ is not independent from the answer of the E-query with input x (in particular, from the block starting at position r in packet $E_\omega(x)$).

Corresponding to this, *DEP* can be written as

$$DEP = DEP_1 \cup DEP_2,$$

where DEP_1 contains all $\omega \in \Omega^{\text{green}}(\tau)$ for which case (I) is fulfilled and DEP_2 contains all $\omega \in \Omega^{\text{green}}(\tau)$ for which case (II) is fulfilled.

Note that

$$\begin{aligned}\Pr_{\Omega^{\text{green}}(\tau)}[F_\omega(y_\omega) = z^*(\tau)] &= \Pr_{\Omega^{\text{green}}(\tau)}[DEP] \cdot \Pr_{\Omega^{\text{green}}(\tau)}[F_\omega(y_\omega) = z^*(\tau) \mid DEP] \\ &\quad + \Pr_{\Omega^{\text{green}}(\tau)}[IND] \cdot \Pr_{\Omega^{\text{green}}(\tau)}[F_\omega(y_\omega) = z^*(\tau) \mid IND] \\ &\leq \Pr_{\Omega^{\text{green}}(\tau)}[DEP] + \Pr_{\Omega^{\text{green}}(\tau)}[F_\omega(y_\omega) = z^*(\tau) \mid IND],\end{aligned}$$

i.e.,

$$\Pr_{\Omega^{\text{green}}(\tau)}[F_\omega(y_\omega) = z^*(\tau)] \le \Pr_{\Omega^{\text{green}}(\tau)}[DEP_1] + \Pr_{\Omega^{\text{green}}(\tau)}[DEP_2] + \Pr_{\Omega^{\text{green}}(\tau)}[F_\omega(y_\omega) = z^*(\tau) \mid IND]. \tag{21}$$

It is quite obvious that

$$\Pr_{\Omega^{\text{green}}(\tau)}[F_\omega(y_\omega) = z^*(\tau) \mid IND] = 2^{-n}, \tag{22}$$

as $\omega \in IND$ implies that $F_\omega(y_\omega)$ can take all values in $\{0,1\}^n$ with the same probability.

Next observe that for all $\omega \in \Omega^{\text{green}}(\tau)$ it holds that $\omega \in DEP_1$ if and only if

$$y_\omega \in \bigcup_{y \in Y(\tau)} \{\pi^i(y); -(n-1) \le i \le n-1\},$$

where the set at the right hand side has size at most $(2n-1)M$.

As $x^*(\tau) \notin X(\tau)$, it follows by Corollary 2, part (b), that

$$\Pr_{\Omega^{\text{green}}(\tau)}[DEP_1] \le (2n-1) \cdot M \cdot 9 \cdot 2^{-(n-1)}. \tag{23}$$

Observe further that for all $\omega \in \Omega^{\text{green}}(\tau)$ it holds that $\omega \in DEP_2$ if and only if

$$\pi^i\left(P_\omega(x^*(\tau) \oplus k_\omega) \oplus k_\omega\right) = P_\omega(x \oplus k_\omega) \oplus k_\omega$$

for some $x \in X(\tau)$ and number i, $-(R+n-2) \le i \le n-1$.

As $x^*(\tau) \notin X(\tau)$, it follows by Corollary 2, part (c), that

$$\Pr_{\Omega^{\text{green}}(\tau)}[DEP_2] \le (R+2n-2) \cdot M \cdot 11 \cdot 2^{-(n-1)}. \tag{24}$$

Putting relations (21), (22), (23), and (24) together yields

$$\begin{aligned}\Pr_{\Omega^{\text{green}}(\tau)}[\Omega^{\text{succ}}] &\le (2 + (2n-1) \cdot M \cdot 9 + (R+2n-2) \cdot M \cdot 11) \cdot 2^{-(n-1)} \\ &< 11 \cdot (R+4n) \cdot M \cdot 2^{-(n-1)}.\end{aligned}$$

□

5.12 Bounding the probability of red and blue transcripts (part (iv) of Lemma 2)

We have to show that

$$\Pr_{\Omega}\left[\Omega^{\text{red}} \cup \Omega^{\text{blue}}\right] \le M \cdot e^{-n}. \tag{25}$$

In the proof, we will use a Chernoff bound argument, which is described in Appendix A.

Proof of Relation (25) Note first that for all $\omega \in \Omega^{\text{red}} \cup \Omega^{\text{blue}}$, there is some j, $1 \le j \le M$, such that the j-th query makes ω red or blue. Consequently,

$$\Omega^{\text{red}} \cup \Omega^{\text{blue}} = \bigcup_{j=1}^{M} \Omega^{j-1}_{\text{green}} \cap \left(\Omega^j_{\text{red}} \cup \Omega^j_{\text{blue}}\right),$$

which implies

$$\begin{aligned}\Pr_{\Omega}\left[\Omega^{\text{red}} \cup \Omega^{\text{blue}}\right] &\leq \sum_{j=1}^{M} \Pr_{\Omega}\left[\Omega_{\text{green}}^{j-1} \cap \left(\Omega_{\text{red}}^{j} \cup \Omega_{\text{blue}}^{j}\right)\right] \\ &= \sum_{j=1}^{M} \Pr_{\Omega}\left[\Omega_{\text{red}}^{j} \cup \Omega_{\text{blue}}^{j} \mid \Omega_{\text{green}}^{j-1}\right] \cdot \Pr_{\Omega}\left[\Omega_{\text{green}}^{j-1}\right] \\ &\leq \sum_{j=1}^{M} \Pr_{\Omega}\left[\Omega_{\text{red}}^{j} \cup \Omega_{\text{blue}}^{j} \mid \Omega_{\text{green}}^{j-1}\right].\end{aligned}$$

Hence, for proving Relation (25), it is sufficient to show that for all j, $1 \leq j \leq M$, it holds

$$\Pr_{\Omega_{\text{green}}^{j-1}}\left[\Omega_{\text{red}}^{j} \cup \Omega_{\text{blue}}^{j}\right] = \Pr_{\Omega}\left[\Omega_{\text{red}}^{j} \cup \Omega_{\text{blue}}^{j} \mid \Omega_{\text{green}}^{j-1}\right] < e^{-n}. \tag{26}$$

We show Relation (26): Note first that Relation (26) is true if $j < \frac{\Delta}{R+n-1}$, as then for all transcripts τ with j queries it holds that the cardinalities of $X(\tau)$ and $\bar{Y}(\tau)$ are smaller than Δ.

We fix some arbitrary number j, $\frac{\Delta}{R+n-1} \leq j \leq M$.

For all J, $1 \leq J \leq j-1$, we define a random variable $DB_J \in \{0,1\}$ over Ω, where $DB_J(\omega) = 1$ if and only if ω is J-alive and the J-th query along τ_ω increases $(X(\tau_\omega) \oplus k_\omega) \cap U(\tau_\omega)$ or increases $(\bar{Y}(\tau_\omega) \oplus k_\omega) \cap V(\tau_\omega)$ or increases $X^*(\tau_\omega)$. Formally,

$$\begin{aligned}&DB_J(\omega) = 1 \iff \\ &|(X(\tau_\omega^{\leq J}) \oplus k_\omega) \cap U(\tau_\omega^{\leq J})| > |(X(\tau_\omega^{\leq J-1}) \oplus k_\omega) \cap U(\tau_\omega^{\leq J-1})| \text{ or} \\ &|(\bar{Y}(\tau_\omega^{\leq J}) \oplus k_\omega) \cap V(\tau_\omega^{\leq J})| > |(\bar{Y}(\tau_\omega^{\leq J-1}) \oplus k_\omega) \cap V(\tau_\omega^{\leq J-1})| \text{ or} \\ &|X^*(\tau_\omega^{\leq J})| > |X^*(\tau_\omega^{\leq J-1})|.\end{aligned}$$

Note that the event $\omega \in \Omega_{\text{red}}^{j} \cap \Omega_{\text{blue}}^{j}$ implies the event that

$$\sum_{J=1}^{j-1} DB_J(\omega) \geq \frac{\Delta - (R+n-1)}{R+n-1}. \tag{27}$$

This is because each query along τ_ω increases $(X(\tau_\omega) \oplus k_\omega) \cap U(\tau_\omega)$ by at most one and $(\bar{Y}(\tau_\omega) \oplus k_\omega) \cap V(\tau_\omega)$ by at most $R+n-1$ and $X^*(\tau_\omega)$ by at most two.

