Ciphertext-only attacks and weak long-term keys in T-310

Nicolas T. Courtois and Maria-Bristena Oprisanu

University College London, Gower Street, London, UK

Abstract. T-310 is an important Cold War cipher [22]. It was the principal encryption algorithm used to protect various state communication lines in Eastern Germany in the 1980s. The cipher is quite robust and it outputs extremely few bits from the internal state. In this article we study the choice of the long-term key in T-310. The main result is to show that if a key is faulty, communications can be decrypted in a ciphertext-only scenario. The attack becomes possible when the round function is not bijective. For example we demonstrate that this can happen if we omit to check just one highly technical condition out of many which the long-term keys are expected to satisfy. We provide mathematical proofs that the main historical key classes KT1 and KT2 are secure against such attacks.

Key Words: Cold War, block ciphers, T-310, Unbalanced Feistel ciphers, differential cryptanalysis, correlation attacks, weak key attacks, ciphertext-only attacks.

Publication Information: This article is our private postprint published on a personal web-site and is essentially the same as the one officially published. Our copyright agreement authorizes this. This article has been produced in 2019 but the content is older, and it is no longer guaranteed to be fully up-to-date. The official "Version of Record" of this manuscript subject to copyright has been published and in available in Cryptologia journal, vol 42, iss. 4, pp. 316-336, May 2018, DOI= 10.1080/01611194.2017.1362065,
`http://www.tandfonline.com/doi/full/10.1080/01611194.2017.1362065`.

Extended Version: This article as such (taken alone) does not have an up-to-date extended version. However the attacks studied in this paper are contained inside another much longer paper which is our "Master Paper" or our monography study of the T-310 block cipher, which contains countless additional details, further cryptanalysis work and even some source code and software tools, and which is available on eprint. [14]. The direct link is `https://ia.cr/2017/440`. This document was also revised in 2019 however it is no longer guaranteed to be fully up-to-date neither.

1 Introduction

T-310 is an important historical cipher which was used in East Germany during the last period of the Cold War. According to [12, 22], in 1989 there were some 3,800 T-310 cipher machines in active service across all sorts of government, party and internal security services.

1.1 Basic Description of T-310

T-310 is a synchronous stream cipher which derives its keystream from the iteration of a relatively complex block cipher. The main component of T-310 is a keyed permutation which also takes an IV which we will call "the T-310 block cipher". The block size in T-310 is 36 bits only, the secret key has 240 bits which can (but doesn't have to) include 10 parity bits. The IV has 61 bits which are generated at random by the sender. The block cipher is not used directly to encrypt, but it is iterated a large number of times. Some $13 \cdot 127 = 1651$ block cipher rounds are performed in order to extract as few as 10 bits called (B_j, r_j) from the cipher's internal state, which will then be used to encrypt just one 5-bit character of the plaintext by a sort of double one-time pad cf. Fig. 1.

The initial key is $s_{1-120,1-2}$ which is 240 bits. The key used in different encryption rounds repeats every 120 steps:

$$s_{m+120,1-2} = s_{m,1-2}.$$

This periodicity is a key vulnerability which we exploit in this article. In contrast the IV bits are expanded in an aperiodic way from an initial set of 61 bits chosen at random by the sender. The expansion is based on the following LFSR which produces a sequence with a very large prime [21] period of $2^{61} - 1$:

$$f_i = f_{i-61} \oplus f_{i-60} \oplus f_{i-59} \oplus f_{i-56}.$$

This peculiar aperiodic expansion makes T-310 stronger than for example GOST where the same permutation is repeated many times, which is a source of numerous self-similarity attacks [4, 8, 9]. However this sequence remains entirely predictable for the attacker and in this article we show that if we are not careful with the choice of the long-term key, T-310 communications can be decrypted.

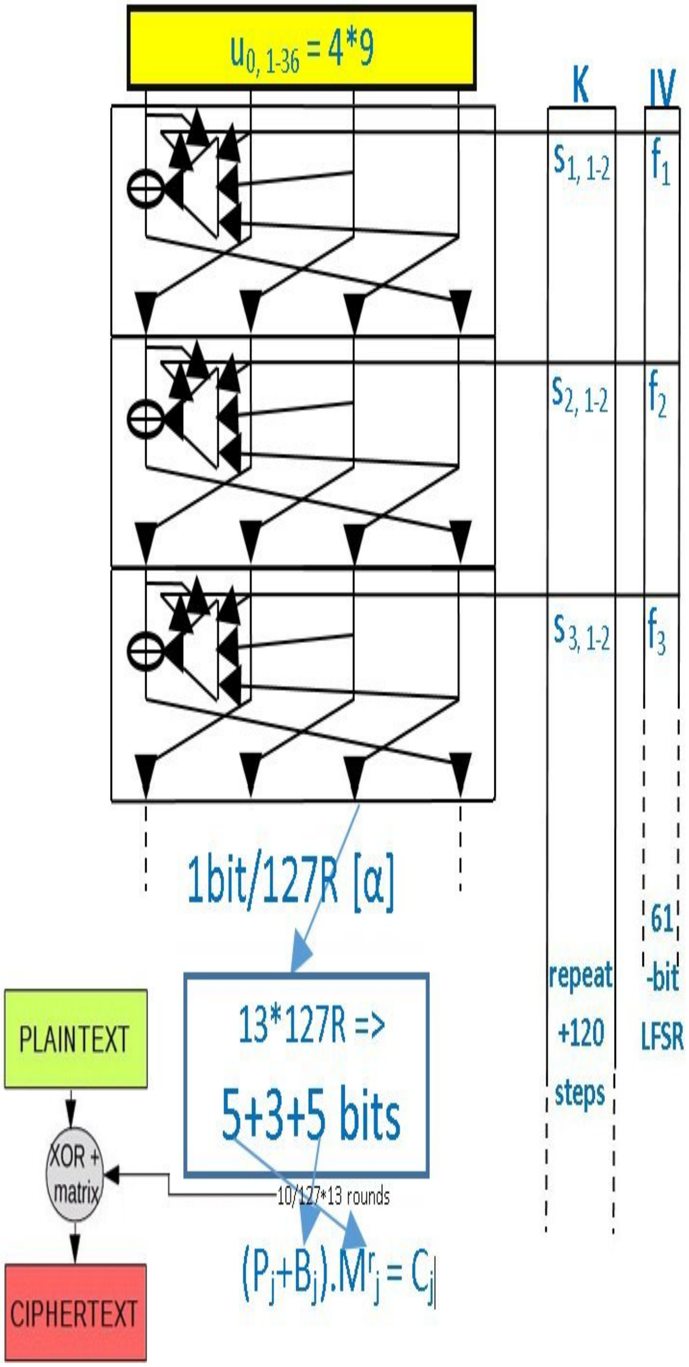

Fig. 1. T-310 Cipher.

T-310 mandates a peculiar variant of a so-called "Contracting Unbalanced Feistel cipher" with 4 branches, cf. [19]. The original Feistel cipher construction had 2 branches and was invented around 1971 [15]. Then East German cipher designers had already in 1970s [22] mandated a substantially more complex structure. The actual connections depend on the so-called long-term key, a.k.a. LZS, in German *Langzeitschlüssel*. In this article we show that T-310 can be strong or weak, depending on the LZS. On Fig. 2 we show what happens for the so-called class KT1 of keys which has been the main and primary method used in T-310 history [13]. Another example is key number 15 from [13] which is not at all like in Fig. 2 and belongs to the so-called class KT2 described in [21].

Following [22] we denote by $u_{m,1-36}$ the 36-bit state of the cipher at moment $m = 0, 1, \ldots$. We denote by $\phi : \{0,1\}^3 \times \{0,1\}^{36} \to \{0,1\}^{36}$ the function of one round. We have

$$(u_{m,1-36}) = \phi\,(s_{m,1}, s_{m,2}, f_m; \quad u_{m-1,1-36})\,.$$

The numbering in the cipher is such that the bits numbered $1, 5, 9, \ldots, 33$ will be those created in one encryption round, and the bits numbered $4, 8, \ldots, 36$ are those which are replaced, and all the other bits get shifted by one position i.e. $u_{m+1,i+1} = u_{m,i}$ for any $i \neq 4k$.

