The Conditional Correlation Attack: A Practical Attack on Bluetooth Encryption

Yi Lu[1], Willi Meier[2] and Serge Vaudenay[1]

[1] EPFL, CH-1015 Lausanne, Switzerland
http://lasecwww.epfl.ch
[2] FH Aargau, CH-5210 Windisch, Switzerland
meierw@fh-aargau.ch

Abstract. Motivated by the security of the nonlinear filter generator, the concept of correlation was previously extended to the conditional correlation, that studied the linear correlation of the inputs conditioned on a given (short) output pattern of some specific nonlinear function. Based on the conditional correlations, conditional correlation attacks were shown to be successful and efficient against the nonlinear filter generator. In this paper, we further generalize the concept of conditional correlations by assigning it with a different meaning, i.e. the correlation of the output of an arbitrary function conditioned on the unknown (partial) input which is uniformly distributed. Based on this generalized conditional correlation, a general statistical model is studied for dedicated key-recovery distinguishers. It is shown that the generalized conditional correlation is no smaller than the unconditional correlation. Consequently, our distinguisher improves on the traditional one (in the worst case it degrades into the traditional one). In particular, the distinguisher may be successful even if no ordinary correlation exists. As an application, a conditional correlation attack is developed and optimized against Bluetooth two-level E0. The attack is based on a recently detected flaw in the resynchronization of E0, as well as the investigation of conditional correlations in the Finite State Machine (FSM) governing the keystream output of E0. Our best attack finds the original encryption key for two-level E0 using the first 24 bits of $2^{23.8}$ frames and with 2^{38} computations. This is clearly the fastest and only practical known-plaintext attack on Bluetooth encryption compared with all existing attacks. Current experiments confirm our analysis.

Keywords. Stream Ciphers, Correlation, Bluetooth, E0

1 Introduction

In stream ciphers, correlation properties play a vital role in correlation attacks (to name a few, see [7–9, 15, 18, 19, 26, 27, 30]). For LFSR-based[3] keystream generators, such as the nonlinear filter generator or the combiner, *correlation* commonly means a statistically biased relation between the produced keystream and the output of certain LFSR sequences. In [1, 21, 22], the concept of (ordinary) correlations was further extended to the *conditional correlation* to describe the *linear* correlation of the inputs conditioned on a *given* (short) output pattern of a nonlinear function (with small input size). Based on conditional correlations, the conditional correlation attack received successful studies towards the nonlinear filter generator in [1, 21, 22]. In this paper, we assign a different meaning to conditional correlations, i.e. the correlation of the output of an arbitrary function (with favorable small input size) conditioned on the *unknown* (partial) input which is uniformly distributed. This might be viewed as the generalized opposite of [1, 21, 22]. As a useful application of

[3] LFSR refers to Linear Feedback Shift Register, see [28] for more.

our conditional correlations, imagine the attacker not only observes the keystream, but also has access to an intermediate computation process controlled partly by the key, which outputs a hopefully biased sequence for the right key and (presumably) unbiased sequences for wrong keys. If such side information is available, the *conditional correlation attack* may become feasible, which exploits correlations of the intermediate computation output conditioned on (part of) the inputs. In general, as informally conjectured in [22], conditional correlations are different and often larger than ordinary (unconditional) correlations, which effects reduced data complexity of conditional correlation attacks over ordinary correlation attacks.

Our first contribution consists of extracting a precise and general statistical model for dedicated key-recovery distinguishers based on the generalized conditional correlations. This framework deals with a specific kind of smart distinguishers that exploit correlations conditioned on the (partial) input, which is not restricted to keystream generators and is also applicable to other scenarios (e.g. side channel attacks like fault attacks in [4]). As the ordinary correlation serves as the criterion for the data complexity of the traditional distinguisher (that only exploits ordinary correlations), our result based on the sound theory of traditional distinguisher [5] tells that the conditional correlation serves similarly as the criterion for the data complexity of the smart distinguisher. The construction of the smart distinguisher also solves the unaddressed problem in [1, 21, 22] on how to make the best use of all the collected data, which can be transformed in the context of [1, 21, 22]. We prove that the smart distinguisher improves on the traditional one (in the worst case the smart distinguisher degrades into the traditional one), because our generalized conditional correlation is no smaller than the unconditional correlation. In particular, the smart distinguisher can still work efficiently even though the traditional one fails thoroughly. Meanwhile, we also study the computational complexity of the deterministic smart distinguisher for a special case, in which the essence of the major operation done by the distinguisher is identified to be *nothing but the regular convolution*. Thanks to Fast Walsh Transform[4] (FWT), when the key size is not too large, the smart distinguisher is able to achieve the optimal complete information set decoding and becomes a very powerful computing machine. Nonetheless, in general, with a very large key size, it is unrealistic to use the deterministic distinguisher as complete information set decoding is impractical; many other efficient decoding techniques (e.g. the probabilistic iterative decoding) such as introduced in the previous conditional correlation attacks [22] or the correlation attacks will also apply to our smart distinguisher.

As a second contribution, we apply our smart distinguisher to a conditional correlation attack[5] on two-level E0, the keystream generator that is used in the short-range wireless technology Bluetooth [6]. The attack exploits the resynchronization flaw recently detected in [24]. Whereas in [24], this flaw is used for a traditional distinguisher based on results [12, 16, 17, 23] of ordinary correlations, our conditional correlation attack relies on the systematic investigation of correlations conditioned on the inputs to the FSM in E0. These correlations extend a specific conditional correlation found in [23], which relates to one of the largest known biases in E0 as proved in [23]. The time complexity of our attack is optimized as the smart distinguisher works particularly well in this favorable case. Our best attack recovers the original encryption key for two-level E0 using the first 24 bits of $2^{23.8}$ frames after 2^{38} computations. Note that the number of necessary frames is below the maximum number 2^{26} of resynchronizations with the same user key as specified by

[4] Note that most recently FWT was successfully applied in [9, 23] to optimize different problems in correlation attacks.

[5] For the conditional correlation attack related to the previous work [1, 21, 22] on Bluetooth E0, see [16].

Bluetooth [6]. Compared with all existing attacks [13,14,16,20,24,29] on two-level E0, our attack is clearly the fastest and only practical resynchronization attack[6] so far. Note that the resynchronization attacks on one-level E0 were well studied in [3, 14,24] to be much more efficient.

The rest of the paper is structured as follows. In Section 2 we introduce some notations and give preliminaries. In Section 3, based on the generalized conditional correlation, the practical statistical model on smart distinguishers with side information is formalized and analyzed. In Section 4 we review the description of Bluetooth two-level E0 as well as the resynchronization flaw. In Section 5, correlations conditioned on input weights of E0 FSM are investigated. In Section 6, a key-recovery attack on two-level E0 is developed and optimized together with experimental results. Finally, we conclude in Section 7.

2 Notations and Preliminaries

Given the function $f : \mathcal{E} \to GF(2)^\ell$, define the distribution D_f of $f(X)$ with X uniformly distributed, i.e. $D_f(a) = \frac{1}{|\mathcal{E}|}\sum_{X\in\mathcal{E}} \mathbf{1}_{f(X)=a}$ for all $a \in GF(2)^\ell$. Following [5], recall that the Squared Euclidean Imbalance (SEI) of the distribution D_f is defined by

$$\Delta(D_f) = 2^\ell \sum_{a\in GF(2)^\ell} \left(D_f(a) - \frac{1}{2^\ell}\right)^2 . \tag{1}$$

For $\ell = 1$, it's easy to see that $\Delta(D_f)$ is closely related to the well known term *correlation*[7] $\epsilon(D_f)$ by $\Delta(D_f) = \epsilon^2(D_f)$. For brevity, we adopt the simplified notations $\epsilon(f), \Delta(f)$ to denote $\epsilon(D_f), \Delta(D_f)$ respectively hereafter. From the theory of hypothesis testing and Neyman-Pearson likelihood ratio (see [5]), $\Delta(f)$ tells us that the minimum number n of samples for an optimal distinguisher to effectively distinguish a sequence of n output samples of f from $(2^L - 1)$ truly random sequences of equal length is

$$n = \frac{4L\log 2}{\Delta(f)}. \tag{2}$$

Note that the result in Eq.(2) with $\ell = 1$ has long been known up to a constant factor $\frac{1}{2}$ in the theory of channel coding. In fact, correlation attacks has been very successful for almost two decades to apply the distinguisher that analyzes the biased sample of a single bit (i.e. the case $\ell = 1$) in order to reconstruct the L-bit key (or subkey), where only the right key can produce a biased sequence while all the wrong keys produce unbiased sequences. More recently, on the sound theoretical basis [5] of the generalized distinguisher, it was shown that this generalized distinguisher helps to improve the correlation attack when considering multi-biases simultaneously (for details see the key-recovery attack [23] on one-level E0 which halves the time and data complexities).

3 A Smart Distinguisher with Side Information

Given a function $f : GF(2)^u \times GF(2)^v \to GF(2)^r$, let $f_{\mathcal{B}}(X) = f(\mathcal{B}, X)$ for $\mathcal{B} \in GF(2)^u$ and $X \in GF(2)^v$, where the notation $f_{\mathcal{B}}(\cdot)$ is used to replace $f(\cdot)$ whenever $\mathcal{B}$ is given. Consider such a game between a player and an oracle. Each time the

[6] A resynchronization attack on stream cipher (a.k.a. the related-key attack) refers to the one that needs many frames of keystreams produced by different IVs (i.e. the public frame counter) and the same key in order to recover the key given the IVs.