In particular, each E-query can increases $(X(\tau_\omega) \oplus k_\omega) \cap U(\tau_\omega)$ by at most one and $X^*(\tau_\omega)$ by at most two, each P- or P^{-1}-query can increase $(X(\tau_\omega) \oplus k_\omega) \cap U(\tau_\omega)$ and $(\bar{Y}(\tau_\omega) \oplus k_\omega) \cap V(\tau_\omega)$ by at most one, and each F-query can increase $(\bar{Y}(\tau_\omega) \oplus k_\omega) \cap V(\tau_\omega)$ by at most $R+n-1$ and $X^*(\tau_\omega)$ by at most one.

We bound the probability of the event in Relation (27) over $\Omega_{\text{green}}^{j-1}$. We do this by bounding the probability of the event $DB_J(\omega) = 1$ over $\Omega_{\text{green}}^{j-1}$ for all $J = 1, \cdots, j-1$. Let us fix a number J, $1 \leq J \leq j-1$.

Note that

$$\Pr_{\Omega_{\text{green}}^{j-1}}[DB_J(\omega) = 1] = \sum_{\tau \in \mathcal{T}_{\text{green}}^{J}} \Pr_{\Omega_{\text{green}}^{j-1}}[\tau] \cdot \Pr_{\Omega_{\text{green}}^{j-1}(\tau)}[DB_J(\omega) = 1].$$

Note further that for all $\tau \in \mathcal{T}_{\text{green}}^{J}$ and $\omega \in \Omega_{\text{green}}^{j-1}(\tau)$ it holds that $DB_J(\omega) = 1$ if and only if at least one of the following conditions is fulfilled.

(A) The J-th query in τ is a P-query with input u or a P^{-1}-query with output u and $k_\omega \in u \oplus X(\tau)$.
(B) The J-th query in τ is a P-query with output v or a P^{-1}-query with input v and $k_\omega \in v \oplus \bar{Y}(\tau)$.
(C) The J-th query in τ is an F-query with input y and there is some r, $-(n-1) \leq r \leq R-1$, such that $k_\omega \in \pi^{-r}(y) \oplus V(\tau)$.
(D) The J-th query in τ is an E-query with input x and $k_\omega \in x \oplus U(\tau)$.
(E) The J-th query in τ is an E-query with input x and $P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y}$ for some $\bar{y} \in \bar{Y}(\tau)$.
(F) The J-th query in τ is an F-query with input y and $y = \pi^r(P_\omega(x \oplus k_\omega) \oplus k_\omega)$ for some $x \in X(\tau) \setminus X^*(\tau)$ and some number r, $-(n-1) \leq r \leq R-1$.
(G) The J-th query in τ is an E-query with input x and there is some r, $-(R-1) \leq r \leq R-1$, and some $x' \in X(\tau)$ such that $\pi^r(P_\omega(x \oplus k_\omega) \oplus k_\omega) = P_\omega(x' \oplus k_\omega) \oplus k_\omega$.

Note that (A) and (D) are the situations in which query J increases $(X(\tau_\omega) \oplus k_\omega) \cap U(\tau_\omega)$, that (B) and (C) are the situations in which query J increases $(\bar{Y}(\tau_\omega) \oplus k_\omega) \cap V(\tau_\omega)$, that (E) and (F) are the situations in which query J generates a new structural EF-collision (i.e., increases $X^*(\tau_\omega)$ by one), and that (G) is the situation in which query J generates a new structural EE-collision (i.e., increases $X^*(\tau_\omega)$ by one or two).

Note further that conditions (A,D) imply that k_ω belongs to a set of at most $J-1$ elements. Conditions (B,C) imply that k_ω belongs to a set of at most $(R+n-1)\cdot(J-1)$ elements. From Corollary 2, part (a), it follows that these events have probability at most $2^{-(n-1)} \cdot (R+n-1) \cdot (J-1)$.

From Corollary 2, part (b), it follows that condition (E) has probability at most $9 \cdot |\bar{Y}(\tau_\omega)| \cdot 2^{-(n-1)} \leq 9 \cdot (R+n-1) \cdot (J-1) \cdot 2^{-(n-1)}$, and that condition (F) has probability at most $9 \cdot (R+n-1) \cdot |X(\tau_\omega)| \cdot 2^{-(n-1)} \leq 9 \cdot (R+n-1) \cdot (J-1) \cdot 2^{-(n-1)}$.

From Corollary 2, part (c), it follows that condition (G) has probability at most $11 \cdot (2R-1) \cdot |X(\tau_\omega)| \cdot 2^{-(n-1)} \leq 11 \cdot (2R-1) \cdot (J-1) \cdot 2^{-(n-1)}$.

We obtain that for all J, $1 \leq J \leq j-1$,

$$\begin{aligned} \Pr_{\Omega_{\text{green}}^{j-1}}[DB_J(\omega) = 1] &\leq 11 \cdot 2^{-(n-1)} \cdot (2R-1) \cdot (J-1) \\ &< 22 \cdot 2^{-(n-1)} \cdot R \cdot (j-1). \end{aligned} \tag{28}$$

Relation (28) now enables us to apply the Chernoff bound method from Lemma 7 in Appendix A with $N = j-1$, $p = 22 \cdot 2^{-(n-1)} \cdot R \cdot (j-1)$ and $D = n$, and to obtain directly that

$$\Pr_{\Omega_{\text{green}}^{j-1}}\left[\sum_{J=1}^{j-1} DB_J(\omega) > 22 \cdot 2^{-(n-1)} \cdot R \cdot (j-1)^2 + \sqrt{\frac{n \cdot (j-1)}{2}}\right] < e^{-n}. \tag{29}$$

Note that item (2) of Theorem 6 yields that

$$\begin{aligned} &22 \cdot 2^{-(n-1)} \cdot R \cdot (j-1)^2 + \sqrt{\frac{n \cdot (j-1)}{2}} \\ &< 22 \cdot 2^{-(n-1)} \cdot R \cdot M^2 + \sqrt{\frac{n \cdot M}{2}} \\ &\leq \frac{\Delta - (R+n-1)}{R+n-1}. \end{aligned} \tag{30}$$

Thus, Relation (29) together with Relation (30) proves relations (26) and (25), and, consequently, Lemma 2, part (iv). □

6 Conclusion

In this paper, we introduced for the first time a random oracle model for KSG-based stream ciphers and proved a sharp asymptotic $(2/3)n$ bound on the security of the LIZARD-construction, which underlies the stream cipher LIZARD [22], against generic chosen-IV key recovery and packet prediction TMD tradeoff attacks. We hope that the security model and the lower bound techniques developed in this paper help to prove similar sharp security bounds for other stream cipher constructions such as the concatenation method underlying the state initialization of Trivium and Grain (see relations (3) and (4)). We have further shown that for a packet length $R > n$, where n denotes the inner state length of the underlying KSG, KSG-based stream ciphers can be only $n/2$-secure w.r.t. generic TMD tradeoff distinguishing attacks (see Corollary 2). From a theoretical point of view, it would be interesting to analyze the case $R = n$. Our conjecture is that for $R = n$, the LIZARD-construction is $(2/3)n$-secure even against distinguishing attacks.

Appendix A: Basic Methods III: A Short Excursion to Chernoff Bounds

At several places of our proof, we have to apply a technique called *Chernoff bounds* in the literature. The basic Chernoff bound argument is the following.

Theorem 7 *Let N be a positive integer, $p \in (0, 1)$, and $A_1, \cdots, A_N$ be a set of mutually independent random variables, where, for all $i = 1, \cdots, N$, it holds that $\Pr[A_i = 1 - p] = p$ and $\Pr[A_i = -p] = 1 - p$. Let $A = \sum_{i=1}^{N} A_i$. Then*

$$\Pr[A > a] < e^{-2 \cdot a^2 / N}$$

for all $a > 0$.

For a **proof** see, e.g., Alon, Spencer, Erdos, The Probabilistic Method, Wiley Interscience 1992, Theorem A4 on page 235 [1].

We derive from Theorem 7 a corresponding result for random $\{0, 1\}$-variables.

Lemma 6 *Let p, N, and A_i for $i = 1, \cdots, N$ be defined as in Theorem 7, and let $B_i = A_i + p$. Note that $B_i \in \{0, 1\}$ and $\Pr[B_i = 1] = p$. Let $B = \sum_{i=1}^{N} B_i$. Then, for all $d > 0$, it holds*

$$\Pr[B > (p + d)N] < e^{-2 \cdot d^2 \cdot N}.$$

Proof of Lemma 6 By definition, $B = A + N \cdot p$. The proof is completed by putting $a = d \cdot N$ into the relation in Theorem 7. □

We will apply Chernoff bound arguments in the following modified scenario. Before doing this we introduce a denotation.

Definition 20 Let $N \geq 1$ and $X_1, \cdots, X_N$ denote a collection of random $\{0,1\}$-variables. For all i, $1 \leq i \leq N$, and for all $b = (b_1, \cdots, b_i) \in \{0,1\}^i$ let $X(b)$ denote the event that $X_j = b_j$ for all $j = 1, \cdots, i$.