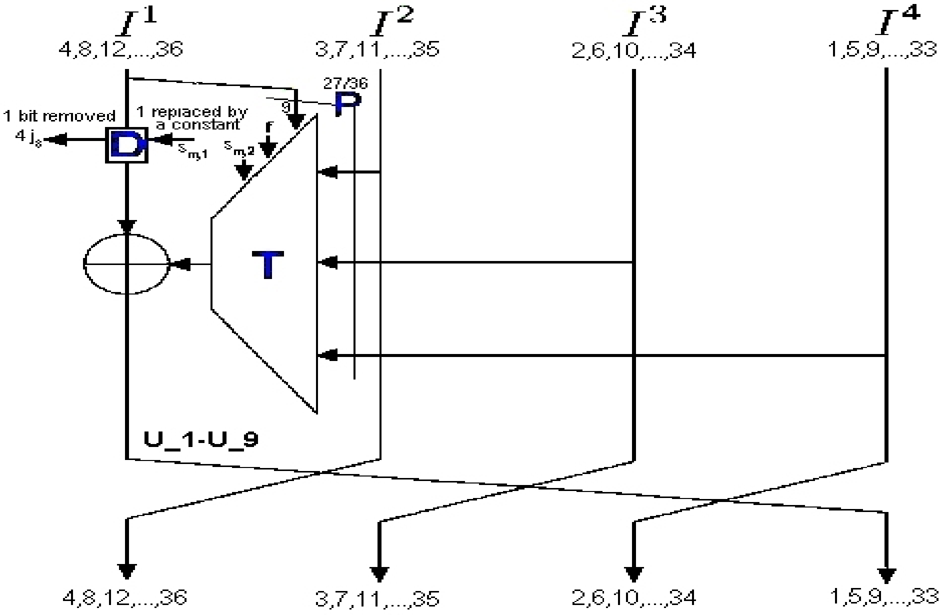

Fig. 2. The internal structure of one encryption round for T-310 in the KT1 case.

1.2 One Block Cipher Round ϕ

Let U_{1-9} be the 9 newly created bits. By definition after one round we have

$$(u_{m+1,1}, u_{m+1,5}, u_{m+1,9}, \ldots, u_{m+1,29}, u_{m+1,33}) = (U_1, U_2, U_3, \ldots, U_8, U_9)$$

It remains to specify how the U_{1-9} are computed inside one round. In [22] this is defined using a function $T : \{0,1\}^{2+27} \to \mathbb{F}_2^9$ which is also illustrated in Fig. 3 below. In this article we use a different particularly compact way to describe this computation: bits can be computed in order starting from U_9 as follows:

$$\begin{aligned}
u_0 &\overset{def}{=} s_1 \\
U_9 &= u_{D(9)} \oplus f \\
U_8 &= u_{D(8)} \oplus U_9 \oplus u_{D(9)} \oplus Z_1(s_2, u_{P(1-5)}) \\
U_7 &= u_{D(7)} \oplus U_8 \oplus u_{D(8)} \oplus u_{P(6)} \\
U_6 &= u_{D(6)} \oplus U_7 \oplus u_{D(7)} \oplus Z_2(u_{P(7-12)}) \\
U_5 &= u_{D(5)} \oplus U_6 \oplus u_{D(6)} \oplus u_{P(13)} \\
U_4 &= u_{D(4)} \oplus U_5 \oplus u_{D(5)} \oplus Z_3(u_{P(14-19)}) \oplus s_2 \\
U_3 &= u_{D(3)} \oplus U_4 \oplus u_{D(4)} \oplus u_{P(20)} \\
U_2 &= u_{D(2)} \oplus U_3 \oplus u_{D(3)} \oplus Z_4(u_{P(21-26)}) \\
U_1 &= u_{D(1)} \oplus U_2 \oplus u_{D(2)} \oplus u_{P(27)}
\end{aligned}$$

Note: These compact notations require a special convention such that if $D(i) = 0$ for one[1] of the i, we put input $u_{m,0} \overset{def}{=} s_{m+1,1}, \quad m \geq 0$ which is part of the secret key and a constant for any given round.

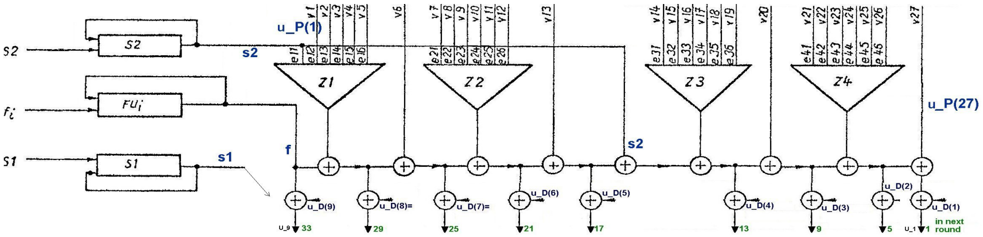

Fig. 3. Internal structure of one round of T-310 based on original drawings [21, 12].

Definition of Z_{1-4}: Finally, in the above, Z_{1-4} are four identical copies of the Boolean function $Z : \mathbb{F}_2^6 \to \mathbb{F}_2$ which is:

$$\begin{aligned}
Z(e_1, e_2, e_3, e_4, e_5, e_6) = e_1 \oplus e_5 \oplus e_6 \oplus e_1e_4 \oplus e_2e_3 \oplus e_2e_5 \oplus e_4e_5 \oplus e_5e_6 \oplus \\
e_1e_3e_4 \oplus e_1e_3e_6 \oplus e_1e_4e_5 \oplus e_2e_3e_6 \oplus e_2e_4e_6 \oplus e_3e_5e_6 \oplus \\
e_1e_2e_3e_4 \oplus e_1e_2e_3e_5 \oplus e_1e_2e_5e_6 \oplus e_2e_3e_4e_6 \oplus e_1e_2e_3e_4e_5 \oplus e_1e_3e_4e_5e_6
\end{aligned}$$

[1] This means that the "Contracting Unbalanced Feistel cipher" structure of [19] is altered. Having (exactly) one i s.t. $D(i) = 0$ is mandatory in [21, 22]. The effect of this is that **one** bit of the left branch is removed, as shown on Fig. 2. Moreover for so called KT1 class of keys [22, 21] and in most of the historical examples of keys in [13] the bit which is removed is $u_{m,4j_8}$ and it is used elsewhere, i.e. we must have $P(i) = 4j_8$ for a certain i.