[7] Correlation is commonly defined by $D_f(1) = \frac{1}{2} + \frac{\epsilon(D_f)}{2}$; and $|\epsilon(D_f)| \leq 1$ by this definition.

oracle secretly generates $\mathcal{B}, X$ independently and uniformly to compute $f_{\mathcal{B}}(X)$; the player, in turn, sends a guess on the current value of the partial input $\mathcal{B}$. Only when he guesses correctly, the oracle would output the value of $f_{\mathcal{B}}(X)$, otherwise, it would output a random and uniformly distributed $Z \in GF(2)^r$. Suppose the player somehow manages to collect 2^L sequences of n interaction samples with the following characteristics: one sequence has n samples $(f_{\mathcal{B}_i^{\mathcal{K}}}(X_i), \mathcal{B}_i^{\mathcal{K}})$ $(i = 1, \dots, n)$ where $\mathcal{B}_i^{\mathcal{K}}$'s and X_i's are independently and uniformly distributed; the remaining $(2^L - 1)$ sequences all consist of n independently and uniformly distributed random variables $(Z_i^K, \mathcal{B}_i^K)$ $(i = 1, \dots, n)$ for $K \neq \mathcal{K}$. One interesting question to the player is how to distinguish the biased sequence from the other sequences using the minimum number n of samples.

Note that the above problem is of special interest in key-recovery attacks, including the related-key attacks, where $\mathcal{B}_i^{\mathcal{K}}$'s are the key-related material (i.e. computable with the key and other random public parameters) and the oracle can be viewed as an intermediate computation process accessible to the attacker with only a limited number of queries. Thus, when the attacker knows the right key $\mathcal{K}$ he can collect n (hopefully biased) samples of f; on the other hand, if he uses the wrong key, he will only collect an unbiased sequence.

From Section 2, we know that the minimum number n of samples for the basic distinguisher which doesn't use the partial input $\mathcal{B}_i$'s is $n = 4L \log 2/\Delta(f)$. When the samples are incorporated with the $\mathcal{B}_i$'s, we can prove the following stronger result.

Theorem 1. *The smart distinguisher (in Algorithm 1) solves our above problem with*

$$n = \frac{4L \log 2}{E[\Delta(f_{\mathcal{B}})]} \tag{3}$$

and the time complexity $O(n \cdot 2^L)$, where the expectation is taken over all the uniformly distributed $\mathcal{B}$. Moreover, the distinguisher can achieve the optimal time complexity $O(n + L \cdot 2^{L+1})$ with precomputation $O(L \cdot 2^L)$ when $\mathcal{B}_i^K$'s and Z_i^K's can be expressed by:

$$\begin{aligned} \mathcal{B}_i^K &= \mathcal{L}(K) \oplus c_i\,, & (4)\\ Z_i^K &= \mathcal{L}'(K) \oplus c'_i \oplus g(\mathcal{B}_i^K)\,, & (5) \end{aligned}$$

for all L-bit K and $i = 1, 2, \dots, n$, where g is an arbitrary function, $\mathcal{L}, \mathcal{L}'$ are $GF(2)$-linear functions, and c_i's, c'_i's are independently and uniformly distributed which are known to the distinguisher.

Remark 2. Our smart distinguisher (Algorithm 1) turns out to be a derivative of the basic distinguisher in [5] and the result Eq.(3) for the simple case $r = 1$ was already pointed out (without proof) in [16] with a mere difference of a negligible constant term $2 \log 2 \approx 2^{0.47}$. Also note that the quantity $\mathrm{E}[\Delta(f_{\mathcal{B}})]$ in Eq.(3) measures the correlation of the output of an arbitrary function conditioned on the (partial) input which is uniformly distributed and *unknown*[8]. In contrast, prior to our work, the conditional correlation, that refers to the linear correlation of the inputs conditioned on a given (short) output pattern of a nonlinear function, was well studied in [1, 21, 22] based on a different statistical distance other than SEI. Highly motivated by the security of the nonlinear filter generator, their research focused on the case where the nonlinear function is the augmented nonlinear filter function (with small input size) and the inputs are the involved LFSR taps. Obviously, the notion of our

[8] According to the rule of our game, it's unknown to the distinguisher whether the sample $\mathcal{B}$ is the correct value used for the oracle to compute $f_{\mathcal{B}}(X)$ or not.

Algorithm 1 The smart distinguisher with side information

```
Parameters:
  $n$ set by Eq.(3)
  $D_{f_{\mathcal{B}}}$ for all $\mathcal{B} \in GF(2)^u$
Inputs:
  uniformly and independently distributed $u$-bit $\mathcal{B}_1^K, \dots, \mathcal{B}_n^K$ for all $L$-bit $K$
  $Z_1^{\mathcal{K}}, \dots, Z_n^{\mathcal{K}} = f_{\mathcal{B}_1^{\mathcal{K}}}(X_1), \dots, f_{\mathcal{B}_n^{\mathcal{K}}}(X_n)$ for one fixed $L$-bit $\mathcal{K}$ with uniformly and independently distributed $v$-bit vectors $X_1, \dots, X_n$
  uniformly and independently distributed sequences $Z_1^K, Z_2^K, \dots, Z_n^K$ for all $L$-bit $K$ such that $K \neq \mathcal{K}$
Goal: find $\mathcal{K}$
Processing:
  for all $L$-bit $K$ do
    $G(K) \leftarrow 0$
    for $i = 1, \dots, n$ do
      $G(K) \leftarrow G(K) + \log_2 \left(2^r \cdot D_{f_{\mathcal{B}_i^K}}(Z_i^K)\right)$
    end for
  end for
  output $\mathcal{K}$ that maximizes $G(\mathcal{K})$
```

conditional correlation can be seen as the generalized opposite of [1, 21, 22], that addresses the issue of how to make the most use of all the data for the success. In Section 6, Theorem 1 is directly applied to Bluetooth two-level E0 for a truly practical attack.

Proof (sketch). Let us introduce a new distribution D over $GF(2)^{r+u}$ from $D_{f_{\mathcal{B}}}$ defined by

$$D(\mathcal{B}, Z) = \frac{1}{2^u} D_{f_{\mathcal{B}}}(Z), \tag{6}$$

for all $\mathcal{B} \in GF(2)^u, Z \in GF(2)^r$. We can see that our original problem is transformed into that of the basic distinguisher to distinguish D from uniform distribution. According to Section 2, we need minimum $n = 4L \log 2 / \Delta(D)$. So we compute $\Delta(D)$ by Eq.(1,6):

$$\begin{aligned}
\Delta(D) &= 2^{r+u} \sum_{\mathcal{B} \in GF(2)^u} \sum_{Z \in GF(2)^r} \left(D(\mathcal{B}, Z) - \frac{1}{2^{r+u}}\right)^2 \\
&= 2^{r+u} \sum_{\mathcal{B} \in GF(2)^u} \sum_{Z \in GF(2)^r} \left(\frac{1}{2^u} D_{f_{\mathcal{B}}}(Z) - \frac{1}{2^{r+u}}\right)^2 \\
&= 2^{-u} \sum_{\mathcal{B} \in GF(2)^u} 2^r \sum_{Z \in GF(2)^r} \left(D_{f_{\mathcal{B}}}(Z) - \frac{1}{2^r}\right)^2 \\
&= \mathrm{E}[\Delta(f_{\mathcal{B}})]. \qquad (7)
\end{aligned}$$

Meanwhile, the best distinguisher tries to maximize the probability $\prod_{i=1}^n D(\mathcal{B}_i, Z_i)$, i.e. the conditioned probability $\prod_{i=1}^n D_{f_{\mathcal{B}_i}}(Z_i)$. As the conventional approach, we know that this is equivalent to maximize $G = \sum_{i=1}^n \log_2(2^r \cdot D_{f_{\mathcal{B}_i}}(Z_i))$ as shown in Algorithm 1. The time complexity of the distinguisher[9] is obviously $O(n \cdot 2^L)$.

[9] In this paper, we only discuss the deterministic distinguisher. For the probabilistic distinguisher, many efficient and general decoding techniques (e.g. the probabilistic iterative decoding), which are successful in correlation attacks, were carefully presented in the related work [22] and such techniques also apply to our distinguisher.

Now, to show how to optimize the time complexity of the smart distinguisher when $\mathcal{B}_i^K$'s and Z_i^K's exhibit the special structure of Eq.(4, 5) for the second part of the theorem, let us first introduce two functions $\mathcal{H}, \mathcal{H}'$:

$$\mathcal{H}(K) = \sum_{i=1}^{n} \mathbf{1}_{\mathcal{L}(K)=c_i \text{ and } \mathcal{L}'(K)=c'_i} \tag{8}$$

$$\mathcal{H}'(K) = \log_2 \left(2^r \cdot D_{f_{\mathcal{L}(K)}} \left(\mathcal{L}'(K) \oplus g\left(\mathcal{L}(K)\right)\right)\right) \tag{9}$$

for $K \in GF(2)^L$. We can see that $G(K)$ computed in Line 7 to 10, Algorithm 1 is nothing but a simple convolution (denoted by $\otimes$) between $\mathcal{H}$ and $\mathcal{H}'$:

$$G(K) = (\mathcal{H} \otimes \mathcal{H}')(K) \overset{\text{def}}{=} \sum_{K' \in GF(2)^L} \mathcal{H}(K')\mathcal{H}'(K \oplus K'), \tag{10}$$

for all $K \in GF(2)^L$. It's known that convolution and Walsh transform (denoted by the hat symbol) are transformable, so we have

$$G(K) = \frac{1}{2^L} \widehat{\widehat{\mathcal{H} \otimes \mathcal{H}'}}(K) = \frac{1}{2^L} \widehat{\mathcal{H}''}(K), \tag{11}$$

where $\mathcal{H}''(K) = \widehat{\mathcal{H}}(K) \cdot \widehat{\mathcal{H}'}(K)$. This means that after computing $\mathcal{H}$ and $\mathcal{H}'$, the time complexity of our smart distinguisher would be dominated by three times of FWT, i.e. $\widehat{\mathcal{H}}, \widehat{\mathcal{H}'}, \widehat{\mathcal{H}''}$ in $O(3L \cdot 2^L)$. Moreover, since only c_i's, c'_i's may vary from one run of the attack to another, which are independent of $\mathcal{H}'$, we can also precompute $\widehat{\mathcal{H}'}$ and store it in the table; finally, the real-time processing only takes time $O(n + L \cdot 2^{L+1})$. □

Property 3. We have

$$\mathrm{E}[\varDelta(f_{\mathcal{B}})] \geq \varDelta(f),$$

where equality holds if and only if (iff) $D_{f_{\mathcal{B}}}$ is independent of $\mathcal{B}$.