Lemma 7 *Let $C_1, \cdots, C_N$ denote a collection of random $\{0,1\}$-variables fulfilling the following two conditions for some probability bound p, $0 < p < 1$:*

(a) $\Pr[C_1 = 1] = p_1 < p$.

(b) *For all i, $2 \leq i \leq N$, and all $b \in \{0,1\}^{i-1}$, there is some number $p(b) < p$, which can be computed from b, and for which it holds that*

$$\Pr[C_i = 1 \mid C(b)] = p(b).$$

Let $C = \sum_{i=1}^{N} C_i$. Then, for all $d > 0$, it holds

$$\Pr[C > (p+d)N] < e^{-2 \cdot d^2 \cdot N}. \tag{31}$$

Comment 1 Note that $C_1, \cdots, C_N$ are allowed to be statistically dependent. However, conditions (a) and (b) imply that only a weak sort of dependence is allowed.

For illustrating this, observe that for satisfying Relation (31) it is not sufficient to require condition

(a') $\Pr[C_j = 1] < p$ for all $j = 1, \cdots, N$.

A counterexample is given by the case that $C_1 = C_2 = \cdots = C_N$, which satisfies (a') if $\Pr[C_1 = 1] < p$, but which does not satisfies Relation (31).

Condition (a') only implies condition (a) and the condition

(b') For all i, $2 \leq i \leq N$,

$$\sum_{b \in \{0,1\}^{i-1}} \Pr[C(b)] \cdot \Pr[C_i = 1 \mid C(b)] < p.$$

Note that condition (b') is much weaker than condition (b), as condition (b) requires that $\Pr[C_i = 1 \mid C(b)] < p$ holds not only in the average but for all $b \in \{0,1\}^{i-1}$.

Comment 2 At several places we will take $d = \sqrt{D/(2N)}$ and obtain

$$\Pr[C > (p+d)N] = \Pr\left[C > pN + \sqrt{\frac{D \cdot N}{2}}\right] < e^{-D}.$$

Proof of Lemma 7 We construct a collection of mutually independent binary random variables $B_1, \cdots, B_N$ satisfying

- $C_i = 1$ implies $B_i = 1$,
- $\Pr[B_i = 1] = p$

for all i, $1 \leq i \leq N$.

This proves our Lemma 7, as $\sum_{i=1}^{N} C_i \leq \sum_{i=1}^{N} B_i$ with probability one, and as Lemma 6 can be applied to $B = \sum_{i=1}^{N} B_i$.

We first describe the experiment behind the random $\{0,1\}$-variables $C_1, \cdots, C_N$ and $B_1, \cdots, B_N$. Suppose that the experiments behind $C_1, \cdots, C_N$ and $B_1, \cdots, B_N$ are performed by a person named Tom. Tom uses a device with the following input/output behavior: If Tom inputs a number q, $0 \leq q \leq 1$, into the device, then the device outputs a

random bit $D \in \{0, 1\}$ with $\Pr[D = 1] = q$. The device has no memory, i.e., if we contact the device at different times, then the corresponding outputs are mutually independent.

The experiments behind $C_1, \cdots, C_N$ and $B_1, \cdots, B_N$ refer to a probability bound p, $0 < p < 1$, a probability value $p_1 < p$, and an algorithm which assigns for each j, $1 \leq j \leq N-1$, and each $\{0, 1\}$-vector $b \in \{0, 1\}^j$ a probability value $p(b) < p$.

The generation of the random $\{0, 1\}$-values $C_1, \cdots, C_N$ Tom generates C_1 by inputting p_1 into the device and taking the output as C_1. This implies $\Pr[C_1 = 1] = p_1$. Moreover, Tom stores the result C_1 in his personal memory.

Now fix some j, $2 \leq j \leq N$, and suppose that Tom has already generated the random $\{0, 1\}$-values $C_1, \cdots, C_{j-1}$ and stored the outcomes $C_1 = b_1, \cdots, C_{j-1} = b_{j-1}$ in his personal memory. This defines the $\{0, 1\}$-vector $b = (b_1, \cdots, b_{j-1}) \in \{0, 1\}^{j-1}$.

Then Tom computes the probability value $p(b) < p$, inputs $p(b)$ to the device, takes the output as C_j, and stores the result for C_j in his personal memory.

This implies $\Pr[C_j = 1 \mid C(b)] = p(b)$.

Moreover, this implies that

$$\Pr[C_j = 1] = \sum_{b \in \{0,1\}^{j-1}} \Pr[C(b)] \cdot p(b) < p.$$

Note that there may exist $\{0, 1\}$-vectors $b \in \{0, 1\}^{j-1}$ such that $\Pr[C_j = 1] \neq \Pr[C_j = 1 \mid C(b)] = p(b)$, i.e., it may happen that C_j is not statistically independent from $C_1, \cdots, C_{j-1}$.

However, if the $p(b)$-values are equal for all $b \in \{0, 1\}^{j-1}$, then C_j is statistically independent from $C_1, \cdots, C_{j-1}$.

The generation of the random $\{0, 1\}$-values $B_1, \cdots, B_N$ Tom generates B_1 as follows. First, he generates again C_1 by inputting p_1 into the device, taking the output as C_1, and stores C_1 in his personal memory. If $C_1 = 1$, then he defines $B_1 = 1$. If $C_1 = 0$, then Tom inputs the probability value $\frac{p - p_1}{1 - p_1}$ into the device and takes the output as B_1.

Note that

$$\Pr[B_1 = 1] = \Pr[C_1 = 1] + \Pr[C_1 = 0] \cdot \frac{p - p_1}{1 - p_1} = p_1 + (1 - p_1) \frac{p - p_1}{1 - p_1} = p. \quad (32)$$

Now we fix some j, $2 \leq j \leq N$, and describe how Tom generates B_j. Suppose that Tom has already generated the random $\{0, 1\}$-values $C_1, \cdots, C_{j-1}$ and the $\{0, 1\}$-values $B_1, \cdots, B_{j-1}$, and that he has stored the outcomes $C_1 = b_1, \cdots, C_{j-1} = b_{j-1}$, corresponding to the $\{0, 1\}$-vector $b = (b_1, \cdots, b_{j-1}) \in \{0, 1\}^{j-1}$, in his personal memory.

Then Tom first generates C_j by computing the probability value $p(b) < p$, inputting $p(b)$ to the device, taking the output as C_j, and stores the result for C_j in his personal memory.

If $C_j = 1$, then he defines $B_j = 1$.

If $C_j = 0$, then Tom inputs the probability value $\frac{p - p(b)}{1 - p(b)}$ into the device and takes the output as B_j.

By the same argument as in Relation (32), it holds that

$$\begin{aligned} \Pr[B_j = 1 \mid C(b)] &= \Pr[C_j = 1 \mid C(b)] + \Pr[C_j = 0 \mid C(b)] \cdot \frac{p - p(b)}{1 - p(b)} \\ &= p(b) + (1 - p(b)) \frac{p - p(b)}{1 - p(b)} = p. \end{aligned} \quad (33)$$

This implies that

$$\Pr[B_j = 1] = \sum_{b\in\{0,1\}^{j-1}} \Pr[C(b)] \cdot \Pr[B_j = 1 \mid C(b)] = \sum_{b\in\{0,1\}^{j-1}} (\Pr[C(b)] \cdot p)$$
$$= p \cdot \sum_{b\in\{0,1\}^{j-1}} \Pr[C(b)] = p \cdot 1 = p.$$

The proof that $B_1, \cdots, B_N$ are mutually independent For all j, $1 \le j \le N$, and $\{0,1\}$-vectors $b, b' \in \{0,1\}^j$, we write $b' \le b$ if $b'_i \le b_i$ for all $i = 1, \cdots, j$.

The way the random variables B_j and C_j, $1 \le j \le N$, are generated, implies the following two facts:

(1) For all j, $2 \le j \le N$, and $\{0,1\}$-vectors $b \in \{0,1\}^{j-1}$, the event $B(b)$ implies the event $\bigcup_{b'\in\{0,1\}^{j-1}, b'\le b} C(b')$. This implies that

$$\Pr[B(b)] = \sum_{b'\in\{0,1\}^{j-1}, b'\le b} \Pr[B(b) \wedge C(b')].$$

(2) For all j, $2 \le j \le N$, and $\{0,1\}$-vectors $b, b' \in \{0,1\}^{j-1}$ with $b' \le b$, it holds that

$$\Pr[B_j = 1 \mid B(b) \wedge C(b')] = \Pr[B_j = 1 \mid C(b')] = p.$$

Item (2) follows from the fact that the behavior of Tom when generating B_j depends only on b'.