1.3 How Encryption is Performed - Double One-Time Pad

From our iterated block cipher we extract just 1 bit per 127 rounds: $a_1 \overset{def}{=} u_{127,\alpha}$, then $a_2 \overset{def}{=} u_{254,\alpha}$, where $\alpha \in \{1 \ldots 36\}$ is a constant which is a part of the long-term key and which is called d in [13]. Then we also use $u_{3\cdot 127,\alpha}, \ldots, u_{1651,\alpha}$ and all these will be used to encrypt just one character of the plaintext(!). More generally let $a_i \overset{def}{=} u_{127i,\alpha}$ for any i. Out of these bits, for every 13 bits we discard 3 and use 5+5 bits. Then the encryption is performed as follows:

$$C_j = (P_j \oplus B_j) \cdot M^{r_j},$$

where P_j/C_j is the plaintext/ciphertext character on 5 bits, respectively, then $B_j = (a_{7+13(j-1)}, \ldots, a_{11+13(j-1)})$ are 5 consecutive bits out of the 13 previously discussed and r_j is a "stepping" output which is derived from the FIRST consecutive 5 bits out of the 13 as follows:

$$r_j = \begin{cases} 0 & \text{if} \quad R_j = (0,0,0,0,0) \\ 0 & \text{if} \quad R_j = (1,1,1,1,1) \\ 31 - r & \text{if } R_j \cdot M^r = (1,1,1,1,1) \end{cases}$$

where $R_j \overset{def}{=} (a_{1+13(j-1)}, \ldots, a_{5+13(j-1)})$ and

$$M = \begin{pmatrix} 0\,0\,0\,0\,1 \\ 1\,0\,0\,0\,0 \\ 0\,1\,0\,0\,1 \\ 0\,0\,1\,0\,0 \\ 0\,0\,0\,1\,0 \end{pmatrix}, \quad \text{which is such that} \quad M^{31} = Id.$$

This selection of extremely few bits is where T-310 appears to be a particularly strong cipher, potentially stronger than most block ciphers used in traditional ways such as CBC mode. This is an incredibly low quantity and the cryptanalytic literature knows extremely few examples where the cipher would actually be broken under such difficult circumstances. One major example is the so-called "Dark Side Attack" on MiFare classic [7] one of the most widely used cryptosystems on our planet, with approximately 2 billion RFID smart cards sold. In this attack the attacker obtains only 4 bits from each encryption [7]. Here we can obtain only 1 bit per 127 rounds of encryption. The more rounds, the harder it becomes to develop any sort of cryptographic attack.

2 Analysis of T-310

2.1 The Zero Value Attack on T-310

The Zero-Value attack is a well-known folklore[2] attack in side channel cryptanalysis. The key vulnerability is nicely summarized in the PhD thesis by Matthieu Rivain [20], where we read that "multiplicative masking has a serious drawback: it does not mask the zero value". We have exactly the same problem here with $\cdot M^r$ masking in the T-310. We recall the encryption formula from Section 1.3:

$$C_j = (P_j \oplus B_j) \cdot M^{r_j},$$

Theorem 2.1.1 (Zero-Value Vulnerability in T-310 block cipher). If $C_j = 0^5$ on 5 bits, then $P_j = B_j$ regardless of what the R_j/r_j values are. The converse also holds: if $P_j = B_j$ on 5 bits, then we must have $C_j = 0$.

Notes. This property shows that the "double" one-time pad of T-310 has a security flaw, and shows it could become the equivalent of a "single" one-time pad, if we restrict our attention to a subset of encrypted characters. Unhappily, the designers of T-310 did well to make this sort of attack relatively unattractive: following Section 1.3, the first bit of B_j comes from a_7, which comes from round $7 \cdot 127 = 889$. Breaking T-310 with state bit(s) after 889 rounds seems ambitious.

2.2 Is the Round Function ϕ Bijective?

In theory, from a pure encryption point of view, **nothing** forces ϕ to be invertible. Decryption, in the sense of computing the previous states of our T-310 generalized Feistel cipher variant, is **not** needed in the normal operation of the cipher. However ϕ is bijective in any version of T-310 we have ever heard of. The original documentation clearly says that it must be a bijection cf. pages 47 and 56 in [21]. It appears therefore that if ϕ is required to always be bijective, this will be for security reasons, not for purely functional encryption reasons. The question of whether different LZS will always make ϕ bijective in T-310 and what happens in the contrary case is the central question in this article. Our main result is that more or less every non-bijective LZS can be broken quite badly cf. Table 2. Accordingly a secure setup of T-310 should come with a proof that ϕ is always bijective cf. Section 5 and Section 7.

2.3 Vanishing Differential Attacks

One reason for ϕ to be bijective is that it prevents some **very strong** attacks on block ciphers. Such attacks are very well known for example in mobile telephone SIM cards. These attacks can be called by many different names such as vanishing differentials, all-zero output difference attacks, or collision attacks. For example

[2] It is typically attributed to Golic and Tymen cf. [20, 11] which proposed on solution to the problem, which is not the only one known cf. [11].

in the last 20 years it was relatively easy[3] to extract keys from SIM cards for certain mobile phone operators, and these attacks exploit precisely vanishing differentials cf. [1, 5]. In general, the question of avoiding such rather strong differential properties is precisely the reason why we have many bijections in the design of block ciphers and hash functions. For example S-boxes in SERPENT, PRESENT, GOST [8, 10] and many other ciphers are permutations on 4 bits. DES S-boxes can be viewed as four such permutations, cf. [2]. It is also well-known that DES was designed to make all-zero differentials on 1 or 2 S-boxes impossible, cf. [2, 3].

2.4 Passive and Active Attacks

A vanishing differential attack can definitely be a problem in T-310. However this is **not** the attack we will study in this article. One reason for this is that in the normal operation of T-310 the attacker does NOT have direct access to the cipher's internal state. It is difficult to detect if a collision takes place and it seems only possible in active attack scenarios such as a chosen or related IV attacks, which could also be seen as a form of self-similarity attacks [8]. With two identical IVs in 2 encryptions, if two 36-bit states could be made to be identical, and if the key bits are also aligned appropriately, this would generate two identical keystream sequences later on. This could be detected by the attacker and leak exploitable information about the (very few) keys bits which would appear in the round in which the collision occurs.

In this article we do not study such attacks. Instead we would like to see if there are **stronger** attacks, where the attacker is passive and does not have a possibility to influence the random IVs used in each encryption. Moreover, we assume that the attacker does not have access to the plaintext either. Our goal is therefore to see if and how T-310 with a weak LZS, which leads to a non-bijective ϕ, could be broken in a pure ciphertext-only scenario which is a very ambitious goal[4] which we aim to achieve in this article.

2.5 Weak Keys and the Space Shrinking Property

In this article we show that using a non-bijective ϕ typically has very strong indirect consequences and leads to powerful correlation attacks. The starting point is that each time ϕ is not a permutation, a certain amount of "shrinking" occurs after a few rounds of encryption. The space with initially 2^{36} elements

[3] We and our students have extracted many different keys from SIM cards as recently as in 2012 primarily from Chinese SIM cards, and we have also discovered that certain European mobile operators still used COMP128v1 until 2012. The basic attack was first outlined by Briceno-Goldberg-Wagner cf. [1], and in Section 13.1 slides 249-255 in [5]. Moreover there exist more efficient variants of this attack which we have developed cf. slide 230 in [6]. These attacks do not concern SIM cards which use more recent crypto algorithms.

[4] It is quite difficult to break a cipher in the ciphertext-only setting. For example ciphertext-only attacks on Enigma were shown to be possible only very recently, more than half a century after the WW2 attacks which relied on cribs, cf. [16, 18].

will inevitably shrink, this is if the key and IV bits used in several consecutive rounds are fixed.

In order to show this we have generated a number of non-standard long-term keys for T-310 and also looked at some “anomalous” testing keys specified in [13]. For example we can follow the excessively complex recommendations of [21] for the so-called KT2 keys, which document specifies not less than 40 very technical conditions which these keys need to satisfy, cf. pages 59-60,114-115 and 117 in [21]. Then we look at the very last condition specified on page 60 in [21], which we call M_9. What could possibly go wrong if we omit to verify **just one** of some 40 highly complex conditions? An example of such a key is key 206 below. We call this sort of keys “Rank-Deficient” KT2 keys. As we will show below, this will be sufficient for our cipher to be broken in a ciphertext-only attack scenario. In Table 1 we present the results of computer simulations we have done to see how the output space shrinks after 1, 4, 16 and 24 rounds of encryption ϕ for different keys. In these results we ignore bits $\subseteq$ 1-36 which are never used.