For $r = 1$, this can be easily shown as follows. From Section 2, we have $\mathrm{E}[\varDelta(f_{\mathcal{B}})] = \mathrm{E}[\epsilon^2(f_{\mathcal{B}})] \geq E^2[\epsilon(f_{\mathcal{B}})] = \epsilon^2(f) = \varDelta(f)$ where equality holds iff $\epsilon(f_{\mathcal{B}})$ is independent of $\mathcal{B}$. In Appendix, we give the complete proof for the general case $\mathrm{E}[\varDelta(f_{\mathcal{B}})] \geq \varDelta(f)$.

Remark 4. As $\mathrm{E}[\varDelta(f_{\mathcal{B}})], \varDelta(f)$ measures the conditional correlation and the unconditional correlation respectively, this property convinces us that the former is no smaller than the latter. This relationship between the conditional correlation and the unconditional correlation was informally conjectured in [22]. We conclude from Eq.(3) that the smart distinguisher having partial (or side) information (i.e. $\mathcal{B}$ herein) about the biased source generator (i.e. $f_{\mathcal{B}}$ herein) always works better than the basic distinguisher governing no knowledge of that side information, as long as the generator is statistically dependent on the side information. Our result verifies the intuition that the more the distinguisher knows about the generation of the biased source, the better it works. In particular, Property 3 implies that even if the fact that $\varDelta(f) = 0$ causes the basic distinguisher to be completely useless as it needs infinite data complexity, in contrast, the smart distinguisher would still work as long as $D_{f_{\mathcal{B}}}$ is dependent on $\mathcal{B}$, i.e. $\mathrm{E}[\varDelta(f_{\mathcal{B}})] > 0$. In Section 5, we give two illustrative examples $\mathrm{E}[\varDelta(f_{\mathcal{B}})]$ on the core of Bluetooth E0 to be compared with their counterparts $\varDelta(f)$.

4 Review on Bluetooth Two-level E0

The core (Fig. 1) of Bluetooth keystream generator E0 (also called one-level E0) consists of four regularly-clocked LFSRs of a total 128 bits and a Finite State

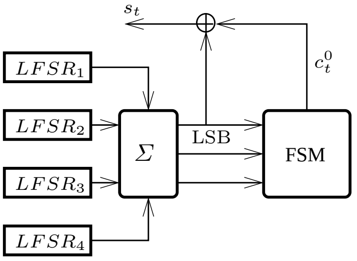

Fig. 1. The core of Bluetooth keystream generator E0

Machine (FSM) of 4 bits. Denote $B_t \in GF(2)^4$ the four output bits of LFSRs at time instance t, and $X_t \in GF(2)^4$ the FSM state at time instance t. Note that X_t contains the bit c_t^0 as well as the bit c_{t-1}^0 (due to the effect of a delay cell inside the FSM). Also note that the computation of the FSM next state X_{t+1} only depends on its current state X_t together with the Hamming weight $w(B_t)$ of B_t. At each time instance t, the core produces one bit $s_t = (w(B_t) \mod 2) \oplus c_t^0$, and then updates the states of LFSRs and FSM.

According to the Bluetooth standard [6], this core is used with a two-level initialization scheme to produce the keystream for encryption. That is, after a first initialization of LFSRs by an affine transformation of the effective encryption key $\mathcal{K}$ and the public nonce[10] $\mathcal{P}^i$ for the i-th frame, E0 runs at level one, whose last 128 output bits are permuted into LFSRs at level two for reinitialization; then E0 runs at level two to produce the final keystream $z_{t'}^i$ for $t' = 1, 2, \ldots, 2745$ (for clarity, we refer the time instance t and t' to the context of E0 level one and E0 level two respectively).

In order to review the reinitialization flaw discovered in [24], we first introduce some notations. Define the binary vector $\gamma = (\gamma_0, \gamma_1, \ldots, \gamma_{\ell-1})$ of length $|\gamma| = \ell \geq 3$ with $\gamma_0 = \gamma_{\ell-1} = 1$ and let $\bar{\gamma} = (\gamma_{\ell-1}, \gamma_{\ell-2}, \ldots, \gamma_0)$ represent the vector in reverse order of γ. Given ℓ and t, for the one-level E0, we define $\mathcal{B}_{t+1} = B_{t+1}B_{t+2}\ldots B_{t+\ell-2}$ and $C_t = (c_t^0, \ldots, c_{t+\ell-1}^0)$. Then, the function $h_{\mathcal{B}_{t+1}}^{\gamma} : X_{t+1} \mapsto \gamma \cdot C_t$ is well defined[11] for all t, where the dot operator between two vectors represents the inner $GF(2)$-product. Now, we let $(\mathcal{B}_{t+1}^i, X_{t+1}^i)$ (resp. $(\mathcal{B}_{t'+1}^i, X_{t'+1}^i)$) control the FSM to compute C_t^i (resp. $C_{t'}^i$) at E0 first (resp. second) level for the i-th frame. Note that initialization of LFSRs at E0 level one by an affine transformation of $\mathcal{K}, \mathcal{P}^i$ can be expressed by

$$\mathcal{B}_t^i = \mathcal{G}_t(\mathcal{K}) \oplus \mathcal{G}_t'(\mathcal{P}^i), \tag{12}$$

where $\mathcal{G}_t, \mathcal{G}_t'$ are public linear functions (which are dependent on ℓ but omitted from notations for simplicity). Moreover, we let $Z_{t'}^i = (z_{t'}^i, \ldots, z_{t'+\ell-1}^i)$. Then, as pointed out and detailed in [24], the critical reinitialization flaw of Bluetooth two-level E0 can be expressed as

$$\bar{\gamma} \cdot (Z_{t'}^i \oplus \mathcal{L}_{t'}(\mathcal{K}) \oplus \mathcal{L}_{t'}'(\mathcal{P}^i)) = \bigoplus_{j=1}^{4} (\gamma \cdot C_{t_j}^i) \oplus (\bar{\gamma} \cdot C_{t'}^i), \tag{13}$$

[10] $\mathcal{P}^i$ includes a 26-bit counter and some user-dependent constant.

[11] because c_t^0, c_{t+1}^0 are contained in X_{t+1} already and we can compute $c_{t+2}^0, \ldots, c_{t+\ell-1}^0$ by $\mathcal{B}_{t+1}, X_{t+1}$. Actually, the prerequisite $\gamma_0 = \gamma_{\ell-1} = 1$ on γ is to guarantee that knowledge of $\mathcal{B}_{t+1}, X_{t+1}$ is necessary and sufficient to compute $\gamma \cdot C_t$.

for any i and γ of length ℓ such that $3 \leq \ell \leq 8$, and $t' \in \bigcup_{k=0}^{2}\{8k+1, \ldots, 8k+9-\ell\}$, where $t_1, \ldots, t_4$ are functions[12] in terms of t' only, and $C_{t_1}^i, \ldots, C_{t_4}^i$ share no common coordinate, and $\mathcal{L}_{t'}, \mathcal{L}'_{t'}$ are fixed linear functions which can be expressed by t', ℓ from the standard. By definition of h, Eq.(13) can be put equivalently as:

$$\bar{\gamma} \cdot (Z_{t'}^i \oplus \mathcal{L}_{t'}(\mathcal{K}) \oplus \mathcal{L}'_{t'}(\mathcal{P}^i)) = \bigoplus_{j=1}^{4} h_{\mathcal{B}_{t_j+1}^i}^{\gamma}(X_{t_j+1}^i) \oplus h_{\mathcal{B}_{t'+1}^i}^{\bar{\gamma}}(X_{t'+1}^i), \qquad (14)$$

for any i, any γ with $3 \leq \ell \leq 8$ and $t' \in \bigcup_{k=0}^{2}\{8k+1, \ldots, 8k+9-\ell\}$. Note that the usage of the bar operator actually reflects the fact that the loading of LFSRs at E0 level two for reinitialization is in reverse order of the keystream output at E0 level one.