For showing the mutual independence of $B_1, \cdots, B_N$, it is sufficient to show that for all j, $2 \le j \le N$, and all $\{0,1\}$-vectors $b \in \{0,1\}^{j-1}$, it holds

$$\Pr[B_j = 1 \mid B(b)] = \Pr[B_j = 1] = p.$$

Note that

$$\Pr[B_j = 1 \mid B(b)] = \frac{\Pr[B_j = 1 \wedge B(b)]}{\Pr[B(b)]}$$
$$= \sum_{b'\in\{0,1\}^{j-1}, b'\le b} \frac{\Pr[B_j = 1 \wedge B(b) \wedge C(b')]}{\Pr[B(b)]}$$
$$= \sum_{b'\in\{0,1\}^{j-1}, b'\le b} \frac{\Pr[B(b) \wedge C(b')]}{\Pr[B(b)]} \cdot \frac{\Pr[B_j = 1 \wedge B(b) \wedge C(b')]}{\Pr[B(b) \wedge C(b')]}$$
$$= \sum_{b'\in\{0,1\}^{j-1}, b'\le b} \frac{\Pr[B(b) \wedge C(b')]}{\Pr[B(b)]} \cdot \Pr[B_j = 1 \mid B(b) \wedge C(b')]$$
$$= p \cdot \frac{1}{\Pr[B(b)]} \cdot \sum_{b'\in\{0,1\}^{j-1}, b'\le b} \Pr[B(b) \wedge C(b')] = p.$$

□

Appendix B: The Proof of Corollary 2, Parts (b.4) and (c)

Let us fix an arbitrary number j, $1 \leq j \leq M$, and a green transcript $\tau \in \mathcal{T}^j_{\text{green}}$. We assume that part (a) of Corollary 2 holds, i.e., that for all $k \in K^{\text{green}}(\tau)$

$$\Pr_{\Omega^{\text{green}}(\tau)}[k] \leq 2^{-(n-1)}. \tag{34}$$

Let us first prove part (b.4) of Corollary 2. We fix some $x, \bar{y} \in \{0,1\}^n$, where $x \notin X^*(\tau)$ and $\bar{y} \notin \bar{Y}^*(\tau)$ and ($x \notin X(\tau)$ or $\bar{y} \notin \bar{Y}(\tau)$). We have to show

$$\Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y}] \leq 9 \cdot 2^{-(n-1)}. \tag{35}$$

Proof of Relation (35) We divide $\Omega^{\text{green}}(\tau)$ into two subsets Ω_1 and Ω_2, where

$$\begin{aligned} \Omega_1 &= \{\omega \in \Omega^{\text{green}}(\tau); x \oplus k_\omega \notin U(\tau)\}, \\ \Omega_2 &= \{\omega \in \Omega^{\text{green}}(\tau); x \oplus k_\omega \in U(\tau)\} \end{aligned} \tag{36}$$

and denote

$$\begin{aligned} K_1 &= \{k \in K^{\text{green}}(\tau); x \oplus k \notin U(\tau)\}, \\ K_2 &= \{k \in K^{\text{green}}(\tau); x \oplus k \in U(\tau)\}. \end{aligned} \tag{37}$$

The sets Ω_2 and K_2 define another set $W \subseteq \{0,1\}^n$ by

$$W = \{P_\omega(x \oplus k_\omega) \oplus k_\omega; \omega \in \Omega_2\} = \{P_\tau(x \oplus k) \oplus k; k \in K_2\}.$$

Here, P_τ denotes the restriction of P_ω to $U(\tau)$ which, by definition, is equal for all $\omega \in \Omega(\tau)$.

Note that $|W| \leq |K_2| \leq |U(\tau)| \leq j \leq M$.

Let us now define an equivalence relation on K_2. For keys $k, k' \in K_2$ we define that $k \equiv k'$ if $P_\tau(x \oplus k) \oplus k = P_\tau(x \oplus k') \oplus k'$.

Let $L_1, \cdots, L_s$ denote the equivalence classes corresponding to the equivalence relation $\equiv$ on K_2.

Clearly, $s = |W|$ and for each class L_l, $1 \leq l \leq s$, there is exactly one $w \in W$ such that $P_\tau(x \oplus k) \oplus k = w$ for all $k \in L_l$.

Note that $k \equiv k'$ implies that $x \oplus k \equiv_{P_\tau} x \oplus k'$ in the sense of Definition 12 and remember that, as τ is not black, $Max(P_\tau, U(\tau)) \leq 5$. This implies that for all $w \in W$

$$|\{k \in K_2; P_\tau(x \oplus k) \oplus k = w\}| \leq 5. \tag{38}$$

Note that

$$\begin{aligned} &\Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y}] \\ = &\Pr_{\Omega^{\text{green}}(\tau)}[\Omega_1] \cdot \Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y} \mid \Omega_1] \\ &+ \Pr_{\Omega^{\text{green}}(\tau)}[\Omega_2] \cdot \Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y} \mid \Omega_2], \end{aligned}$$

i.e.,

$$\begin{aligned} &\Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y}] \\ \leq &\Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y} \mid \Omega_1] \\ &+ \Pr_{\Omega^{\text{green}}(\tau)}[\Omega_2] \cdot \Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y} \mid \Omega_2]. \end{aligned} \tag{39}$$

For estimating $\Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y} \mid \Omega_1]$ note that if $x \oplus k_\omega \notin U(\tau)$ and $x \in X(\tau)$, then $P_\omega(x \oplus k_\omega)$ takes all values in $\{0,1\}^n$ which are outside of $V(\tau)$ and which are outside of $\bar{Y}(\tau) \oplus k_\omega$ with the same probability (see the proof of Lemma 3).[4] If $x \notin X(\tau)$ then $P_\omega(x \oplus k_\omega)$ takes all values in $\{0,1\}^n$, which are outside of $V(\tau)$ with the same probability (see the proof of Lemma 3).

This implies that

$$\Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y} \mid \Omega_1] \leq \frac{1}{2^n - (R+n)M} \leq 2^{-(n-1)} \tag{40}$$

if n is large enough.

Observe next that by Relation (34) it holds that

$$\Pr_{\Omega^{\text{green}}(\tau)}[\Omega_2] \leq 2^{-(n-1)} \cdot |K_2|. \tag{41}$$

For estimating $\Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y} \mid \Omega_2]$ we first consider the case that $\bar{y} \notin W$. Then, by the definition of W, it holds that

$$\Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y} \mid \Omega_2] = 0.$$

Assume now that $\bar{y} \in W$. From Relation 38 and the Smoothness Lemma (Lemma 5) it follows that

$$\Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y} \mid \Omega_2] \leq \sqrt{2}\frac{5}{|K_2|} \leq \frac{8}{|K_2|}. \tag{42}$$

Inserting relations (40), (41), and (42) into Relation (39) yields

$$\begin{aligned}\Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x \oplus k_\omega) \oplus k_\omega = \bar{y}] &\leq 2^{-(n-1)} + 2^{-(n-1)} \cdot |K_2| \cdot \frac{8}{|K_2|} \\ &= 9 \cdot 2^{-(n-1)}.\end{aligned}$$

□

Let us now prove part (c) of Corollary 2. We fix some $x \neq x' \in \{0,1\}^n$, where $x \notin X(\tau)$ and $x' \in X(\tau)$, and some number i, $-(R+n) \leq i \leq R+n$. We have to show

$$\Pr_{\Omega^{\text{green}}(\tau)}[Ev(x, x', i)] \leq 11 \cdot 2^{-(n-1)}, \tag{43}$$

where the event $Ev(x, x', i) \subseteq \Omega^{\text{green}}(\tau)$ is defined to be

$$Ev(x, x', i) = \left\{\omega \in \Omega^{\text{green}}(\tau); \pi^i\left(P_\omega(x \oplus k_\omega) \oplus k_\omega\right) = P_\omega(x' \oplus k_\omega) \oplus k_\omega\right\}.$$

Proof of Relation (43) Let us first handle the case that $x' \in X^*(\tau)$ and denote by y' the unique value for which $(x', y') \in Coll(\tau)$. Then, by the definition of structural *EF*-collisions, it holds that

$$P_\omega(x' \oplus k_\omega) \oplus k_\omega = y'$$

for all $\omega \in \Omega^{\text{green}}(\tau)$.

Consequently, for all $\omega \in \Omega^{\text{green}}(\tau)$ it holds that $\omega \in Ev(x, x', i)$ if and only if

$$P_\omega(x \oplus k_\omega) \oplus k_\omega = \pi^{-i}(y').$$

[4] Here, $P_\omega(x \oplus k_\omega) \in \bar{Y}(\tau) \oplus k_\omega$ would imply that $x \in X^*(\tau)$, which contradicts the assumption.