Table 1. Space shrinking simulations for different long-term keys.

key nb	D	P	$M_{\phi}1$	$M_{\phi}4$	$M_{\phi}16$	$M_{\phi}24$
15	0,4,17,12,35,32,2,24,20	15,13,33,34,6,8,5,3,9,18, 14,22,28,30,21,31,7,25,26, 16,27,11,23,29,19,1,36	$2^{36.0}$	$2^{36.0}$	$2^{36.0}$	$2^{36.0}$
206	4,0,32,2,35,17,12,20,24	15,13,33,18,34,8,5,6,9,30, 22,14,16,3,21,31,7,25,26, 28,27,11,23,29,19,1,36	$2^{34.4}$	$2^{33.8}$	$2^{32.9}$	$2^{32.3}$
925	0,16,36,12,32,28,4,8,24	34,24,33,26,14,4,5,28,9, 32,12,18,36,16,21,15,8,25, 35,20,1,6,23,29,19,27,13	$2^{35.0}$	$2^{32.6}$	$2^{30.6}$	$2^{30.5}$
934	0,4,20,12,14,9,19,7,10	21,3,16,25,28,30,26,11,1, 5,6,32,36,29,24,2,23,33, 27,34,8,18,17,31,35,13,22	$2^{32.8}$	$2^{27.7}$	$2^{24.3}$	$2^{23.7}$
27	8,3,5,2,4,6,7,9,1	10,21,18,4,5,8,16,12,6,24, 2,7,3,25,17,26,9,14,22,1, 20,11,19,15,13,23,27	$2^{24.2}$	$2^{18.8}$	$2^{16.1}$	$2^{15.2}$

Keys 15 and 27 are historical keys from [13]. Key 934 was generated at random mandating only that D, P should be bijective and that their outputs should not overlap, and that $D(i) = 0$ for some i, which we consider to be a minimal subset of rules of KT2 which do not have any obvious weakness.

An important observation is that the space shrinking is strong only for a limited number of rounds. Accordingly we will not attempt to guess the state after shrinking, instead we will look at correlations [or conditional biases].

2.6 Correlation Attacks vs. Weak Keys in T-310

The next step is to show that a moderate amount of space shrinking is sufficient in order to obtain an attack on T-310. The main remark is that when the space shrinks from 2^{36} to say M elements, this set of elements is expected to behave as a **random** set of elements of $\mathbb{F}_2^{36}$. And this fact alone, leads to the single bits of type $u_{m,\alpha}$ which are used for encryption in T-310 to become almost inevitably strongly biased, this is when both key and IV bits used are fixed. We illustrate this in Table 2 which shows the **average** bias observed for a few different LZS.

Table 2. Simulations for ϕ^{16} which show the best and average observed bias β on different U_α for 16 rounds of encryption and for a random choice of key/IV bits.

LZS	$M_{\phi^{16}}$	α_{best}	$\lvert P(U_{\alpha_{best}} = 0) - 1/2\rvert$	average($\alpha \in \{1 - 36\}$,keys)
206	$2^{32.9}$	12	$2^{-14.8}$	$2^{-15.5}$
934	$2^{24.3}$	23	$2^{-10.5}$	$2^{-12.9}$
27	$2^{16.1}$	9	$2^{-8.4}$	$2^{-9.4}$
925	$2^{30.6}$	25	$2^{-5.4}$	$2^{-8.4}$

We observe that the bias is quite substantial for **any** value of α and for any weak key studied. Moreover in many cases we observed that it follows a simple law $\mathcal{O}(\sqrt{1/M})$ which is what we would expect for a random function with M possible outputs. This is except for key 925 which is an outlier and a weak key for which the bias substantially worse than $\sqrt{1/M}$.

2.7 Weak Keys and Conditional Correlations

More importantly, it appears that any non-bijective LZS and any α are vulnerable. This is a bit counterintuitive. For example for one round, most of the time we do not expect the output bits to be correlated to any key bits. In Table 3 below we present one example of a weak long-term key when this happens, but in general this will be exceptional. Now in Table Table 2 we see that for ϕ^{16} it looks rather like all non-bijective long-term keys are vulnerable and this is, more or less, for any α with limited variations.

Table 3. Correlations for one particularly weak LZS where the bit S2 leaks to the attacker directly after just 1 round.

LZS nb	sum	α	$Pr[U_\alpha = sum]$
925	s2+f	25	1/2+$2^{-5.6}$

LZS nb	s1	s2	f	α	$Pr[u_\alpha = 0]$
925	0	1	0	25	1/2 -$2^{-5.0}$

The right hand table can be compared to Table 2. The linear approximation in the left table suggests that the output will be biased for any choice of $s1, s2, f$ in this round, however the actual biases are variable and their signs depend on the key/IV bits involved.

3 Useful Natural Language Statistics

Our ciphertext-only attack will rely on some basic facts about the bias on individual bits for German language plaintexts in a realistic teletype setting. For example we look at bit I and observe that the probability of 0 is consistently higher than 0.5. In the table below we report precise results on these probabilities based on simulations with 750 Mbytes of German language corpus downloaded from the online archives of Zeit magazine from 1980-2000, cf. `www.zeit.de`.

Table 4. Statistics for the bias on different bits which occur for German text with 5-bit Baudot-Murray ITA-2 encoding.

$P(\text{bit I} = 0)$	$P(\text{bit II} = 0)$	$P(\text{bit III} = 0)$	$P(\text{bit IV} = 0)$	$P(\text{bit V} = 0)$
$1/2 + 2^{-2.32}$	$1/2 - 2^{-3.67}$	$1/2 + 2^{-4.06}$	$1/2 - 2^{-3.89}$	$1/2 + 2^{-2.27}$

These statistics were computed for plain text with letters and numbers, with spaces and special characters removed, we have converted all letter to lowercase, and we have converted the “umlaut” accented characters as follows: German ü becomes ue, etc, while ß becomes ss. These statistics could be different in a real-life attack setting due to special rules used by T-310 operators or for transmitting files or documents of specific type.

Note. An extended table with 2 different character encodings can be found in Section 18.1 page 56 of [14].

4 A Ciphertext-Only Correlation Attack on T-310

In this section we show how to combine the biases of ϕ^k output in Table 2 and biases on the plaintext due to Table 4 and Thm. 2.1.1 in order to decrypt T-310 communications in the ciphertext-only scenario. Our correlation attack works as follows:

1. We apply the Zero-Value attack of Thm. 2.1.1 and we exploit a proportion of 2^{-5} of the available ciphertext data. We discard all of the other data.
2. We recall from Section 2.1 that if $C_j = 0^5$ we have $P_j = B_j$.
3. We can now express certain, but not all, bits of the plaintext as a function of the internal state bits as
$$P_{j,I-V} = B_{j,0-5}$$
which equation holds for **all** ciphertext characters $C_j = 0$ we selected.
4. We can then approximate the 5 bits of B_j knowing that
$$B_{j,0-5} = (a_{7+13(j-1)}, \ldots, a_{11+13(j-1)})$$
and all these bits are biased using Table 2.
5. We know the expected average value of the bias but we do not know the sign of the bias. The sign of the bias depends on the values of the key and IV bits preceding any of the $(a_{7+13(j-1)}, \ldots, a_{11+13(j-1)})$ which by definition are equal to $u_{127(7+13(j-1)),\alpha}, \ldots, u_{127(11+13(j-1)),\alpha}$. We know the IV bits at any location, we just need to guess key bits at certain locations.
6. In our attack we are going to guess a window of say 48 keys bits for a window of 24 consecutive rounds. The same window of 48 bits is repeated every 120 rounds, (with different IVs which are known to the attacker).
7. We will work on individual bits, and if we want to be able to know the sign of a bias reported in Table 2, we need to know the 32 key bits for 16 rounds preceding the actual bit extracted which are $u_{m,\alpha}$ with $m = 127(B + 13(j-1))$ with five possible $B \in \{7 - 11\}$.
8. We assume that the attacker disposes of a pre-computed table which indicates the sign $\sigma_{K,IV} = +1$ or -1 for the bias for any 32 key bits and any 16 bit IV for ϕ_s^{16}. This table requires only 1 Terabyte of storage (2^{48} bits).
9. We have a window of 24 rounds where the key bits are known and it is repeated with a period of 120 rounds. We consider that positions of type $m = 127(B + 13(j-1))$ span the interval $0 - 119$ uniformly at random. We are interested in positions where key bits are known for at least 16 rounds before m, i.e. the window $m - 15, \ldots, m$ must fall within our window of 24 rounds. The probability of this is $(24 - 16)/120 \approx 2^{-3.9}$.
10. Accordingly, the probability that any $B_{j,0-5}$ we want to compute, can be approximated as a biased bit of type say $1/2 - 2^{-5.8}$ with the sign known to the attacker, is equal to $2^{-3.9}$.
11. In order to simplify our attack, we will only work on plaintext bits I and V in Table 4 which both have a bias of approximately $\pm 2^{-2.3}$. We need to pay attention to the signs; let $\sigma_I = +1$ and $\sigma_V = +1$ for these two bits.
12. The attacker will now compute many biased bits which are all more likely to be 0 than 1, and which combine the biases due to the plaintext and due to ϕ^{16}. Then he will count 0s and 1s and if the bias is sufficiently large he will be able to confirm if his choice of 48 was correct.