5 Correlations Conditioned on Input Weights of FSM

Recall it has been observed in [23] that if $w(B_t)w(B_{t+1})w(B_{t+2})w(B_{t+3}) = 2222$ is satisfied, then, we always have

$$c_t^0 \oplus c_{t+1}^0 \oplus c_{t+2}^0 \oplus c_{t+3}^0 \oplus c_{t+4}^0 = 1. \qquad (15)$$

Let $\alpha_t = \gamma \cdot C_t$ with $\gamma = (1,1,1,1,1)$ and $\ell = 5$. Thus α_t is the sum on the left-hand side of Eq.(15). From Section 4 we know that given $\mathcal{B}_{t+1} = B_{t+1}B_{t+2}B_{t+3} \in GF(2)^{12}$, the function $h_{\mathcal{B}_{t+1}}^{\gamma} : X_{t+1} \mapsto \alpha_t$ is well defined for all t. Let $W(\mathcal{B}_{t+1}) \stackrel{\text{def}}{=} w(B_{t+1})w(B_{t+2}) \cdots w(B_{t+\ell-2})$. Thereby, we deduct from [23] that $\alpha_t = 1$ whenever $W(\mathcal{B}_{t+1}) = 222$. In contrast to the (unconditional) correlation as mentioned in Section 2, we call it a conditional correlation[13], i.e. the correlation $\epsilon(h_{\mathcal{B}_{t+1}}^{\gamma}) = 1$ conditioned on $W(\mathcal{B}_{t+1}) = 222$.

This motivates us to study the general correlation $\epsilon(h_{\mathcal{B}_{t+1}}^{\gamma})$ conditioned on $\mathcal{B}_{t+1}$, or more precisely $W(\mathcal{B}_{t+1})$, when X_{t+1} is uniformly distributed. All the non-zero conditional correlations $\epsilon(h_{\mathcal{B}_{t+1}}^{\gamma})$ are shown in Table 1 in descending order of the absolute value, where $|\mathcal{B}_{t+1}|$ denotes the cardinality of $\mathcal{B}_{t+1}$ admitting any weight triplet in the group. As the unconditioned correlation $\epsilon(h^{\gamma})$ of the bit α_t always equals the mean value[14] $\mathrm{E}[\epsilon(h_{\mathcal{B}_{t+1}}^{\gamma})]$ over the uniformly distributed $\mathcal{B}_{t+1}$, we can use Table 1 to verify $\epsilon(h^{\gamma}) = \frac{25}{256}$ (denote this value[15] by λ). Let $f_{\mathcal{B}} = h_{\mathcal{B}_{t+1}}^{\gamma}$ with $\gamma = (1,1,1,1,1)$. Now, to verify Property 3 in Section 3 we compute $\mathrm{E}[\Delta(f_{\mathcal{B}})] = \frac{544}{2^{12}} \approx 2^{-2.9}$, which is significantly larger than $\Delta(f) = \lambda^2 \approx 2^{-6.67}$. As another example, consider now $f_{\mathcal{B}} = h_{\mathcal{B}_{t+1}}^{\gamma}$ with $\gamma = (1,1,0,1)$ and $u = 8, v = 4, r = 1$. Similarly, the conditioned correlation of the corresponding sum $c_t^0 \oplus c_{t+1}^0 \oplus c_{t+3}^0$ (denoted by α'_t) is shown in Table 2. From Table 2, we get a quite large $\mathrm{E}[\Delta(f_{\mathcal{B}})] =$

[12] additionally, given t', the relation $t_1 < t_2 < t_3 < t_4$ always holds that satisfies $t_2 - t_1 = t_4 - t_3 = 8$ and $t_3 - t_2 \geq 32$.

[13] Note that earlier in [16], correlations conditioned on keystream bits (both with and without one LFSR outputs) were well studied for one-level E0, which differ from our conditional correlations and do not fit in the context of two-level E0 if the initial state of E0 is not recovered level by level.

[14] Note that $\mathrm{E}[\epsilon(h_{\mathcal{B}_{t+1}}^{\gamma})]$ is computed by an exhaustive search over all possible $X_{t+1} \in GF(2)^4$, $\mathcal{B}_{t+1} \in GF(2)^{12}$ and thus does not depend on t.

[15] this unconditional correlation was discovered by [12, 16] and proved later on by [23] to be one of the two largest unconditioned correlations up to 26-bit output sequence of the FSM.

2^{-3} as well; in contrast, we can check that as already pointed out in Section 3, the unconditional correlation[16] $\Delta(f) = 0$ from Table 2.

Table 1. Weight triplets to generate the biased bit α_t with $\gamma = (1,1,1,1,1)$ and $\ell = 5$

bias of α_t $\epsilon(h^{\gamma}_{\mathcal{B}_{t+1}})$	weight triplet(s) $W(\mathcal{B}_{t+1})$	cardinality $\lvert\mathcal{B}_{t+1}\rvert$
-1	220, 224	72
1	222	216
-0.5	120, 124, 210, 214 230, 234, 320, 324	192
0.5	122, 212, 322, 232	576
-0.25	110, 111, 114, 130 131, 134, 310, 311 314, 330, 331, 334	384
0.25	112, 113, 132, 133 312, 313, 332, 333	640

Table 2. Weight pairs to generate the biased bit α'_t with $\gamma = (1,1,0,1)$ and $\ell = 4$

bias of α'_t $\epsilon(h^{\gamma}_{\mathcal{B}_{t+1}})$	weight pairs $W(\mathcal{B}_{t+1})$	cardinality $\lvert\mathcal{B}_{t+1}\rvert$
-1	01, 43	8
1	03, 41	8
-0.5	11, 33	32
0.5	13, 31	32

6 Key-recovery Attack on Bluetooth Two-level E0

6.1 Basic Idea

Given the binary vector γ (to be determined later) with $3 \leq \ell \leq 8$, for all $\mathcal{B} \in GF(2)^{4(\ell-2)}$ such that $\epsilon(h^{\gamma}_{\mathcal{B}}) \neq 0$, define the function

$$g^{\gamma}(\mathcal{B}) = \begin{cases} 1, & \text{if } \epsilon(h^{\gamma}_{\mathcal{B}}) > 0 \\ 0, & \text{if } \epsilon(h^{\gamma}_{\mathcal{B}}) < 0 \end{cases}$$

to estimate the effective value of $h^{\gamma}_{\mathcal{B}}(X)$ (defined in Section 4) for some unknown $X \in GF(2)^4$. For a fixed $t' \in \bigcup_{k=0}^{2}\{8k+1, \ldots, 8k+9-|\gamma|\}$, let us guess the subkey $K_1 \overset{\text{def}}{=} (\mathcal{G}_{t_1}(\mathcal{K}), \ldots, \mathcal{G}_{t_4}(\mathcal{K}))$ of $16(\ell-2)$ bits by $\widehat{K_1}$ and the one-bit subkey $K_2 \overset{\text{def}}{=} \bar{\gamma} \cdot \mathcal{L}_{t'}(\mathcal{K})$ by $\widehat{K_2}$. We set $K = (K_1, K_2)$, $\widehat{K} = (\widehat{K_1}, \widehat{K_2})$. As $\mathcal{P}^i$'s are public, for every frame i, we can use Eq.(12) to compute the estimate $\widehat{\mathcal{B}^i_{t_j+1}}$ for $\mathcal{B}^i_{t_j+1}$ for

[16] Note that on the other hand the unconditional correlation $\epsilon(h^{\gamma}) = 2^{-4}$ with $\gamma = (1,0,1,1)$ (denote this value by λ'), shown first in [17], was proved by [23] to be the only second largest unconditioned correlations up to 26-bit output sequence of the FSM.

$j = 1, \ldots, 4$ with $\widehat{K_1}$. Denote

$$\begin{aligned}\mathcal{B}^i &= (\mathcal{B}^i_{t_1+1}, \mathcal{B}^i_{t_2+1}, \mathcal{B}^i_{t_3+1}, \mathcal{B}^i_{t_4+1}),\\ \mathcal{X}^i &= (X^i_{t_1+1}, X^i_{t_2+1}, X^i_{t_3+1}, X^i_{t_4+1}, X^i_{t'+1}, \mathcal{B}^i_{t'+1}, \widehat{K}).\end{aligned}$$

Define the probabilistic mapping $\mathcal{F}^{\gamma}_{\mathcal{B}^i}(\mathcal{X}^i)$ to be a truly random bit with uniform distribution for all i such that $\prod_{j=1}^{4} \epsilon(h^{\gamma}_{\widehat{\mathcal{B}^i_{t_j+1}}}) = 0$; otherwise, we let

$$\mathcal{F}^{\gamma}_{\mathcal{B}^i}(\mathcal{X}^i) = \bigoplus_{j=1}^{4} \left(h^{\gamma}_{\mathcal{B}^i_{t_j+1}}(X^i_{t_j+1}) \oplus g^{\gamma}(\widehat{\mathcal{B}^i_{t_j+1}}) \right) \oplus h^{\bar{\gamma}}(\mathcal{B}^i_{t'+1}, X^i_{t'+1}). \tag{16}$$

Note that given $\widehat{K_2}$, $\mathcal{F}^{\gamma}_{\mathcal{B}^i}(\mathcal{X}^i)$ is accessible in the latter case as we have

$$\mathcal{F}^{\gamma}_{\mathcal{B}^i}(\mathcal{X}^i) = \bar{\gamma} \cdot \left(Z^i_{t'} \oplus \mathcal{L}'_{t'}(\mathcal{P}^i) \right) \oplus \widehat{K_2} \oplus \bigoplus_{j=1}^{4} g^{\gamma}(\widehat{\mathcal{B}^i_{t_j+1}}),$$

for all i such that $\prod_{j=1}^{4} \epsilon(h^{\gamma}_{\widehat{\mathcal{B}^i_{t_j+1}}}) \neq 0$ according to Eq.(14). With the correct guess $\widehat{K} = K$, Eq.(16) reduces to

$$\mathcal{F}^{\gamma}_{\mathcal{B}^i}(\mathcal{X}^i) = \bigoplus_{j=1}^{4} \left(h^{\gamma}_{\mathcal{B}^i_{t_j+1}}(X^i_{t_j+1}) \oplus g^{\gamma}(\mathcal{B}^i_{t_j+1}) \right) \oplus h^{\bar{\gamma}}(\mathcal{B}^i_{t'+1}, X^i_{t'+1}), \tag{17}$$

for all i such that $\prod_{j=1}^{4} \epsilon(h^{\gamma}_{\mathcal{B}^i_{t_j+1}}) \neq 0$. As the right-hand side of Eq.(17) only involves the unknown $X^i = (X^i_{t_1+1}, X^i_{t_2+1}, X^i_{t_3+1}, X^i_{t_4+1}, X^i_{t'+1}, \mathcal{B}^i_{t'+1})$, we denote the mapping in this case by $f^{\gamma}_{\mathcal{B}^i}(X^i)$. With appropriate choice of γ as discussed in the next subsection, we can have $\mathrm{E}[\Delta(f^{\gamma}_{\mathcal{B}^i})] > 0$. With each wrong guess $\widehat{K} \neq K$, however, as shown in Appendix, we estimate $\mathcal{F}^{\gamma}_{\mathcal{B}^i}(\mathcal{X}^i)$ to be uniformly and independently distributed for all i (i.e. $\mathrm{E}[\Delta(\mathcal{F}^{\gamma}_{\mathcal{B}^i})] = 0$).