From part (b) of Corollary 2 it follows that if $x' \in X^*(\tau)$, then

$$\Pr_{\Omega^{\text{green}}(\tau)}[Ev(x, x', i)] \leq 9 \cdot 2^{-(n-1)} < 11 \cdot 2^{-(n-1)}.$$

Now let us consider the case that $x' \in X(\tau) \setminus X^*(\tau)$.

For arbitrary points $z \in \{0, 1\}^n$ we define

$$\Omega^{\text{green}}(\tau, z) = \{\omega \in \Omega^{\text{green}}(\tau);\, P_\omega(x \oplus k_\omega) \oplus k_\omega = z\}$$

and

$$K^{\text{green}}(\tau, z) = \{k \in \{0, 1\}^n ; \exists \omega \in \Omega^{\text{green}}(\tau, z) : k_\omega = k\}.$$

Moreover, for $b \in \{1, 2\}$ we define

$$\Omega_b(z) = \Omega^{\text{green}}(\tau, z) \cap \Omega_b$$

and

$$K_b(z) = K^{\text{green}}(\tau, z) \cap K_b,$$

where the sets Ω_1 and Ω_2 and the sets K_1 and K_2 are defined as in relations (36) and (37).

Let us clarify how the keys in elementary events in $\Omega_1(z)$ and $\Omega_2(z)$ and the keys in $K_1(z)$ and $K_2(z)$ look like.

It can be easily checked that for all $\omega = (k_\omega, P_\omega, F_\omega) \in \Omega_1$ it holds $\omega \in \Omega_1(z)$ if and only if $z \oplus k_\omega \notin V(\tau)$ and $P_\omega(x \oplus k_\omega) = z \oplus k_\omega$.

Moreover, for all $\omega = (k_\omega, P_\omega, F_\omega) \in \Omega_2$ it holds $\omega \in \Omega_2(z)$ if and only if $P_\tau(x \oplus k_\omega) \oplus k_\omega = z$, which implies by Relation 38 that

$$|K_2(z)| \leq 5. \tag{44}$$

We obtain that

$$\begin{aligned} |K^{\text{green}}(\tau, z)| \geq |K_1(z)| &\geq |K_1| - |V(\tau)| \\ &= |K^{\text{green}}(\tau)| - |K_2| - |V(\tau)| \\ &\geq |K^{\text{green}}(\tau)| - 2 \cdot |V(\tau)| \\ &\geq |K^{\text{green}}(\tau)| - 2M \geq \frac{1}{\sqrt{2}} \cdot 2^n \end{aligned} \tag{45}$$

if n is large enough. The last inequation follows from $M < 2^{(2/3)n}$ and the proof of the Smoothness Lemma (Lemma 5): It is a straightforward consequence of relations (52) and (53) in Appendix C that for all green transcripts τ and all constants $\delta < 1$ it holds that $|K^{\text{green}}(\tau)| \geq \delta \cdot 2^n$ if n is large enough. Lemma 5 states this only for $\delta = \frac{1}{\sqrt{2}}$.

By exactly the same arguments as in the Smoothness Lemma (Lemma 5) one can show that for all $k, k' \in K^{\text{green}}(\tau, z)$ it holds that

$$\Pr_{\Omega^{\text{green}}(\tau,z)}[k] \leq \sqrt{2} \cdot \Pr_{\Omega^{\text{green}}(\tau,z)}[k']. \tag{46}$$

if n is large enough.

Relations (45) and (46) imply directly that for all $k \in K^{\text{green}}(\tau, z)$ it holds

$$\Pr_{\Omega^{\text{green}}(\tau,z)}[k] \leq 2^{-(n-1)} \tag{47}$$

if n is large enough.

Note that

$$\begin{aligned}&\Pr_{\Omega^{\text{green}}(\tau)}[Ev(x,x',i)]\\ &= \sum_{z\in\{0,1\}^n} \Pr_{\Omega^{\text{green}}(\tau)}[P_\omega(x\oplus k_\omega)\oplus k_\omega = z]\cdot \Pr_{\Omega^{\text{green}}(\tau,z)}[Ev(x,x',i)].\end{aligned} \tag{48}$$

For deriving an upper bound for $\Pr_{\Omega^{\text{green}}(\tau,z)}[Ev(x,x',i)]$ we write

$$\begin{aligned}\Pr_{\Omega^{\text{green}}(\tau,z)}[Ev(x,x',i)] &= \Pr_{\Omega^{\text{green}}(\tau,z)}[\Omega_1(z)]\cdot \Pr_{\Omega_1(z)}[Ev(x,x',i)]\\ &\quad + \Pr_{\Omega^{\text{green}}(\tau,z)}[\Omega_2(z)]\cdot \Pr_{\Omega_2(z)}[Ev(x,x',i)]\\ &\le \Pr_{\Omega_1(z)}[Ev(x,x',i)] + \Pr_{\Omega^{\text{green}}(\tau,z)}[\Omega_2(z)]\\ &\le \Pr_{\Omega_1(z)}[Ev(x,x',i)] + 5\cdot 2^{-(n-1)},\end{aligned} \tag{49}$$

where the last inequality follows from relations (44) and (47).

We write $K_1(z)$ as

$$K_1(z) = K_3(z)\cup K_4(z)\cup K_5(z),$$

where

- $K_3(z) = \{k\in K_1(z); x'\oplus k\in U(\tau),\ P_\tau(x'\oplus k)\oplus k = \pi^i(z)\}$,
- $K_4(z) = \{k\in K_1(z); x'\oplus k\in U(\tau),\ P_\tau(x'\oplus k)\oplus k \neq \pi^i(z)\}$,
- $K_5(z) = \{k\in K_1(z); x'\oplus k\notin U(\tau)\}$.

From Relation 38 we know that $|K_3(z)|\le 5$. Moreover, for all $k\in K_4(z)$ it holds

$$\Pr_{\Omega_1(z)}[Ev(x,x',i)\cap(k_\omega\in K_4(z))] = 0.$$

Consequently,

$$\Pr_{\Omega_1(z)}[Ev(x,x',i)] \le \Pr_{\Omega_5(z)}[Ev(x,x',i)] + 5\cdot 2^{-(n-1)}, \tag{50}$$

where $\Omega_5(z) = \{\omega\in\Omega_1(z); k_\omega\in K_5(z)\}$.

Note that for all $\omega\in\Omega_5(z)$, the condition that $\omega\in Ev(x,x',i)$ is equivalent to

$$P_\omega(x'\oplus k_\omega) = \pi^i(z)\oplus k_\omega,$$

which has probability 0 if $\pi^i(z)\oplus k_\omega\in V(\tau)$ and probability at most

$$\frac{1}{2^n - (|V(\tau)|+1) - |\bar{Y}(\tau)|} \le 2^{-(n-1)}$$

if $\pi^i(z)\oplus k_\omega\notin V(\tau)$ and n is large enough, see Relation (40) and the comment before Relation (40).

We obtain that

$$\Pr_{\Omega_5(z)}[Ev(x,x',i)] \le 2^{-(n-1)} \tag{51}$$

if n is large enough.

Putting relations (51), (50), and (49) together, we obtain that for all $z\in\{0,1\}^n$

$$\Pr_{\Omega^{\text{green}}(\tau,z)}[Ev(x,x',i)] \le 11\cdot 2^{-(n-1)},$$

which implies by Relation (48) that

$$\Pr_{\Omega^{\text{green}}(\tau)}[Ev(x,x',i)] \le 11\cdot 2^{-(n-1)}.$$

□

Appendix C: The Proof of the Smoothness Lemma (Lemma 5), Part (II)

We fix an arbitrary number j, $1 \leq j \leq M$, and a green transcript $\tau \in \mathcal{T}^j_{\text{green}}$. We analyze the probability distribution $\Pr_{\Omega(\tau)}$ on $K^{\text{green}}(\tau)$ by showing that for all $k \in K^{\text{green}}(\tau)$ it holds that this distribution is close to the uniform distribution on $K^{\text{green}}(\tau)$. More precisely,

$$\Pr_{\Omega(\tau)}[k] \leq \delta \cdot |K^{\text{green}}(\tau)|^{-1}$$
$$\text{for } \delta = \left(\frac{2^{n-1}}{2^{n-1} - (R+n)j}\right)^{2\Delta}. \tag{52}$$

Note that part (3) of Theorem 6 implies $\delta \leq \sqrt{2}$. This is because for $\theta = (R+n)j$ and $N = 2^{n-1}$ we can write δ as

$$\delta = \left(\frac{N}{N-\theta}\right)^{2\Delta} = \left(\frac{1}{1-\theta/N}\right)^{2\Delta} = \left(\left(\frac{1}{1-\theta/N}\right)^{N/\theta}\right)^{\frac{2\Delta\theta}{N}} \approx e^{\frac{2\Delta\theta}{N}}. \tag{53}$$