13. The attacker assumes that $B_{j,0} = (1 + \sigma_{K,IV})/2$ which is true with probability of about $0.5 + \beta$ where β is a positive value from Table 2, for example for LZS-16 we have $\beta = 2^{-8}$. Similarly we have $B_{j,2} = (1 + \sigma_{K,IV})/2$ for a different choice of 32 key bits and 16 IV bits which pertain to this position.
14. We know that $B_{j,0} = P_{j,I}$ and $B_{j,2} = P_{j,V}$ for all ciphertext positions with $C_j = 0^5$ selected. The sequence of bits the attacker produces will be simply all the $(1 + \sigma_I \sigma_{K,IV})/2$ or $(1 + \sigma_V \sigma_{K,IV})/2$ for all the cases considered. We call these bits available to the attacker "the $B - I$ set".
15. We apply Matsui's piling-up lemma [17] and we see that the overall bias for our bits which are $(1 + \sigma_I \sigma_{K,IV})/2$ or $(1 + \sigma_V \sigma_{K,IV})/2$ is going to be equal to $\gamma = 2^{-2.3}\beta$.
16. In order to distinguish these biased distributions and have results which is stronger than 8 standard deviations we need to generate about $8^2\gamma^{-2} \approx 2^{16+4.6}\beta^{-2}$ biased bits in "the $B - I$ set".
17. We need to work with 8 standard deviations exactly: we apply the Gauss Error function cf. [10] which leads to a probability of $2^{-49.5}$ of a false positive which is sufficient to confirm if our 48-bit key is correct.
18. We get 2 bits for our "the $B-I$ set" when we have ciphertext character with $C_j = 0^5$ which happens with probability 2^{-5} AND when simultaneously the window of 32 bits needed is contained within our window of 48 bits which happens with probability $2^{-3.9}$.
19. Therefore we need overall $2^{16+4.6+3.9+5}\beta^{-2}/2 \approx 2^{28.5}\beta^{-2}$ encrypted characters in order to recover 48 bits of the key in time which is approximately $2^{48+28.5-5-3.9}\beta^{-2} \approx 2^{68}\beta^{-2}$. Here $-5 - 3.9$ comes from the fact the we can pre-select ciphertext bytes and m values for the attack independently of the key depending on the window position.
20. Once we have a plausible candidate for 48 key bits, we can re-do the whole attack with a different and preferably overlapping interval of 24 consecutive rounds and 48 key bits. Making these intervals overlap with those where key bits are already known makes that these extra steps will be substantially faster and easier and their cost can be neglected.

Application using key 206: With our "Rank-Deficient" key 206, we have $\beta = 2^{-15}$ and the attacker can recover the full 240-bit encryption key in a time of 2^{98} given about 2^{59} characters of encrypted data in the ciphertext-only scenario.

Application using key 27: With the original key 27 which is an anomalous key not recommended for encryption from [13], we have $\beta = 2^{-8.0}$ typically and the attacker can recover the full 240-bit encryption key in a time of 2^{84} given about 2^{45} characters of encrypted data in the ciphertext-only scenario.

Better Attacks: An improved attack needs to be non-uniform: use more ciphertext bits, and exploit the fact that for many windows we know the key for **more** than 16 rounds. The number of key bits being guessed needs then to be optimized.

5 A Security Proof For KT1 Keys

We provide a description of the key class KT1 following page 58 and Section 2.2 of Annex 1 on pages 114-115 and also Section 4.1 page 117 in [21]. An incomplete (and therefore not quite correct) description which only included the conditions from page 58 of [21] was published in [22].

$(P, D, \alpha) \in KT1 \Leftrightarrow$ all of the following hold:

D and P are injective
$P(3) = 33, P(7) = 5, P(9) = 9, P(15) = 21, P(18) = 25, P(24) = 29$
Let $W = \{5, 9, 21, 25, 29, 33\}$, $\forall_{1 \geq i \geq 9}\ D(i) \notin W$, $\alpha \notin W$
Let $T = (\{0, 1, \ldots, 12\} \backslash W) \cap (\{P(1-24)\} \cup \{D(4-9)\} \cup \{\alpha\})$
Let $U = (\{13, \ldots, 36\} \backslash W) \cap (\{P(26), P(27)\} \cup \{D(1), D(2), D(3)\})$
$|T \backslash \{P(25)\}| + |U \backslash \{P(25)\}| \leq 12$
$D(1) = 0$
There exist $\{j_1, j_2, \ldots, j_7, j_8\}$ a permutation of $\{2, 3, \ldots, 9\}$ which defines $D(i)$ for every $i \in \{2, 3, \ldots, 9\}$ as follows:
$D(j_1) = 4, D(j_2) = 4j_1, D(j_3) = 4j_2, \ldots, D(j_8) = 4j_7$
$P(20) = 4j_8$ (note: this value is not any of the $D(i)$)
$(D(5), D(6)) \in \{8, 12, 16\} \times \{20, 28, 32\} \cup \{24, 28, 32\} \times \{8, 12, 16\}$
$P(6) = D(8)$ and $P(13) = D(7)$
$P(27) \neq 0 \bmod 4$
$\forall_{1 \geq l \geq 9} \exists_{1 \geq i \geq 26} P(i) = 4 \cdot l$
$D(3) \in \{P(1), P(2), P(4), P(5)\}$
$D(4) \notin \{P(14), P(16), P(17), P(19)\}$
$\{P(8), P(10), P(11), P(12)\} \cap \{D(4), D(5), D(6)\} = \emptyset$

We will now show that these KT1 conditions imply that ϕ is a permutation. Following Section 1.2 we re-write the equations for one encryption round which will be numbered here (1-9) in the order of the 9 "fresh" outputs U_{1-9}, and knowing that $D(1) = 0$ for all KT1 keys [22, 13, 21].

$$U_1 \oplus s_1 = U_2 \oplus u_{D(2)} \oplus u_{P(27)} \tag{1}$$
$$U_2 \oplus u_{D(2)} = U_3 \oplus u_{D(3)} \oplus Z_4(u_{P(21-26)}) \tag{2}$$
$$U_3 \oplus u_{D(3)} = U_4 \oplus u_{D(4)} \oplus u_{P(20)} \tag{3}$$
$$U_4 \oplus u_{D(4)} = U_5 \oplus u_{D(5)} \oplus Z_3(u_{P(14-19)}) \oplus s_2 \tag{4}$$
$$U_5 \oplus u_{D(5)} = U_6 \oplus u_{D(6)} \oplus u_{P(13)} \tag{5}$$
$$U_6 \oplus u_{D(6)} = U_7 \oplus u_{D(7)} \oplus Z_2(u_{P(7-12)}) \tag{6}$$
$$U_7 \oplus u_{D(7)} = U_8 \oplus u_{D(8)} \oplus u_{P(6)} \tag{7}$$
$$U_8 \oplus u_{D(8)} = U_9 \oplus u_{D(9)} \oplus Z_1(s_2, u_{P(1-5)}) \tag{8}$$
$$U_9 \oplus u_{D(9)} = f \tag{9}$$