As we are interested in small ℓ for low time complexity, e.g. $|\ell| < 6$ as explained immediately next, we can assume from this constraint[17] that X^i's are uniformly distributed and that all X^i's, $\mathcal{B}^i$'s are independent. Submitting 2^L sequences of n pairs $(\mathcal{F}^{\gamma}_{\mathcal{B}^i}(\mathcal{X}^i), \widehat{\mathcal{B}^i})$ (for $i = 1, 2, \ldots, n$) to the distinguisher, we can fit in the smart distinguisher of Section 3 with $L = 16(\ell-2)+1, u = 16(\ell-2), v = 20+4(\ell-2), r = 1$ and expect it to successfully recover L-bit K with data complexity n sufficiently large as analyzed later. Note that the favourable $L < 64$ necessitates that $\ell < 6$.

6.2 Complexity Analysis and Optimization

From Eq.(3) in Section 3, the smart distinguisher needs data complexity

$$n = \frac{4L \log 2}{\mathrm{E}\left[\Delta\left(f^{\gamma}_{\mathcal{B}^i}\right)\right]}. \tag{18}$$

To compute n, we introduce another probabilistic mapping $f^{'\gamma}_{\mathcal{B}^i}$ similar to $f^{\gamma}_{\mathcal{B}^i}$:

$$f^{'\gamma}_{\mathcal{B}^i}(\mathcal{X}^i) \stackrel{\text{def}}{=} \bigoplus_{j=1}^{4} h^{\gamma}_{\mathcal{B}^i_{t_j+1}}(X^i_{t_j+1}) \oplus h^{\bar{\gamma}}(\mathcal{B}^i_{t'+1}, X^i_{t'+1}). \tag{19}$$

[17] however, the assumption does not hold for $\ell = 7, 8$: with $\ell = 8$, we know that $X^i_{t_2+1}$ is fixed given $X^i_{t_1+1}$ and $\mathcal{B}^i_{t_1+1}$ as we have $t_2 = t_1 + 8$ from Section 4; with $\ell = 7$, two bits of $X^i_{t_2+1}$ are fixed given $X^i_{t_1+1}$ and $\mathcal{B}^i_{t_1+1}$. Similar statements hold concerning $X^i_{t_3+1}, \mathcal{B}^i_{t_3+1}$ and $X^i_{t_4+1}$.

Theorem 5. *For all $\mathcal{B}^i = (\mathcal{B}^i_{t_1+1}, \mathcal{B}^i_{t_2+1}, \mathcal{B}^i_{t_3+1}, \mathcal{B}^i_{t_4+1}) \in GF(2)^{16(\ell-2)}$, we always have*

$$\Delta(f^{\gamma}_{\mathcal{B}^i}) = \Delta(f^{'\gamma}_{\mathcal{B}^i}).$$

Proof. This is trivial for the case where $\prod_{j=1}^4 \epsilon(h^{\gamma}_{\mathcal{B}^i_{t_j+1}}) = 0$, because by definition $D_{f^{\gamma}_{\mathcal{B}^i}}$ is a uniform distribution and so is $D_{f^{'\gamma}_{\mathcal{B}^i}}$ by the famous Piling-up lemma (see [25]). Let us discuss the case where $\prod_{j=1}^4 \epsilon(h^{\gamma}_{\mathcal{B}^i_{t_j+1}}) \neq 0$. In this case we know that given $\mathcal{B}^i$, $\bigoplus_{j=1}^4 g^{\gamma}(\mathcal{B}^i_{t_j+1})$ is well-defined and it is a fixed value that doesn't depend on the unknown X^i. Consequently, we have $\Delta(f^{\gamma}_{\mathcal{B}^i}) = \Delta(f^{'\gamma}_{\mathcal{B}^i} \oplus \text{const.}) = \Delta(f^{'\gamma}_{\mathcal{B}^i})$. □

We can use Theorem 5 to compute $\frac{4L\log 2}{n}$ from Eq.(18) as $\frac{4L\log 2}{n} = \mathrm{E}[\Delta(f^{\gamma}_{\mathcal{B}^i})] = \mathrm{E}[\Delta(f^{'\gamma}_{\mathcal{B}^i})]$. Next, the independence of $\mathcal{B}^i$'s allows us to apply Piling-up Lemma [25] to continue as follows,

$$\frac{4L\log 2}{n} = \mathrm{E}\left[\Delta(h^{\bar{\gamma}}) \prod_{j=1}^{4} \Delta\left(h^{\gamma}_{\mathcal{B}^i_{t_j+1}}\right)\right] = \Delta(h^{\bar{\gamma}}) \prod_{j=1}^{4} \mathrm{E}\left[\Delta\left(h^{\gamma}_{\mathcal{B}^i_{t_j+1}}\right)\right].$$

Because we know from Section 5 that $\mathrm{E}[\Delta(h^{\gamma}_{\mathcal{B}^i_{t+1}})]$ does not depend on t and i, we finally have

$$\frac{4L\log 2}{n} = \Delta(h^{\bar{\gamma}}) \cdot \mathrm{E}^4\left[\Delta\left(h^{\gamma}_{\mathcal{B}_{t+1}}\right)\right]. \tag{20}$$

As we want to minimize n, according to Eq.(18), we would like to find some γ ($3 \leq |\gamma| < 6$) such that $\mathrm{E}[\Delta(f^{\gamma}_{\mathcal{B}^i})]$ is large, and above all, strictly positive. In order to have $\mathrm{E}[\Delta(f^{\gamma}_{\mathcal{B}^i})] > 0$, we must have $\Delta(h^{\bar{\gamma}}) > 0$ first, by Eq.(20). According to results of [16, 17, 12, 23], only two aforementioned choices satisfy our predefined prerequisite about γ (i.e. both the first and last coordinates of γ are one): either $\gamma = (1,1,1,1,1)$ with $\Delta(h^{\bar{\gamma}}) = \lambda^2 \approx 2^{-6.71}$, or $\gamma = (1,1,0,1)$ with $\Delta(h^{\bar{\gamma}}) = \lambda'^2 = 2^{-8}$. For $\gamma = (1,1,1,1,1)$, from last section, we know that $\mathrm{E}[\Delta(h^{\gamma}_{\mathcal{B}_{t+1}})] \approx 2^{-2.9}$. So we conclude from Eq.(20) that $n \approx 2^{25.4}$ frames of keystreams generated by the same key $\mathcal{K}$ suffice to recover the $L = 49$-bit subkey K. Analogously, for $\gamma = (1,1,0,1)$, we have $\mathrm{E}[\Delta(h^{\gamma}_{\mathcal{B}_{t+1}})] = 2^{-3}$ from last section. And it results in $n \approx 2^{26.5}$ frames to recover $L = 33$-bit subkey.

Let us discuss the time complexity of the attack now. For all $J = (J_1, J_2) \in GF(2)^{L-1} \times GF(2)$, and let $J_1 = (J_{1,1}, \ldots, J_{1,4})$ where $J_{1,i} \in GF(2)^{4(\ell-2)}$, we define $\mathcal{H}, \mathcal{H}'$:

$$\mathcal{H}(J) = \textstyle\sum_{i=1}^n \mathbf{1}_{\mathcal{G}'_{t_1}(\mathcal{P}^i),\ldots,\mathcal{G}'_{t_4}(\mathcal{P}^i)=J_1 \text{ and } \bar{\gamma}\cdot(Z^i_{t'} \oplus \mathcal{L}'_{t'}(\mathcal{P}^i))=J_2},$$

$$\mathcal{H}'(J) = \begin{cases} 0, & \text{if } \prod_{i=1}^4 \epsilon(h^{\gamma}_{J_{1,i}}) = 0 \\ \log 2^r \cdot D_{J_1}\left(J_2 \oplus \bigoplus_{i=1}^4 g^{\gamma}(J_{1,i})\right), & \text{otherwise} \end{cases}$$

where $D_{J_1} = D_{h^{\gamma}_{J_{1,1}}} \otimes D_{h^{\gamma}_{J_{1,2}}} \otimes D_{h^{\gamma}_{J_{1,3}}} \otimes D_{h^{\gamma}_{J_{1,4}}}$. Let $\mathcal{H}''(K) = \widehat{\mathcal{H}}(K) \cdot \widehat{\mathcal{H}'}(K)$. By Theorem 1 in Section 3, we have $G(K) = \frac{1}{2^L}\widehat{\mathcal{H}''}(K)$. This means that after precomputing $\widehat{\mathcal{H}'}$ in time $O(L \cdot 2^L)$, our partial key-recovery attack would be dominated by twice FWT, i.e. $\widehat{\mathcal{H}}, \widehat{\mathcal{H}''}$ with time $O(L \cdot 2^{L+1})$. Algorithm 2 illustrates the above basic partial key-recovery attack. Note that without the optimization technique of Theorem 1, the deterministic smart distinguisher has to perform $O(n \cdot 2^L)$ operations otherwise, which makes our attack impractical.