Part (3) of Theorem 6 implies that $\Delta \cdot \theta \leq \frac{\ln(2)\cdot N}{4}$, which is equivalent to

$$\frac{2\Delta\theta}{N} \leq \frac{\ln 2}{2}.$$

The Proof of Relation (52) The proof of Lemma 3 shows how, for keys $k \in K(\tau)$, completions P' of P_τ on $\{0,1\}^n \setminus U(\tau)$ and F' of F_τ on $\{0,1\}^n \setminus Y(\tau)$ have to be constructed such that (k, P', F') belongs to $\Omega(\tau)$. In particular:

(1) The function values of P' on $X^*(\tau) \oplus k$, a set of size $|X^*(\tau)|$, are determined.
(2) The function values of P' on the set $((X(\tau) \setminus X^*(\tau)) \oplus k) \setminus U(\tau)$ are forbidden to fall into the set $\bar{Y}(\tau) \oplus k$.
(3) We assume that P' is an arbitrarily fixed completion of P_τ satisfying (1) and (2) and describe how a completion F' of F_τ has to be constructed in such a way that (k, P', F') belongs to $\Omega(\tau)$. The function values $F'(y)$ are determined or partly determined on a set $Z(k) = Z_1(k) \cup Z_2(k)$, where $Z_1(k)$ contains all those $y \in \{0,1\}^n$ for which there is some r, $-(n-1) \leq r \leq R-1$, such that $\pi^{-r}(y) = P'(x \oplus k) \oplus k$ for some $x \in X(\tau) \setminus X^*(\tau)$, and $Z_2(k)$ contains all those $y \in \{0,1\}^n$ for which there is some i, $-(n-1) \leq i \leq n-1$, such that $\pi^i(y) \in Y(\tau)$. Note that $Z_1(k) \cap Z_2(k) = \emptyset$ and $|Z_1(k)| = |Z_1(k')|$ and $|Z_2(k)| = |Z_2(k')|$ for all $k, k' \in K^{\text{green}}(\tau)$. This implies that $Z(k)$ has the same size for all $k \in K^{\text{green}}(\tau)$.

We use the following fact. Let A_1, B_1, A_2, B_2 be finite sets fulfilling $A_1 \cap A_2 = B_1 \cap B_2 = \emptyset$, $|A_1 \cup A_2| = |B_1 \cup B_2|$, $|A_2| < |B_1|$ and $|B_2| < |A_1|$. Then the number of bijective mappings $f : A_1 \cup A_2 \longrightarrow B_1 \cup B_2$ for which $f(A_2) \subseteq B_1$ is

$$\frac{|A_1|! \cdot |B_1|!}{(|A_1| - |B_2|)!} = \frac{|A_1|! \cdot |B_1|!}{(|B_1| - |A_2|)!}. \tag{54}$$

In the following, A_1 corresponds to the set $\{0,1\}^n \setminus (U(\tau) \cup (X(\tau) \oplus k))$, A_2 to the set $((X(\tau) \setminus X^*(\tau)) \oplus k) \setminus U(\tau)$, B_1 to the set $\{0,1\}^n \setminus (V(\tau) \cup (\bar{Y}(\tau) \oplus k))$ and B_2 to the set $((\bar{Y}(\tau) \setminus \bar{Y}^*(\tau)) \oplus k) \setminus V(\tau)$. We denote

- $T = 2^n - |X(\tau)| - |U(\tau)|$,
- $t = |X(\tau) \setminus X^*(\tau)|$,
- $S = 2^n - |\bar{Y}(\tau)| - |V(\tau)|$,

– $s = |\bar{Y}(\tau) \setminus \bar{Y}^*(\tau)|$.

It holds that $S > 2^{n-1}$ and $T > 2^{n-1}$ if n is large enough.

Note that for proving Lemma 5 it is sufficient to show that for all pairs $k, k' \in K^{\text{green}}(\tau)$

$$\frac{\Pr_{\Omega(\tau)}[k]}{\Pr_{\Omega(\tau)}[k']} \leq \delta.$$

This is because τ is green, which, together with $\Pr_{\Omega^{\text{blue}}(\tau)}[k] = \Pr_{\Omega^{\text{blue}}(\tau)}[k'] = 0$ for $k, k' \in K^{\text{green}}(\tau)$, implies

$$\begin{aligned}\frac{\Pr_{\Omega(\tau)}[k]}{\Pr_{\Omega(\tau)}[k']} &= \frac{\Pr_{\Omega(\tau)}[\Omega^{\text{green}}(\tau)] \cdot \Pr_{\Omega^{\text{green}}(\tau)}[k] + \Pr_{\Omega(\tau)}[\Omega^{\text{blue}}(\tau)] \cdot \Pr_{\Omega^{\text{blue}}(\tau)}[k]}{\Pr_{\Omega(\tau)}[\Omega^{\text{green}}(\tau)] \cdot \Pr_{\Omega^{\text{green}}(\tau)}[k'] + \Pr_{\Omega(\tau)}[\Omega^{\text{blue}}(\tau)] \cdot \Pr_{\Omega^{\text{blue}}(\tau)}[k']} \\ &= \frac{\Pr_{\Omega^{\text{green}}(\tau)}[k]}{\Pr_{\Omega^{\text{green}}(\tau)}[k']}.\end{aligned}$$

For this purpose, let us denote by $ConsP_\tau(k)$ the set of all τ-consistent completions P' of P_τ on $\{0,1\}^n \setminus U(\tau)$, i.e., of all completions P' of P_τ for which there is some completion F' of F_τ on $\{0,1\}^n \setminus Y(\tau)$ such that $(k, P', F') \in \Omega(\tau)$.

The above statement (3) implies that for all $k \in K(\tau)$ and completions $P' \in ConsP_\tau(k)$, the number of such completions F' is the same, i.e., does not depend on k.

This implies that

$$\frac{\Pr_{\Omega(\tau)}[k]}{\Pr_{\Omega(\tau)}[k']} = \frac{|ConsP_\tau(k)|}{|ConsP_\tau(k')|}.$$

Note that due to requirement (2) (see above), the size of $ConsP_\tau(k)$ depends on the sizes of the sets $((X(\tau) \setminus X^*(\tau)) \oplus k) \setminus U(\tau)$ and $((\bar{Y}(\tau) \setminus \bar{Y}^*(\tau)) \oplus k) \setminus V(\tau)$. As k is not blue, the sizes of the corresponding intersections $((X(\tau) \setminus X^*(\tau)) \oplus k) \cap U(\tau)$ and $((\bar{Y}(\tau) \setminus \bar{Y}^*(\tau)) \oplus k) \cap V(\tau)$ can vary only between 0 and Δ (see Definition 19).

Thus, for $k \in K^{\text{green}}(\tau)$, the value $|ConsP_\tau(k)|$ is minimal if

$$|((X(\tau) \setminus X^*(\tau)) \oplus k) \cap U(\tau)| = |((\bar{Y}(\tau) \setminus \bar{Y}^*(\tau)) \oplus k) \cap V(\tau)| = 0.$$

By Relation (54) this implies

$$|ConsP_\tau(k)| = \frac{S! \cdot T!}{(T-s)!} = \frac{S! \cdot T!}{(S-t)!}. \tag{55}$$

The value $|ConsP_\tau(k)|$ is maximal if

$$\begin{aligned} &|((X(\tau) \setminus X^*(\tau)) \oplus k) \cap U(\tau)| = \min\{\Delta, t\} \\ \text{and} \quad &|((\bar{Y}(\tau) \setminus \bar{Y}^*(\tau)) \oplus k) \cap V(\tau)| = \min\{\Delta, s\}. \end{aligned} \tag{56}$$

We now have to distinguish three cases corresponding to whether $|X(\tau) \setminus X^*(\tau)| > \Delta$ or not and whether $|\bar{Y}(\tau) \setminus \bar{Y}^*(\tau)| > \Delta$ or not.

Case 1 $|X(\tau) \setminus X^*(\tau)| > \Delta$ and $|\bar{Y}(\tau) \setminus \bar{Y}^*(\tau)| > \Delta$.