We have the following result:

Theorem 5.0.1 (KT1 Invertibility Theorem). For every key in the class KT1, as defined in Section 5, and for every 3 bits $s1, s2, f$ the round function ϕ is bijective and given the 36 outputs, the internal bits and the 9 input bits of the form $4 \cdot k$ which are the only bits which are modified, can be computed in the order defined by the following sequence (written in a compact notation):

`0 D1 P27 D9 D2 D7 P13 Z2 D6 D5 Z3 D4 Z4 D3 Z1 D8 P20`

Proof: We need to recover 9 bits which are of type u_{4k}. For the class KT1, cf. Section 5, it is easy to see that inside these u_{4k} we have 8 which are of type $u_{D(i)}$ and one which is always $u_{P(20)}$. All the remaining 27 bits are known from the start, cf. Fig. 2 above. Thus we only need to show how to compute $u_{D(1-9)}$ and then $u_{P(20)}$ given the U_{1-9}.

`D1` We use the notation $D1$ in our compact notation to say that we know from the start that $u_{D(1)} = s_1$.

`P27` We have $P(27) \neq 0 \bmod 4$ for the KT1 keys, cf. Sec. 5, therefore we know $u_{P(27)}$.

`D2` The equation (1) can be used to compute $u_{D(2)} = U_1 \oplus s_1 \oplus U_2 \oplus u_{P(27)}$.

`D7` Then we use the fact that $P(6) = D(8)$ in KT1 keys, cf. Sec. 5. Then equation (7) becomes $U_7 \oplus u_{D(7)} = U_8$ and we can compute $u_{D(7)} = U_7 \oplus U_8$.

`P13` We observe that for all KT1 keys $P(13) = D(7)$, cf. Sec. 5.

`D9` From equation (9) we get: $u_{D(9)} = U_9 \oplus f$.

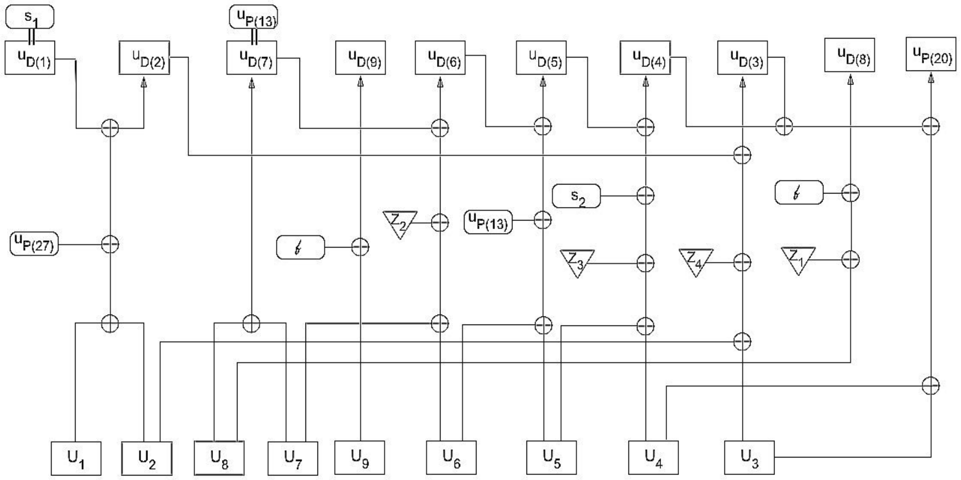

Fig. 4. A method for inverting ϕ which works for ANY key of type KT1.

`Z2` Now we are going to show that we know all the inputs of Z2, which are $u_{P(7-12)}$, which is not quite obvious. At this moment we have already obtained 4 bits of the 10 planned, and there are only SIX bits of type u_{4*k} which remain unknown. These are $u_{D(3-6)}, u_{D(8)}$ and $u_{P(20)}$. Now $D(8) = P(6)$ cf. Sec. 5.
In order to show that $Z_2(u_{P(7-12)})$ can be computed we need to show that: $\{D(3-6), P(6), P(20)\} \cap \{P(7-12)\} = \emptyset$. Moreover knowing that P is injective, we can exclude 6,20 and we just need to show that: $\{D(3-6)\} \cap \{P(7-12)\} = \emptyset$. Moreover, $\{D(3-6)\}$ only contains multiples of 4 and we have $P(7) = 5$ and $P(9) = 9$ due to the W conditions in Sec. 5. It remains to show that:

$$\{D(3-6)\} \cap \{P(8), P(10-12)\} = \emptyset.$$

Now also following Sec. 5, we have $D(3) \in \{P(1), P(2), P(4), P(5)\}$ and P is injective, so we can exclude $D(3)$ and it remains to show that:

$$\{D(4-6)\} \cap \{P(8), P(10-12)\} = \emptyset$$

which is exactly the last KT1 condition in Sec. 5. This ends the proof that Z_2 is known.

D6 Now we compute D6 using equation (6): $u_{D(6)} = U_6 \oplus U_7 \oplus u_{D(7)} \oplus Z_2(u_{P(7-12)})$.

D5 Then after D6 we use equation (5) to compute $u_{D(5)}$ as:

$$u_{D(5)} = U_5 \oplus u_{D(6)} \oplus U_6 \oplus u_{P(13)}$$

Z3 The inputs of Z_3 are $Z_3(u_{P(14-19)})$.
At this moment there are only FOUR bits of type u_{4*k} which remain unknown. These are $u_{D(3-4)}, u_{D(8)}$ and $u_{P(20)}$. Discarding two, $P(20), P(6)$ due to injectivity of P as before, we need to show that:
It remains to show that:

$$\{D(3-4)\} \cap \{P(14-19)\} = \emptyset.$$

We have $P(15) = 21$ and $P(18) = 25$ due to the W conditions. and according to the penultimate condition in Sec. 5, $D(4)$ can be excluded because it says precisely that $D(4) \notin \{P(14), P(16), P(17), P(19)\}$ and $P(15)$ and $P(18)$ were already excluded as not being multiples of 4. It remains to show that:

$$D(3) \notin \{P(14), P(16), P(17), P(19)\},$$

which is ensured by the injectivity of P and pre-penultimate condition in Sec. 5, which says that $D(3) \in \{P(1), P(2), P(4), P(5)\}$.

D4 Now that the the D5 and Z3 steps are done, we use equation (4) to compute $u_{D(4)}$ as:

$$u_{D(4)} = U_4 \oplus U_5 \oplus u_{D(5)} \oplus Z_3(u_{P(14-19)}) \oplus s_2.$$

Z4 The next step is to compute $Z_4(u_{P(21-26)})$. Can this intersect with any of the three remaining unknowns $u_{D(3)}, u_{D(8)}, u_{P(20)}$? The intersection is empty as $D(8) = P(6)$ and $D(3) \in \{P(1), P(2), P(4), P(5)\}$ and P injective makes that none of these can intersect with $P(21-26)$.

D3 From Z_4 and $u_{D(2)}$ we compute $u_D(3)$ using equation (2). We obtain $u_{D(2)} = U_2 \oplus U_3 \oplus u_{D(3)} \oplus Z_4(u_{P(21-26)})$.

Z1 This will enable the computation of $Z_1(s_2, u_{P(1-5)})$. Can this intersect with any of remaining unknowns $u_{D(8)}, u_{P(20)}$? Again no, because $D(8) = P(6)$ and P is injective.

D8 From $Z1$ we can deduce $u_{D(8)}$ using equation (8) and we have: $u_{D(8)} = U_8 \oplus f \oplus Z_1(s_2, u_{P(1-5)})$.