Algorithm 2 The basic partial key-recovery attack on two-level E0

```
Parameters:
1: $\gamma, t', t_1, t_2, t_3, t_4, L$
2: $n$ set by Eq.(20)
Inputs:
3: $\mathcal{P}^i$ for $i = 1, 2, \ldots, n$
4: $Z^i_{t'}$ for $i = 1, 2, \ldots, n$
Preprocessing:
5: compute $H', \widehat{H'}$
Processing:
6: compute $H, \widehat{H}$
7: compute $H'' = \widehat{H} \cdot \widehat{H'}$ and $\widehat{H''}$
8: output $K$ with the maximum $\widehat{H''}(K)$
```

Furthermore, by Table 2, we discovered a special property

$$\epsilon(h^{\gamma}_{B_{t+1}B_{t+2}}) \equiv \epsilon(h^{\gamma}_{\overline{B}_{t+1}\overline{B}_{t+2}}) \equiv -\epsilon(h^{\gamma}_{\overline{B}_{t+1}B_{t+2}}) \equiv -\epsilon(h^{\gamma}_{B_{t+1}\overline{B}_{t+2}}) \qquad (21)$$

for all $\mathcal{B}_{t+1} = B_{t+1}B_{t+2} \in GF(2)^8$ with $\gamma = (1, 1, 0, 1)$, where the bar operator denotes the bitwise complement of the 4-bit binary vector. This means that for our 33-bit partial key-recovery attack, we always have $4^4 = 256$ equivalent key candidates[18] (see Appendix for details), which helps to decrease the computation time on $\widehat{H''}$ (see [23]) from $33 \times 2^{33} \approx 2^{38}$ to $25 \times 2^{25} \approx 2^{30}$. In total we have the running time $2^{38} + 2^{30} \approx 2^{38}$ for Algorithm 2.

We have implemented the full Algorithm 2 with $\gamma = (1, 1, 0, 1), t' = 1, n = 2^{26}$ frames (slightly less than the theoretical estimate $2^{26.5}$) on the Linux platform, 2.4G CPU, 2G RAM, 128GB hard disk. It turned out that after one run of a 37-hour precomputation (i.e. Line **5** in Algorithm 2 which stores a 64GB table in the hard disk), of all the 30 runs tested so far, our attack *never fails to successfully* recover the right 25-bit key in about 19 hours. Computing $H, \widehat{H}, H'', \widehat{H''}$ takes time 27 minutes, 18 hours, 45 minutes and 20 seconds respectively. The running time is dominated by FWT[19] $\widehat{H}$, which only takes a negligible portion of CPU time and depends dominantly on the performance of the hardware, i.e. the external data transfer rate[20] between the hard disk and PC's main memory.

Inspired by the multi-bias analysis on the traditional distinguisher in [23], the advanced multi-bias analysis (see Appendix) which is an extension of this section allows us to reach the data complexity $n \approx 2^{23.8}$ frames with the same time complexity. Once we recover the first $(33 - 8) = 25$-bit subkey, we just increment (or decrement) t' by one and use the knowledge of those subkey bits to reiterate Algorithm 2 to recover more key bits similarly as was done in [24]. Since only 17 new key bits are involved, which reduce to the 13-bit equivalent key, it's much faster to recover those key bits. Finally, we perform an exhaustive search over the equivalent key candidates in negligible time, whose total number is upper bounded by $2^{\frac{8|\mathcal{K}|}{32}} = 2^{\frac{|\mathcal{K}|}{4}}$. The final complexity of the complete key-recovery attack is bounded by one run of Algorithm 2, i.e. $O(2^{38})$. Table 3 compares our attacks with the best known attacks [13, 14, 16, 24] on two-level E0 for effective key size $|\mathcal{K}| = 128$. Note that with $|\mathcal{K}| = 64$, Bluetooth key loading at E0 level one makes the bits of the subkey K linearly independent for all $t' \in \bigcup_{k=0}^{2}\{8k+1, \ldots, 8k+5\}$. Therefore, the attack complexities remain to be on the same order.

[18] The term "equivalent key candidate" is exclusively used for our attack, which doesn't mean that they are equivalent keys for the Bluetooth encryption.

[19] The result is stored in a 32GB table in the hard disk.

[20] In our PC it is 32MB/s, which is common in the normal PC nowadays.

Table 3. Comparison of our attacks with the best attacks on two-level E0 for $|\mathcal{K}| = 128$

Attack		Precomputation	Time	Frames	Data	Memory
Fluhrer-Lucks	[13]	-	2^{73}	-	2^{43}	2^{51}
Fluhrer	[14]	2^{80}	2^{65}	2	$2^{12.4}$	2^{80}
Golić et al.	[16]	2^{80}	2^{70}	45	2^{17}	2^{80}
Lu-Vaudenay	[24]	-	2^{40}	2^{35}	$2^{39.6}$	2^{35}
Our Attacks	basic	2^{38}	2^{38}	$2^{26.5}$	$2^{31.1}$	2^{33}
	advanced	2^{38}	2^{38}	$2^{23.8}$	$2^{28.4}$	2^{33}

7 Conclusion

In this paper, we have generalized the concept of conditional correlations in [1, 21, 22] to study conditional correlation attacks against stream ciphers and other cryptosystems, in case the computation of the output allows for side information related to correlations conditioned on the input. A general framework has been developed for smart distinguishers, which exploit those generalized conditional correlations. In particular, based on the theory of the traditional distinguisher [5] we derive the number of samples necessary for a smart distinguisher to succeed. It is demonstrated that the generalized conditional correlation is no smaller than the unconditional correlation. Consequently, the smart distinguisher improves on the traditional basic distinguisher (in the worst case the smart distinguisher degrades into the traditional one); the smart distinguisher could be efficient even if no ordinary correlations exist. As an application of our generalized conditional correlations, a conditional correlation attack on the two-level Bluetooth E0 is developed and optimized. Whereas the analysis in [24] was based on a traditional distinguishing attack using the strongest (unconditional) 5-bit correlation, we have successfully demonstrated the superiority of our attack over [24] by showing a best attack using 4-bit conditional correlations, which are *not* suitable for the attack in [24] as the corresponding ordinary correlations are all *zeros*. Our best attack fully recovers the original encryption key using the first 24 bits of $2^{23.8}$ frames and with 2^{38} computations. Compared with all existing attacks [13, 14, 16, 20, 24, 29], this is clearly the fastest and only practical known-plaintext attack on Bluetooth encryption so far. It remains to be an interesting challenge to investigate the redundancy in the header of each frame for a practical ciphertext-only attack on Bluetooth encryption.

Acknowledgments

This work is supported in part by the National Competence Center in Research on Mobile Information and Communication Systems (NCCR-MICS), a center of the Swiss National Science Foundation under the grant number 5005-67322. The second author also receives his partial funding through Gebert Rüf Stiftung.

We owe many grateful thanks to Antoine Joux for his inspiring comments. We would also like to thank the anonymous reviewers for many helpful suggestions. The first author would like to thank LASEC staff Gildas Avoine, Thomas Baignères, Martine Corval, Matthieu Finiasz, Jean Monnerat and the former member Pascal Junod for their support during this project. The arduous experiment wouldn't succeed without great support of Thomas Baignères, Matthieu Finiasz and Pascal Junod as well as our robust computer.