In this case, it follows from relations (54) and (56) that

$$|ConsP_\tau(k)| \leq \frac{(S+\Delta)! \cdot (T+\Delta)!}{(S+\Delta-(t-\Delta))!} = \frac{(S+\Delta)! \cdot (T+\Delta)!}{(S+2\Delta-t)!}. \tag{57}$$

Relations (55) and (57) imply that the $\Pr_{\Omega(\tau)}$-values of elements from $K^{\text{green}}(\tau)$ can differ by a factor δ which is at most $\delta_1 \cdot \delta_2$, where

$$\begin{aligned}\delta_1 &= \frac{(T+1)(T+2)\cdots(T+\Delta)}{(S-(t-1))(S-(t-1)+1)\cdots(S-(t-1)+\Delta-1)} \\ &\leq \left(\frac{T}{S-t}\right)^{\Delta} \end{aligned} \tag{58}$$

and

$$\begin{aligned}\delta_2 &= \frac{(S+1)(S+2)\cdots(S+\Delta)}{(S-(t-1)+\Delta)(S-(t-1)+\Delta+1)\cdots(S-(t-1)+2\Delta-1)} \\ &\leq \left(\frac{S}{S-(t-\Delta)}\right)^{\Delta} \leq \left(\frac{S}{S-j}\right)^{\Delta} \leq \left(\frac{2^{n-1}}{2^{n-1}-j}\right)^{\Delta}.\end{aligned}$$

Here we used the fact that from $a \geq b$ it follows that $\frac{a}{b} \geq \frac{a+1}{b+1}$.

For upper bounding δ_1 we have to distinguish the two cases $S \geq T$ and $S < T$.

If $S \geq T$, then, by Relation (58),

$$\delta_1 \leq \left(\frac{S}{S-t}\right)^{\Delta} \leq \left(\frac{2^{n-1}}{2^{n-1}-j}\right)^{\Delta}. \tag{59}$$

If $S < T$, then observe that $S \geq T - (R+n-1)j$. This holds because $|\bar{Y}(\tau)| \leq (R+n-1)j$. Consequently,

$$\begin{aligned}\delta_1 &\leq \left(\frac{T}{T-t-(R+n-1)j}\right)^{\Delta} \\ &\leq \left(\frac{T}{T-j-(R+n-1)j}\right)^{\Delta} \\ &= \left(\frac{T}{T-(R+n)j}\right)^{\Delta} \leq \left(\frac{2^{n-1}}{2^{n-1}-(R+n)j}\right)^{\Delta}.\end{aligned} \tag{60}$$

Case 2 $|X(\tau) \setminus X^*(\tau)| \leq \Delta$, i.e., $t \leq \Delta$.

Here, $|ConsP_\tau(k)|$ is maximal if $(X(\tau) \setminus X^*(\tau)) \oplus k$ is a subset of $U(\tau)$, which implies that $|ConsP_\tau(k)|$ is $(T+t)!$ and that, by Relation (55), the $|ConsP_\tau(k)|$-values for $k \in K^{\text{green}}(\tau)$ cannot differ by a factor larger than

$$\begin{aligned}\frac{(T+t)! \cdot (S-t)!}{T! \cdot S!} &= \frac{(T+1)\cdot\ldots\cdot(T+t)}{(S-t+1)\cdot\ldots\cdot S} \\ &\leq \left(\frac{T}{S-t}\right)^{t} \leq \left(\frac{T}{S-t}\right)^{\Delta} \leq \left(\frac{2^{n-1}}{2^{n-1}-(R+n)j}\right)^{\Delta}.\end{aligned}$$

Note that the last inequation follows from the same case distinction (i.e., $S \geq T$ and $S < T$) that was already performed as part of *Case 1* above (see relations (59) and (60)).

Case 3 $|\bar{Y}(\tau) \setminus \bar{Y}^*(\tau)| \leq \Delta$, i.e., $s \leq \Delta$.

Here, $|ConsP_\tau(k)|$ is maximal if $(\bar{Y}(\tau) \setminus \bar{Y}^*(\tau)) \oplus k$ is a subset of $V(\tau)$, which implies that $|ConsP_\tau(k)|$ is $(S+s)!$ and that, by Relation (55), the $|ConsP_\tau(k)|$-values for $k \in$

$K^{\text{green}}(\tau)$ cannot differ by a factor larger than

$$\begin{aligned}
&\frac{(S+s)!\cdot(T-s)!}{T!\cdot S!} = \frac{(S+1)\cdot\ldots\cdot(S+s)}{(T-s+1)\cdot\ldots\cdot T} \\
&\leq \left(\frac{S}{T-s}\right)^{s} \leq \left(\frac{S}{T-s}\right)^{\Delta} \leq \left(\frac{S}{S-j-s}\right)^{\Delta} \\
&\leq \left(\frac{S}{S-j-(R+n-1)j}\right)^{\Delta} = \left(\frac{S}{S-(R+n)j}\right)^{\Delta} \leq \left(\frac{2^{n-1}}{2^{n-1}-(R+n)j}\right)^{\Delta}.
\end{aligned}$$

Summarizing all three cases we obtain that

$$\delta \leq \left(\frac{2^{n-1}}{2^{n-1}-(R+n)j}\right)^{2\Delta}.$$

□

Appendix D: A List of Denotations used in the Proof of Theorem 6

- Elementary Events $\omega = (k_\omega, P_\omega, F_\omega)$ and the Probability Space Ω: Definition 4 in Section 4.
- ω-critical Triples: Definition 5 in Section 5.1.
- ω-dangerous Triples: Definition 5 in Section 5.1.
- ω-sudden death Triples: Definition 5 in Section 5.1.
- The number $B(M, R, n)$: Definition 6 in Section 5.2.
- Chernoff Bounds: Appendix A.
- Structural EF- and EE-Collisions: Definition 7 in Section 5.3.
- The Friendly Alice: Definition 8 in Section 5.3.
- Sudden Death: Definition 9 in Section 5.3.
- Transcripts τ: Section 5.4.
- Ω^{succ}, $\Omega^{\text{s.death}}$: Section 5.4.
- Set of transcripts $\mathcal{T}^j$: beginning of Section 5.5.
- The sets $X(\tau)$, $Y(\tau)$, $U(\tau)$, $V(\tau)$, $X^*(\tau)$, $\bar{Y}^*(\tau)$, $\bar{Y}^{(r)}(\tau)$, $\bar{Y}(\tau)$, $Coll(\tau)$, $\Omega(\tau)$, $K(\tau)$ for transcripts τ: Definition 10 in Section 5.5.
- τ-consistency: Definition 10 in Section 5.5.
- (τ, k)-critical Points: Definition 11 in Section 5.6.
- The relation $\equiv_P$: Definition 12 in Section 5.7.
- τ-critical Keys: Definition 13 in Section 5.7.
- Black Transcripts and Events, Ω^j_{black}, Ω^{black}, $\mathcal{T}^j_{\text{black}}$: Definition 15 in Section 5.7.
- Restrictions $\tau_\omega^{\leq j}$ of transcripts: Definition 15 in Section 5.7.
- Red Transcripts and Events: Definition 16 in Section 5.7.
- Blue Events: Definition 17 in Section 5.7.
- j-alive Elementary Events: Definition 14 in Section 5.7.
- Green and j-green Elementary Events and Green Transcripts: Definition 18 in Section 5.7.
- τ-green resp. τ-blue Keys, $K^{\text{green}}(\tau)$ resp. $K^{\text{blue}}(\tau)$: Definition 19 in Section 5.7.