P20 The last unknown is determined using equation (2): $u_{P(20)} = u_{D(3)} \oplus u_{D(4)} \oplus U_3 \oplus U_4$.

This ends the proof that ϕ is bijective for any KT1 type key which is also a security proof against both the "Vanishing Differentials" attacks cf. Section 2.3 and correlation attacks, as described in Section 4.

6 A Key Property of KT2 Keys and a Wider Class KT2b

In this article we define a new class of keys called KT2b which contains a small subset of the excessively large and complex set of conditions for KT2 specified in pages 59-60,114-115 and 117 in [21]. We are not aware of any attack or security problem with any of the KT2b keys and in Thm. 7.0.1 we show that all KT2b keys, and therefore also all KT2 keys, are immune to vanishing differentials and secure against our ciphertext-only correlation attack.

$(P, D, \alpha) \in KT2b \Leftrightarrow$ all of the following hold:

$$\begin{cases} D \text{ and } P \text{ are injective} \\ P(3) = 33, P(7) = 5, P(9) = 9, P(15) = 21, P(18) = 25, P(24) = 29 \\ \text{Let } \ W = \{5, 9, 21, 25, 29, 33\}, \ \ \forall_{1 \geq i \geq 9} \ D(i) \notin W, \ \ \alpha \notin W \\ A = \{D(1-9)\} \cup \{P(6), P(13), P(20), P(27)\} \\ \forall (i, j) \in \{1, \ldots, 27\} \times \{1, \ldots, 9\} : P_i \neq D_j \\ \exists j_1 \in \{1, \ldots, 7\} : D_{j_1} = 0 \\ \{D(8), D(9)\} \subset \{4, 8, \ldots, 36\} \subset A \\ \text{the “Matrix } rank = 9 \text{ condition” } M_9 \text{ defined in Section 6.1.} \end{cases}$$

We will specify the M_9 condition later, as initially it is **not** trivial that such a matrix actually exists in the first place. We need the following result:

Lemma 6.0.1 (KT2b Separation Lemma). For every key which satisfies the conditions in the class KT2b and ignoring the last M_9 condition, the 4 non-linear functions $Z()$ inside the round function ϕ depend only on variables of I^{2-4} which are not modified by ϕ, i.e. the $Z_{1-4}()$ do **not** depend on any of the input variables of type $4k$ in $I^1 \cup \{0\}$.

Proof: For every KT2b key we have:

$$\{4, 8, \ldots, 32, 36\} \subset \{D(1-9); P(6), P(13), P(20), P(27)\}$$

and all outputs of D and P are disjoint by definition in KT2b. This implies that the inputs of 4 non-linear functions $Z()$ cannot contain any of the $\{4, 8, \ldots, 32, 36\}$. Moreover in KT2b one of $D(1-7)$ will be 0 (which is where $u_{D(i)}$ is replaced by $s1$ in the definition of ϕ). Accordingly, $u_0 = s_1$ cannot be any of the inputs of the $Z()$ either, which are all either of the form $u_{P(i)}$ or s_2.

6.1 The Statement of the M_9 Condition

Now we can provide a statement of the “Matrix $rank = 9$ condition” as follows:

$$M_9 : \begin{cases} \text{The concrete values } D(i)/P(j) \text{ inside the 1+9 formulas of Section 1.2 which} \\ \text{define the 9 “fresh” outputs } \{1, 5, \ldots, 33\} \text{ of } \phi \text{ a.k.a. } U_{1-9} \text{ appear at such places} \\ \text{that all the 9 “fresh” outputs } U_{1-9} \text{ of } \phi \text{ are sums of non-linear parts of type } Z(.), \\ \text{plus affine parts which involve various variables } u_i \text{ with } i \neq 4k, \text{ plus an invertible} \\ \text{matrix } B \text{ of rank 9 applied to the remaining 9 inputs } \{4, 8, \ldots, 36\}. \end{cases}$$

6.2 Computation of the Matrix *B*

We recall our compact description of ϕ from Section 1.2:

$$\begin{aligned}
u_0 &\overset{def}{=} s_1\\
U_9 &= u_{D(9)} \oplus f\\
U_8 &= u_{D(8)} \oplus U_9 \oplus u_{D(9)} \oplus Z_1(s_2, u_{P(1-5)})\\
U_7 &= u_{D(7)} \oplus U_8 \oplus u_{D(8)} \oplus u_{P(6)}\\
U_6 &= u_{D(6)} \oplus U_7 \oplus u_{D(7)} \oplus Z_2(u_{P(7-12)})\\
U_5 &= u_{D(5)} \oplus U_6 \oplus u_{D(6)} \oplus u_{P(13)}\\
U_4 &= u_{D(4)} \oplus U_5 \oplus u_{D(5)} \oplus Z_3(u_{P(14-19)}) \oplus s_2\\
U_3 &= u_{D(3)} \oplus U_4 \oplus u_{D(4)} \oplus u_{P(20)}\\
U_2 &= u_{D(2)} \oplus U_3 \oplus u_{D(3)} \oplus Z_4(u_{P(21-26)})\\
U_1 &= u_{D(1)} \oplus U_2 \oplus u_{D(2)} \oplus u_{P(27)}
\end{aligned}$$

We are now ready to write the matrix B for any KT2b or/and any KT2 key, we just need to discard all the $Z()$ and all the numbers not in $\{4, 8, \ldots, 32, 36\}$ in and we will a obtain a square 9×9 matrix $B = (b_{ij})$. We then have:

$$\begin{pmatrix} U_1\\ U_2\\ U_3\\ U_4\\ U_5\\ U_6\\ U_7\\ U_8\\ U_9 \end{pmatrix} = B \cdot \begin{pmatrix} u_4\\ u_8\\ u_{12}\\ u_{16}\\ u_{20}\\ u_{24}\\ u_{28}\\ u_{32}\\ u_{36} \end{pmatrix} + C \quad \text{where } C \overset{def}{=} \begin{pmatrix} f\\ Z_1(s_2, u_{P(1-5)})\\ u_{P(6)} \oplus \ldots\\ Z_2(u_{P(7-12)}) \oplus \ldots\\ u_{P(13)} \oplus \ldots\\ Z_3(u_{P(14-19)}) \oplus s_2 \oplus \ldots\\ u_{P(20)} \oplus \ldots\\ Z_4(u_{P(21-26)}) \oplus \ldots\\ u_{P(27)} \oplus \ldots \end{pmatrix}$$

Here $\oplus \ldots$ denotes some additional terms and will not occur in the first two lines; they will only occur if some of the $u_{D()}$ in our equations of Section 1.2 reproduced above have terms which are not in $\{4, 8, \ldots, 36\}$, in which case they need to be added to C, with a replacement of u_0 by s_1 in one case.

7 A Security Proof For KT2 Keys

We have the following result:

Theorem 7.0.1 (KT2 and KT2b Invertibility Theorem). For every key in the class KT2b, and therefore also for every KT2 key, and for every 3 bits $s1, s2, f$ the round function ϕ is bijective, and given the 36 outputs, the 9 input bits of the form $4k$, can be computed by solving a linear system of rank 9.

Proof: Again due to KT2 Separation Lemma 6.0.1, we know all of the values in C and the matrix B is assumed to be invertible. Therefore we can do the inversion simply as:

$$\begin{pmatrix} u_4 \\ u_8 \\ u_{12} \\ u_{16} \\ u_{20} \\ u_{24} \\ u_{28} \\ u_{32} \\ u_{36} \end{pmatrix} = B^{-1} \cdot \begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \\ U_6 \\ U_7 \\ U_8 \\ U_9 \end{pmatrix} + B^{-1} \cdot C, \qquad \text{where } C \overset{def}{=} \begin{pmatrix} f \\ Z_1(s_2, u_{P(1-5)}) \\ u_{P(6)} \oplus \ldots \\ Z_2(u_{P(7-12)}) \oplus \ldots \\ u_{P(13)} \oplus \ldots \\ Z_3(u_{P(14-19)}) \oplus s_2 \oplus \ldots \\ u_{P(20)} \oplus \ldots \\ Z_4(u_{P(21-26)}) \oplus \ldots \\ u_{P(27)} \oplus \ldots \end{pmatrix}.$$

Remark: K2 vs. KT1: In KT1 we had a very different situation, many inputs to $Z()$ were not initially known. For KT2 keys the proof is substantially simpler overall and uses extremely few of the conditions mandated for KT2 cf. [21].