References

1. Ross Anderson, *Searching for the Optimum Correlation Attack*, Fast Software Encryption 1994, Lecture Notes in Computer Science, vol.1008, B. Preneel Ed., Springer-Verlag, pp. 137-143, 1994
2. Frederik Armknecht, Matthias Krause, *Algebraic Attacks on Combiners with Memory*, Advances in Cryptology - CRYPTO 2003, Lecture Notes in Computer Science, vol.2729, D. Boneh Ed., Springer-Verlag, pp. 162-175, 2003
3. Frederik Armknecht, Joseph Lano, Bart Preneel, *Extending the Resynchronization Attack*, Selected Areas in Cryptography - SAC 2004, Lecture Notes in Computer Science, vol. 3357, H. Handschuh and A. Hasan Eds., Springer-Verlag, pp. 19-38, 2005 (extended version available at `http://eprint.iacr.org/2004/232`)
4. Frederik Armknecht, Willi Meier, *Fault Attacks on Combiners with Memory*, submitted
5. Thomas Baignères, Pascal Junod, Serge Vaudenay, *How Far Can We Go Beyond Linear Cryptanalysis?*, Advances in Cryptology - ASIACRYPT 2004, Lecture Notes in Computer Science, vol.3329, P. J. Lee Ed., Springer-Verlag, pp. 432-450, 2004
6. Bluetooth$^{\mathrm{TM}}$, *Bluetooth Specification*, version 1.2, pp. 903-948, November, 2003, available at `http://www.bluetooth.org`
7. Anne Canteaut, Michael Trabbia, *Improved Fast Correlation Attacks Using Parity-check Equations of Weight 4 and 5*, Advances in Cryptology - EUROCRYPT 2000, Lecture Notes in Computer Science, vol.1807, B. Preneel Ed., Springer-Verlag, pp. 573-588, 2000
8. Vladimir V. Chepyzhov, Thomas Johansson, Ben Smeets, *A Simple Algorithm for Fast Correlation Attacks on Stream Ciphers*, Fast Software Encryption 2000, Lecture Notes in Computer Science, vol.1978, B. Schneier Ed., Springer-Verlag, pp. 181-195, 2000
9. Philippe Chose, Antoine Joux, Michel Mitton, *Fast Correlation Attacks: An Algorithmic Point of View*, Advances in Cryptology - EUROCRYPT 2002, Lecture Notes in Computer Science, vol.2332, L. R. Knudsen Ed., Springer-Verlag, pp. 209-221, 2002
10. Nicolas T. Courtois, *Fast Algebraic Attacks on Stream Ciphers with Linear Feedback*, Advances in Cryptology - CRYPTO 2003, Lecture Notes in Computer Science, vol.2729, D. Boneh Ed., Springer-Verlag, pp. 176-194, 2003
11. Thomas M. Cover, Joy A. Thomas, *Elements of Information Theory*, Wiley, 1991
12. Patrik Ekdahl, Thomas Johansson, *Some Results on Correlations in the Bluetooth Stream Cipher*, Proceedings of the 10th Joint Conference on Communications and Coding, Austria, 2000
13. Scott Fluhrer, Stefan Lucks, *Analysis of the E0 Encryption System*, Selected Areas in Cryptography - SAC 2001, Lecture Notes in Computer Science, vol. 2259, S. Vaudenay and A. Youssef Eds., Springer-Verlag, pp. 38-48, 2001
14. Scott Fluhrer, *Improved Key Recovery of Level 1 of the Bluetooth Encryption System*, available at `http://eprint.iacr.org/2002/068`
15. Jovan Dj. Golić, *Correlation Properties of a General Binary Combiner with Memory*, Journal of Cryptology, vol. 9, pp. 111-126, Nov. 1996
16. Jovan Dj. Golić, Vittorio Bagini, Guglielmo Morgari, *Linear Cryptanalysis of Bluetooth Stream Cipher*, Advances in Cryptology - EUROCRYPT 2002, Lecture Notes in Computer Science, vol. 2332, L. R. Knudsen Ed., Springer-Verlag, pp. 238-255, 2002
17. Miia Hermelin, Kaisa Nyberg, *Correlation Properties of the Bluetooth Combiner*, Information Security and Cryptology - ICISC'99, Lecture Notes in Computer Science, vol. 1787, JooSeok. Song Ed., Springer-Verlag, pp. 17-29, 2000
18. Thomas Johansson, Frederik Jönsson, *Improved Fast Correlation Attacks on Stream Ciphers via Convolutional Codes*, Advances in Cryptology - CRYPTO'99, Lecture Notes in Computer Science, vol.1666, M. Wiener Ed., Springer-Verlag, pp. 181-197, 1999
19. Thomas Johansson, Frederik Jönsson, *Fast Correlation Attacks through Reconstruction of Linear Polynomials*, Advances in Cryptology - CRYPTO 2000, Lecture Notes in Computer Science, vol.1880, M. Bellare Ed., Springer-Verlag, pp. 300-315, 2000
20. Matthias Krause, *BDD-Based Cryptanalysis of Keystream Generators*, Advances in Cryptology - EUROCRYPT 2002, Lecture Notes in Computer Science, vol. 2332, L. R. Knudsen Ed., Springer-Verlag, pp. 222-237, 2002

21. Sangjin Lee, Seongtaek Chee, Sangjoon Park, Sungmo Park, *Conditional Correlation Attack on Nonlinear Filter Generators*, Advances in Cryptology - ASIACRYPT 1996, Lecture Notes in Computer Science, vol.1163, Kwangjo Kim and Tsutomu Matsumoto Eds., Springer-Verlag, pp. 360-367, 1996
22. Bernhard Löhlein, *Attacks based on Conditional Correlations against the Nonlinear Filter Generator*, available at `http://eprint.iacr.org/2003/020`
23. Yi Lu, Serge Vaudenay, *Faster Correlation Attack on Bluetooth Keystream Generator E0*, Advances in Cryptology - CRYPTO 2004, Lecture Notes in Computer Science, vol.3152, M. Franklin Ed., Springer-Verlag, pp. 407-425, 2004
24. Yi Lu, Serge Vaudenay, *Cryptanalysis of Bluetooth Keystream Generator Two-level E0*, Advances in Cryptology - ASIACRYPT 2004, Lecture Notes in Computer Science, vol.3329, P. J. Lee Ed., Springer-Verlag, pp. 483-499, 2004
25. Mitsuru Matsui, *Linear Cryptanalysis Method for DES Cipher*, Advances in Cryptology - EUROCRYPT'93, Lecture Notes in Computer Science, vol.765, Springer-Verlag, pp. 386-397, 1993
26. Willi Meier, Othmar Staffelbach, *Fast Correlation Attacks on Certain Stream Ciphers*, Journal of Cryptology, vol. 1, pp. 159-176, Nov. 1989
27. Willi Meier, Othmar Staffelbach, *Correlation Properties of Combiners with Memory in Stream Ciphers*, Journal of Cryptology, vol. 5, pp. 67-86, Nov. 1992
28. Alfred J. Menezes, Paul C. van. Oorschot, Scott A. Vanstone, *Handbook of Applied Cryptography*, CRC, 1996
29. Markku Saarinen, *Re: Bluetooth and E0*, Posted at `sci.crypt.research`, 02/09/00
30. Thomas Siegenthaler, *Decrypting a class of Stream Ciphers using Ciphertext only*, IEEE Transactions on Computers, vol. C-34, pp. 81-85, Jan. 1985
31. Serge Vaudenay, *An Experiment on DES - Statistical Cryptanalysis*, Proceedings of the 3rd ACM Conferences on Computer Security, pp. 139-147, 1996

Appendix

Proof for $\mathrm{E}[\Delta(f_{\mathcal{B}})] \geq \Delta(f)$

By Eq.(7), we have

$$\mathrm{E}[\Delta(f_{\mathcal{B}})] = 2^r \sum_{A \in GF(2)^r} \mathrm{E}\left[\left(D_{f_{\mathcal{B}}}(A) - \frac{1}{2^r}\right)^2\right], \tag{22}$$

where the expectation is taken over uniformly distributed $\mathcal{B}$ for the fixed A. On the other hand, since $D_f(A) = \mathrm{E}[D_{f_{\mathcal{B}}}(A)]$ for any fixed A, we have

$$\begin{aligned}\Delta(f) &= 2^r \sum_{A \in GF(2)^r} \left(D_f(A) - \frac{1}{2^r}\right)^2 &(23)\\ &= 2^r \sum_{A \in GF(2)^r} \left(\mathrm{E}\left[D_{f_{\mathcal{B}}}(A)\right] - \frac{1}{2^r}\right)^2 &(24)\\ &= 2^r \sum_{A \in GF(2)^r} \mathrm{E}^2\left[D_{f_{\mathcal{B}}}(A) - \frac{1}{2^r}\right], &(25)\end{aligned}$$

by definition of Eq.(1), with all the expectation taken over uniformly distributed $\mathcal{B}$ for the fixed A. As we know from theory of statistics that for any fixed A,

$$0 \leq \mathrm{Var}\left[D_{f_{\mathcal{B}}}(A) - \frac{1}{2^r}\right] = \mathrm{E}\left[\left(D_{f_{\mathcal{B}}}(A) - \frac{1}{2^r}\right)^2\right] - \mathrm{E}^2\left[D_{f_{\mathcal{B}}}(A) - \frac{1}{2^r}\right] \tag{26}$$

always holds, where equality holds iff $D_{f_{\mathcal{B}}}(A)$ is independent of $\mathcal{B}$. □

Approximation of Distribution of $\mathcal{F}^{\gamma}_{\mathcal{B}^i}(\mathcal{X}^i)$ for Wrong Keys

Firstly, with $\widehat{K_1} \neq K_1$, the reason that we estimate $\mathcal{F}^{\gamma}_{\mathcal{B}^i}(\mathcal{X}^i)$ to be uniformly and independently distributed for all i can be explained as follows for the cases[21] when $\prod_{j=1}^{4} \epsilon(h^{\gamma}_{\mathcal{B}^i_{t_j+1}}) \neq 0$. Assuming that $\mathcal{P}^i$'s are uniformly and independently distributed, we deduct by Eq.(12) that so are $\widehat{\mathcal{B}^i}$'s for every $\widehat{K}$, where $\widehat{\mathcal{B}^i} = (\widehat{\mathcal{B}^i_{t_1+1}}, \ldots, \widehat{\mathcal{B}^i_{t_4+1}})$. Hence, we estimate $g^{\gamma}(\widehat{\mathcal{B}^i_{t_j+1}})$ for $j = 1, \ldots, 4$ are also uniformly and independently distributed, which allows to conclude by Eq.(16) that $D_{\mathcal{F}^{\gamma}_{\mathcal{B}^i}}$ can be approximated by a uniformly distributed sequence.

Secondly, in the remaining one case of wrong guess such that $\widehat{K_1} = K_1$ and $\widehat{K_2} \neq K_2$, $\mathcal{F}^{\gamma}_{\mathcal{B}^i}(\mathcal{X}^i)$ is *no longer uniformly distributed*; but it is more favourable to us, because we have $\mathcal{F}^{\gamma}_{\mathcal{B}^i}(\mathcal{X}^i) = f^{\gamma}_{\mathcal{B}^i}(X^i) \oplus 1$ for all i such that $\prod_{j=1}^{4} \epsilon(h^{\gamma}_{\mathcal{B}^i_{t_j+1}}) \neq 0$, whose distribution has larger Kullback-Leibler distance (see [11]) to $D_{f^{\gamma}_{\mathcal{B}^i}}$ than a uniform distribution does according to [5].