References

1. Alon, N., Spencer, J., Erdös, P.: The Probabilistic method. Wiley Interscience, New York (1992)
2. Andreeva, E., Bogdanov, A., Dodis, Y., Mennink, B., Steinberger, J.P.: On the indifferentiability of key-alternating ciphers. In: Canetti, R., Garay, J.A. (eds.) Advances in cryptology - CRYPTO 2013 - 33rd annual cryptology conference, Santa Barbara, August 18-22, 2013. Proceedings, Part I, Lecture Notes in Computer Science, vol. 8042, pp. 531–550. Springer (2013). https://doi.org/10.1007/978-3-642-40041-4_29
3. Armknecht, F., Hamann, M., Mikhalev, V.: Lightweight authentication protocols on Ultra-Constrained RFIDs - Myths and facts. In: Saxena, N., Sadeghi, A.R. (eds.) Radio frequency identification: security and privacy issues: 10th international workshop, RFIDSec 2014, Oxford, July 21-23, 2014, Revised Selected Papers, pp. 1–18. Springer International Publishing, Cham (2014). https://doi.org/10.1007/978-3-319-13066-8_1
4. Armknecht, F., Mikhalev, V.: On lightweight stream ciphers with shorter internal states. In: Leander, G. (ed.) Fast software encryption: 22nd international workshop, FSE 2015, Istanbul, March 8-11, 2015, Revised Selected Papers, pp. 451–470. Springer, Berlin (2015). https://doi.org/10.1007/978-3-662-48116-5_22
5. Babbage, S.: Improved exhaustive search attacks on stream ciphers. In: European Convention on Security and Detection, 1995, pp. 161–166 (1995). https://doi.org/10.1049/cp:19950490
6. Banik, S.: Some results on sprout. In: Biryukov, A., Goyal, V. (eds.) Progress in cryptology – INDOCRYPT 2015: 16th international conference on cryptology in India, Bangalore, December 6-9, 2015, Proceedings, pp. 124–139. Springer International Publishing, Cham (2015). https://doi.org/10.1007/978-3-319-26617-6_7
7. Banik, S., Isobe, T.: Some cryptanalytic results on Lizard. Cryptology ePrint Archive Report 2017/346 (2017). http://eprint.iacr.org/2017/346
8. Berbain, C., Gilbert, H.: On the security of IV dependent stream ciphers. In: Biryukov, A. (ed.) Fast Software Encryption: 14th International Workshop, FSE 2007, Luxembourg, March 26-28, 2007, Revised Selected Papers, pp. 254–273. Springer, Berlin (2007). https://doi.org/10.1007/978-3-540-74619-5_17
9. Biryukov, A., Shamir, A.: Cryptanalytic Time/Memory/Data tradeoffs for stream ciphers. In: Okamoto, T. (ed.) Advances in cryptology — ASIACRYPT 2000: 6th international conference on the theory and application of cryptology and information security Kyoto, Japan, December 3–7, 2000 Proceedings, pp. 1–13. Springer, Berlin (2000). https://doi.org/10.1007/3-540-44448-3_1
10. Biryukov, A., Shamir, A., Wagner, D.: Real time cryptanalysis of A5/1 on a PC. In: Goos, G., Hartmanis, J., van Leeuwen, J., Schneier, B. (eds.) Fast software encryption: 7th international workshop, FSE 2000 New York, April 10–12, 2000 Proceedings, pp. 1–18. Springer, Berlin (2001). https://doi.org/10.1007/3-540-44706-7_1
11. Bogdanov, A., Knudsen, L.R., Leander, G., Standaert, F.X., Steinberger, J., Tischhauser, E.: Key-Alternating ciphers in a provable setting: encryption using a small number of public permutations. In: Pointcheval, D., Johansson, T. (eds.) Advances in cryptology – EUROCRYPT 2012, lecture notes in computer science, vol. 7237, pp. 45–62. Springer, Berlin (2012). https://doi.org/10.1007/978-3-642-29011-4_5
12. Cannière, C.D., Preneel, B.: Trivium – specifications eSTREAM: the ECRYPT stream cipher project (2005). http://www.ecrypt.eu.org/stream/p3ciphers/trivium/trivium_p3.pdf
13. Chen, S., Lampe, R., Lee, J., Seurin, Y., Steinberger, J.: Minimizing the Two-Round Even-Mansour cipher. In: Garay, J.A., Gennaro, R. (eds.) Advances in cryptology – CRYPTO 2014, lecture notes in computer science, vol. 8616, pp. 39–56. Springer, Berlin (2014). https://doi.org/10.1007/978-3-662-44371-2_3
14. Chen, S., Steinberger, J.: Tight security bounds for Key-Alternating ciphers. In: Nguyen, P.Q., Oswald, E. (eds.) Advances in cryptology – EUROCRYPT 2014, lecture notes in computer science, vol. 8441, pp. 327–350. Springer, Berlin (2014). https://doi.org/10.1007/978-3-642-55220-5_19
15. Dierks, T., Rescorla, E.: The transport layer security (TLS) protocol version 1.2. RFC 5246 (Proposed Standard). http://www.ietf.org/rfc/rfc5246.txt. Updated by RFCs 5746, 5878, 6176, 7465, 7507, 7568, 7627, 7685, 7905, 7919 (2008)
16. Dunkelman, O., Keller, N., Shamir, A.: Minimalism in cryptography: The Even-mansour scheme revisited. In: Proceedings of the 31st annual international conference on theory and applications of cryptographic techniques, EUROCRYPT'12, pp. 336–354. Springer, Berlin, (2012). https://doi.org/10.1007/978-3-642-29011-4_21
17. Esgin, M.F., Kara, O.: Practical cryptanalysis of full sprout with TMD tradeoff attacks. In: Dunkelman, O., Keliher, L. (eds.) Selected areas in cryptography - SAC 2015: 22nd international conference,

Sackville, August 12-14, 2015, Revised Selected Papers, pp. 67–85. Springer International Publishing, Cham (2016). https://doi.org/10.1007/978-3-319-31301-6_4
18. Even, S., Mansour, Y.: A construction of a cipher from a single pseudorandom permutation. In: Imai, H., Rivest, R., Matsumoto, T. (eds.) Advances in cryptology — ASIACRYPT '91, lecture notes in computer science, vol. 739, pp. 210–224. Springer, Berlin (1993). https://doi.org/10.1007/3-540-57332-1_17
19. Gazi, P., Tessaro, S.: Secret-key cryptography from ideal primitives: A systematic overview. Information theory workshop (ITW), 2015 IEEE, pp. 1–5 (2015). https://doi.org/10.1109/ITW.2015.7133163
20. Ghafari, V.A., Hu, H., Xie, C.: Fruit: Ultra-lightweight stream cipher with shorter internal state. Cryptology ePrint Archive Report 2016/355 (2016). http://eprint.iacr.org/2016/355
21. Hamann, M., Krause, M.: Stream cipher operation modes with improved security against generic collision attacks. Cryptology ePrint Archive Report 2015/757 (2015). https://eprint.iacr.org/2015/757
22. Hamann, M., Krause, M., Meier, W.: LIZARD – A lightweight stream cipher for power-constrained devices. IACR Transactions on Symmetric Cryptology **2017**(1), 45–79 (2017). https://doi.org/10.13154/tosc.v2017.i1.45-79. http://tosc.iacr.org/index.php/ToSC/article/view/584
23. Hamann, M., Krause, M., Meier, W., Zhang, B.: Design and analysis of small-state Grain-like stream ciphers Cryptography and Communications. https://doi.org/10.1007/s12095-017-0261-6 (2017)
24. Hell, M., Johansson, T., Maximov, A., Meier, W.: The grain family of stream ciphers. In: Robshaw, M., Billet, O. (eds.) New stream cipher designs: The eSTREAM finalists, pp. 179–190. Springer, Berlin (2008). https://doi.org/10.1007/978-3-540-68351-3_14
25. Hell, M., Johansson, T., Meier, W.: Grain - A stream cipher for constrained environments eSTREAM: The ECRYPT Stream Cipher Project (2006). http://www.ecrypt.eu.org/stream/p3ciphers/grain/Grain_p3.pdf
26. Hoang, V.T., Tessaro, S.: Key-alternating ciphers and key-length extension: Exact bounds and multi-user security. Cryptology ePrint Archive Report 2016/578 (2016). http://eprint.iacr.org/2016/578
27. Lallemand, V., Naya-Plasencia, M.: Cryptanalysis of full sprout. In: Gennaro, R., Robshaw, M. (eds.) Advances in cryptology – CRYPTO 2015: 35th annual cryptology conference, Santa Barbara, August 16-20, 2015, Proceedings, Part I, pp. 663–682. Springer, Berlin (2015). https://doi.org/10.1007/978-3-662-47989-6_32
28. Lu, Y., Meier, W., Vaudenay, S.: The conditional correlation attack: A practical attack on bluetooth encryption. In: Shoup, V. (ed.) Advances in cryptology – CRYPTO 2005: 25th annual international cryptology conference, Santa Barbara, August 14-18, 2005. Proceedings, pp. 97–117. Springer, Berlin (2005). https://doi.org/10.1007/11535218_7
29. Maitra, S., Sinha, N., Siddhanti, A., Anand, R., Gangopadhyay, S.: A TMDTO attack against lizard. Cryptology ePrint Archive Report 2017/647 (2017). http://eprint.iacr.org/2017/647
30. Mikhalev, V., Armknecht, F., Müller, C.: On ciphers that continuously access the non-volatile key. IACR Transactions on Symmetric Cryptology **2016**(2), 52–79 (2017). https://doi.org/10.13154/tosc.v2016.i2.52-79, http://tosc.iacr.org/index.php/ToSC/article/view/565
31. Siddhanti, A.A., Sarkar, S., Maitra, S., Chattopadhyay, A.: Differential fault attack on grain v1, ACORN v3 and lizard. Cryptology ePrint Archive Report 2017/678 (2017). http://eprint.iacr.org/2017/678
32. SIG, B.: Bluetooth core specification 4.2. https://www.bluetooth.org/DocMan/handlers/DownloadDoc.ashx?doc_id=286439 (2014)
33. Zhang, B., Gong, X.: Another tradeoff attack on sprout-like stream ciphers. In: Iwata, T., Cheon, H.J. (eds.) Advances in cryptology – ASIACRYPT 2015: 21st international conference on the theory and application of cryptology and information security, Auckland, November 29 – December 3, 2015, Proceedings, Part II, pp. 561–585. Springer, Berlin (2015). https://doi.org/10.1007/978-3-662-48800-3_23