8 Conclusion

T-310 is an important Cold War cipher. It is essentially a block cipher from which we extract extremely few bits for the actual encryption. This property makes that T-310 is substantially stronger than other ciphers from the same historical period such as DES. The cryptanalytic literature knows extremely few examples where a cipher could actually be broken under such difficult circumstances. In one such example the attacker obtains only 4 bits from each encryption [7]. In T-310 bits from rounds as high as 1397 are used to encrypt just the first character of the plaintext, which character will already depend on all 240 bits of the key. Breaking T-310 in a completely general setting is very difficult[5]. The T-310 encryption process has however a major flaw: the attacker can have access to individual bits of the internal state due to a zero-attack cf. Section 2.1.

Our main result is to show how to recover the 240-bit key of T-310 in a ciphertext-only attack, more or less each time the long-term key is such that the round function is not bijective. It is extremely rare to see a ciphertext-only attack on a real-life government cipher. This for example was not the case for Enigma during WW2, and the first ciphertext-only attack on Enigma was found only in 1995, cf. [16, 18]. A strong attack requires very serious steps to be taken in order to avoid it. In this article we provide detailed mathematical proofs that the historical recommendations for the KT1 and KT2 classes of key from 1970s are provably secure against this type of attacks.

A crucial question in crypto history is, do the rules mandated by the designers matter, and do they make the cipher secure? Our proofs provide valuable insights into the excessively complex set of requirement for the long-term keys mandated in 1970s for T-310 [21]. For KT1 keys we made use of most the rules. For KT2 keys we see very clearly that great majority of the rules are **not required** for our security result to hold. One surprising result in this article is that it is sufficient to omit **just one** of some 40 conditions from the original KT2 class recommendations in [21], to see the T-310 encryption become insecure (cf. our attack for key 206). A weak LZS could also be a deliberate choice in a chosen-LZS attack scenario. Overall in the case of key 206 we present an attack with a time complexity of 2^{98} and a data complexity of about 2^{59} which allows recovery of a 240-bit key in the ciphertext-only setting. The complexity is worse for the historical key 27 from [13]: we obtain a time complexity of 2^{84} and a data complexity of 2^{45} again to recover a 240-bit key in the ciphertext-only setting.

We believe that a secure symmetric encryption standard should have a robust design. Robust could mean for example, that the security should not collapse from 240 bits to less than 100 and in the ciphertext-only setting, if we omitted to check just one highly technical condition inside an excessively complex specification of KT2 keys.

[5] According to [12], a security evaluation of T-310 was done by the BSI after the German re-unification in 1990, and reportedly its conclusion was that T-310 is "extremely secure" [12].

References

1. Mark Briceno, Ian Goldberg, and David Wagner. *GSM Cloning* public web page. http://www.isaac.cs.berkeley.edu/isaac/gsm-faq.html, 1998. Retrieved on 26 July 2013.
2. Nicolas Courtois, Guilhem Castagnos and Louis Goubin: *What do DES S-boxes Say to Each Other ?* Available on https://ia.cr/2003/184/.
3. Nicolas Courtois: *The Best Differential Characteristics and Subtleties of the Biham-Shamir Attacks on DES,* On https://ia.cr/2005/202.
4. Nicolas Courtois: *Security Evaluation of GOST 28147-89 In View Of International Standardisation,* in Cryptologia, volume 36, issue 1, pp. 2-13, 2012.
5. Nicolas Courtois: *Cryptanalysis of GOST,* a very long extended set of slides about the cryptanalysis of GOST, 2010-2014, http://www.nicolascourtois.com/papers/GOST.pdf. An earlier and shorter version was presented at 29C3, Dec 2012, Hamburg, Germany.
6. Nicolas Courtois: *La Carte à Puce,* 293 slides in English, overview of smart card technology, part of COMPGA12 course taught at University College London in 2007-2017, http://www.nicolascourtois.com/papers/smartc.pdf
7. Nicolas T. Courtois: *The Dark Side of Security by Obscurity and Cloning MiFare Classic Rail and Building Passes Anywhere, Anytime,* In SECRYPT 2009 International Conference on Security and Cryptography: pp. 331-338. INSTICC Press 2009, ISBN 978-989-674-005-4.
8. Nicolas Courtois: *Algebraic Complexity Reduction and Cryptanalysis of GOST,* Monograph study on GOST cipher, 2010-2014, 224 pages, available at https://ia.cr/2011/626.
9. Nicolas Courtois: *On Multiple Symmetric Fixed Points in GOST,* in Cryptologia, Iss. 4, vol 39, 2015, pp. 322-334.
10. Nicolas Courtois: *An Improved Differential Attack on Full GOST,* In Cryptology ePrint Archive, Report 2012/138. 15 March 2012, updated December 2015, https://ia.cr/2012/138.
11. Nicolas T. Courtois and Louis Goubin: *An Algebraic Masking Method to Protect AES Against Power Attacks,* https://ia.cr/2005/204.pdf
12. Jörg Drobick: *T-310/50 ARGON,* a web page about T-310 cipher machines consulted 19 March 2017, http://scz.bplaced.net/t310.html
13. Jörg Drobick: *T-310 Schlüsselunterlagen,* a web page which enumerates several different known long-term keys for T-310 from 1973-1990, consulted 21 January 2017, http://scz.bplaced.net/t310-schluessel.html
14. Nicolas T. Courtois, Klaus Schmeh, Jörg Drobick, Jacques Patarin, Maria-Bristena Oprisanu, Matteo Scarlata, Om Bhallamudi: *Cryptographic Security Analysis of T-310,* Monography study on the T-310 block cipher, 132 pages, received 20 May 2017, last revised 29 June 2018, https://ia.cr/2017/440.pdf
15. H. Feistel, W.A. Notz, J.L. Smith, *Cryptographic Techniques for Machine to Machine Data Communications,* Dec. 27, 1971, Report RC-3663, IBM T.J.Watson Research.
16. J. J. Gillogly, *Ciphertext-only cryptanalysis of Enigma, In Cryptologia 19 (4):321413, 1995.*
17. *Mitsuru Matsui: Linear Cryptanalysis Method for DES Cipher,* Eurocrypt'93, LNCS 765, Springer, pp. 386-397, 1993.
18. Olaf Ostwald, Frode Weierud: *Modern breaking of Enigma ciphertexts,* , In Cryptologia, vol. 41, iss. 5, pp. 395-421, 2017.

19. Jacques Patarin, Valérie Nachef, Côme Berbain: *Generic Attacks on Unbalanced Feistel Schemes with Contracting Functions,* in Asiacrypt 2006, pp. 396-411, LNCS 4284, Springer 2006.
20. Matthieu Rivain: *On the Physical Security of Cryptographic Implementations,* PhD thesis, 22 September 2009, University of Luxembourg.
21. Referat 11: *Kryptologische Analyse des Chiffriergerätes T-310/50. Central Cipher Organ, Ministry of State Security of the GDR, document referenced as 'ZCO 402/80', a.k.a. MfS-Abt-XI-594, 123 pages, Berlin, 1980.*
22. Klaus Schmeh: The East German Encryption Machine T-310 and the Algorithm It Used, In Cryptologia, vol. 30, iss. 3, pp. 251 257, 2006.