In all, we can pessimistically approximate $D_{\mathcal{F}^{\gamma}_{\mathcal{B}^i}}$ by a uniform distribution for each wrong guess $\widehat{K} \neq K$.

Advanced Application

Having studied how to apply Section 3 with $r = 1$ (namely the uni-bias-based approach) for an attack to E0 in Section 6, we wonder the possibility of improvement based on multi-biases in the same spirit as in [23], which are utilized by the traditional distinguisher.

For the reason of low time complexity of the attack, we still focus on analysis of 4-bit biases; additionally, we restrict ourselves to bi-biases analysis (i.e. $r = 2$) to simplify the presentation, which will be shown later to be optimal. Let $\Gamma = (\gamma_1, \gamma_2)$, where γ_1 is fixed to $(1, 1, 0, 1)$ and γ_2 with length $\ell_2 \overset{\text{def}}{=} |\gamma_2| = 4$ remains to be determined later such that the data complexity is lowered when we analyze the characteristics of bi-biases simultaneously for each frame instead of conducting the previous uni-bias-based analysis.

Recall that $g^{\gamma_1}(\mathcal{B}) : GF(2)^8 \to GF(2)$ in Section 6 was defined to be the most likely bit of $h^{\gamma_1}_{\mathcal{B}}(X)$ for a uniformly distributed $X \in GF(2)^4$ if it exists (i.e. $\epsilon(h^{\gamma}_{\mathcal{B}}) \neq 0$). We extend $g^{\gamma_1}(\mathcal{B}) : GF(2)^8 \to GF(2)$ to $g^{\Gamma}(\mathcal{B}) : GF(2)^8 \to GF(2)^2$ over all $\mathcal{B} \in GF(2)^8$ such that $\epsilon(h^{\gamma_1}_{\mathcal{B}}) \neq 0$, and let $g^{\Gamma}(\mathcal{B})$ be the most likely 2-bit binary vector $\beta = (\beta_1, \beta_2)$. Note that we can always easily determine the first bit β_1 because of the assumption $\epsilon(h^{\gamma_1}_{\mathcal{B}}) \neq 0$; with regards to determining the second bit β_2 in case that a tie occurs, we just let β_2 be a uniformly distributed bit. Let

$$h^{\Gamma}_{\mathcal{B}}(X) = (h^{\gamma_1}_{\mathcal{B}}(X), h^{\gamma_2}_{\mathcal{B}}(X)), \tag{27}$$

$$h^{\bar{\Gamma}}(\mathcal{B}, X) = (h^{\bar{\gamma}_1}(\mathcal{B}, X), h^{\bar{\gamma}_2}(\mathcal{B}, X)). \tag{28}$$

Note that $h^{\Gamma}_{\mathcal{B}}(X)$ outputs the two bits which are generated by the same unknown X given $\mathcal{B}$; by contrast, $h^{\bar{\Gamma}}(\mathcal{B}, X)$ outputs the two bits which are generated by the unknown X and $\mathcal{B}$. We can extend $\mathcal{F}^{\gamma_1}_{\mathcal{B}^i}(\mathcal{X}^i)$ in Eq.(16) to $\mathcal{F}^{\Gamma}_{\mathcal{B}^i}(\mathcal{X}^i)$ by letting

$$\begin{aligned} &\mathcal{F}^{\Gamma}_{\mathcal{B}^i}(\mathcal{X}^i) \\ &= \left(\bigoplus_{j=1}^{4} h^{\gamma_1}_{\mathcal{B}^i_{t_j+1}}(X^i_{t_j+1}) \oplus h^{\bar{\gamma}_1}(\mathcal{B}^i_{t'+1}, X^i_{t'+1}), \bigoplus_{j=1}^{4} h^{\gamma_2}_{\mathcal{B}^i_{t_j+1}}(X^i_{t_j+1}) \oplus h^{\bar{\gamma}_2}(\mathcal{B}^i_{t'+1}, X^i_{t'+1}) \right) \\ &\quad \oplus g^{\Gamma}(\widehat{\mathcal{B}^i_{t_j+1}}), \end{aligned} \tag{29}$$

[21] By definition of $\mathcal{F}^{\gamma}_{\mathcal{B}^i}$, this is trivial for the cases when $\prod_{j=1}^{4} \epsilon(h^{\gamma}_{\mathcal{B}^i_{t_j+1}}) = 0$.

if $\prod_{j=1}^{4} \epsilon(h^{\gamma_1}_{\widehat{\mathcal{B}^i_{t_j+1}}}) \neq 0$; otherwise, we let it be a uniformly distributed two-bit vector. Similarly, we denote $\mathcal{F}^{\Gamma}_{\mathcal{B}^i}(\mathcal{X}^i)$ corresponding to the correct guess by $f^{\Gamma}_{\mathcal{B}^i}$.

It's easy to verify the assumption holds to apply Section 3 that says $D_{\mathcal{F}^{\Gamma}_{\mathcal{B}^i}}$ can still be approximated by a uniform distribution for each wrong guess on the key $\widehat{K} \neq K$. Moreover, by introducing the extended $f^{'\Gamma}_{\mathcal{B}^i}$ from $f^{'\gamma_1}_{\mathcal{B}^i}$ in Eq.(19) as

$$\begin{aligned} f^{'\Gamma}_{\mathcal{B}^i}(\mathcal{X}^i) &\overset{\text{def}}{=} (f^{'\gamma_1}_{\mathcal{B}^i}(\mathcal{X}^i), f^{'\gamma_2}_{\mathcal{B}^i}(\mathcal{X}^i)) \\ &= \left(\bigoplus_{j=1}^{4} h^{\gamma_1}_{\mathcal{B}^i_{t_j+1}}(X^i_{t_j+1}) \oplus h^{\bar{\gamma}_1}(\mathcal{B}^i_{t'+1}, X^i_{t'+1}), \right. \\ &\quad \left. \bigoplus_{j=1}^{4} h^{\gamma_2}_{\mathcal{B}^i_{t_j+1}}(X^i_{t_j+1}) \oplus h^{\bar{\gamma}_2}(\mathcal{B}^i_{t'+1}, X^i_{t'+1}) \right). \end{aligned} \tag{30}$$

Theorem 5 can be extended to the generalized theorem below

Theorem 6. *For all* $\mathcal{B}^i = (\mathcal{B}^i_{t_1+1}, \mathcal{B}^i_{t_2+1}, \mathcal{B}^i_{t_3+1}, \mathcal{B}^i_{t_4+1}) \in GF(2)^{32}$, *we always have*

$$\Delta(f^{\Gamma}_{\mathcal{B}^i}) = \Delta(f^{'\Gamma}_{\mathcal{B}^i}).$$

Similar computation yields the same formula for data complexity we need as in Eq.(20)

$$\frac{4L \log 2}{n} = \Delta(h^{\bar{\Gamma}}) \cdot \mathrm{E}^4 \left[\Delta \left(h^{\Gamma}_{\mathcal{B}_{t+1}} \right) \right]. \tag{31}$$

Experimental result shows that with $\gamma_1 = (1,1,0,1), \gamma_2 = (1,0,1,1)$, we achieve optimum $\Delta(h^{\Gamma}_{\mathcal{B}_{t+1}}) \approx 2^{-2.415}$ (in comparison to $\Delta(h^{\gamma_1}_{\mathcal{B}_{t+1}}) = 2^{-3}$ previously), though $\Delta(h^{\bar{\Gamma}})$ always equals $\Delta(h^{\bar{\gamma}_1})$ regardless of the choice of γ_2; additionally, $\Delta(h^{\bar{\Gamma}}) \equiv 0$ if $\gamma_1, \gamma_2 \neq (1,1,0,1)$. Therefore, we have the minimum data complexity $n \approx 2^{23.8}$ frames. And the time complexity remains the same according to Theorem 1 in Section 3.

Equivalent Keys

Recall that in Subsection 6.1 we have the 33-bit key $K = (K_1, K_2)$, with $K_1 = (\mathcal{G}_{t_1}(\mathcal{K}), \ldots, \mathcal{G}_{t_4}(\mathcal{K}))$. For simplicity, we let $K_{1,i} = \mathcal{G}_{t_i}(\mathcal{K})$. Define the following 8-bit masks (in hexadecimal):

$$\text{mask}_0 = 0x00, \ \text{mask}_1 = 0xff, \ \text{mask}_2 = 0x0f, \ \text{mask}_3 = 0xf0.$$

Then for any K, we can replace $K_{1,i}$ by $K_{1,i} \oplus \text{mask}_j$ for any $i = 1, 2, \ldots, 4$ and $j \in \{0,1,2,3\}$ and replace K_2 by $K_2 \oplus \lceil \frac{j}{2} \rceil$. Denote this set containing $4^4 = 2^8$ elements by $\langle K \rangle$. We can easily verify that the Walsh coefficients $\widehat{\mathcal{H}''}$ of the element in the set equals by following the definition of convolution between $\mathcal{H}$ and $\mathcal{H}'$:

$$\mathcal{H} \otimes \mathcal{H}'(K) = \sum_{K'} \mathcal{H}(K')\mathcal{H}'(K \oplus K'). \tag{32}$$

Since if $R \in \langle K \rangle$ then $R \oplus K' \in \langle K \oplus K' \rangle$ for all K'. And $\mathcal{H}'$ maps all the elements of the same set to the same value from Section 6, we conclude the set defined above form an equivalent class of the candidate keys. Thus, we have 2^8 equivalent 33-bit keys.