symmetry

MDPI

Article

Algebraic Properties of the Block Cipher DESL

Kenneth Matheis [1,†], Rainer Steinwandt [2,†] and Adriana Suárez Corona [3,*,†]

1 Institute for Mathematics and Computer Science, Boca Raton, FL 33428, USA; kmatheis@imacs.org
2 Department of Mathematical Sciences, Florida Atlantic University, Boca Raton, FL 33431, USA; rsteinwa@fau.edu
3 Department of Mathematical Sciences, Universidad de León, 24071 León, Spain
* Correspondence: asuac@unileon.es
† These authors contributed equally to this work.

Received: 14 October 2019; Accepted: 12 November 2019; Published: 15 November 2019

Abstract: The Data Encryption Standard Lightweight extension (DESL) is a lightweight block cipher which is very similar to DES, but unlike DES uses only a single S-box. This work demonstrates that this block cipher satisfies comparable algebraic properties to DES—namely, the round functions of DESL generate the alternating group and both ciphers resist *multiple right-hand sides* attacks.

Keywords: lightweight cryptography; permutation group; block cipher

1. Introduction

Lightweight cryptography provides solutions tailored for devices with energy or computational constraints, which are increasingly present with the rapid increase of sensors and IoT devices. These requirements should not be met at the cost of losing security properties. Therefore, lightweight ciphers should ensure they offer similar security guarantees to their counterparts.

One of the protocols designed following these principles is DESL, a lightweight cipher very similar to the Data Encryption Standard (DES) [1], proposed by Leander et al. [2]. The proposed cipher introduces one radical change: all substitution boxes in the DES are replaced with a single new S-box. As detailed by Leander et al., this *DES Lightweight extension* (DESL) has very attractive features in terms of implementability on low-cost platforms. The obvious cryptanalytic question is whether these features might have been paid for with a loss of security. In other words, is the security of DESL comparable to that of the original DES? Leander et al.'s original paper [2] shows that DESL offers resistance against several common attack techniques, including certain types of linear and differential cryptanalyses. Finding structural weaknesses in DESL's design remains a challenge, so despite its short key length, DESL continues to attract interest and keeps getting cited [3–5]. Just a few days before submitting this manuscript, Ji et al. used DESL as a testing ground for proposed improvements of Matsui's algorithm [6]. In this contribution, we compare two algebraic properties of DESL with those of DES.

First we show that the round functions of DESL generate the same permutation group as the round functions of DES, namely the alternating group on 2^{64} points. Our proof strategy is the same as taken by Wernsdorf for DES [7], the core part being to establish 3-transitivity for the group in question. It is not surprising that the replacement of DES's S-boxes in DESL necessitates modifications of Wernsdorf's proof, and one might be tempted to hope that facing only one S-box (instead of several as in DES) simplifies the analysis—this did not seem to be the case for the S-box in question.

In the second part of the paper, we compare the resistance of full and reduced round versions of DES and DESL against an algebraic attack technique known as *multiple right-hand sides* (MRHS) [8]. This type of attack seems particularly interesting for Feistel ciphers like DES and DESL MRHS

equations allow a fairly compact encoding of non-linear equations for the secret key, obtained from a known plaintext–ciphertext pair. The operations for solving such equations are in principle suitable for being accelerated through hardware [9], but establishing run-time estimates for such an attack against genuine ciphers is (perhaps unsurprisingly) challenging. While being devised as a tool for cryptanalysis, Raddum and Zajac recently demonstrated that a cipher representation derived from MRHS equations may yield a faster encryption than a reference implementation of a cipher [10]. In [11], Zajac leveraged MRHS equations as a tool to study the connection between the cost of algebraic attacks and the multiplicative complexity of lightweight ciphers. Here we consider the original cryptanalytic application of MRHS equations. The experimental results we found indicate that DESL offers resistance to this type of algebraic attack that is comparable to DES. As an aside, our results falsify a conjecture by Schoonen [12] (Hypothesis 5.1).

To keep our presentation reasonably self-contained, the next section presents the relevant details on the block cipher in question as well as the main ideas underlying an MRHS-based algebraic attack.

2. Preliminaries

With the exception of two modifications, DESL is identical to the Data Encryption Standard; in particular, plaintexts and ciphertexts are elements of $\{0,1\}^{64}$ and the key can be taken for an element of $\{0,1\}^{56}$. The first difference between DES and DESL is not relevant for the group-theoretic property and the algebraic attack we explore: unlike for DES, there is no initial permutation and no final permutation of the data processed in the cipher. The implications of the second modification is less obvious: DESL replaces all eight S-boxes in DES with a single new S-box.

2.1. Description of DESL

Figure 1 illustrates the basic data flow in DESL, and we refer to the DES specification [1] and Leander et al.'s paper [2] for a detailed specification. For our purposes it is enough to be aware of the following:

- There are 16 rounds, each round i implementing a permutation $\pi_i \in S_{2^{64}}$ which depends on a round key $K_i \in \{0,1\}^{48}$. The latter is derived from the secret key $K \in \{0,1\}^{56}$ through a suitable key schedule.
- Each of the 16 rounds involves a round-key-dependent function $F'_{K_i}(R_i) = P \circ \bigoplus \circ S \circ \bigoplus \circ E$ where
  - $E : \{0,1\}^{32} \longrightarrow \{0,1\}^{48}$ is an injective map specified in [1].
  - $\bigoplus : \{0,1\}^{48} \longrightarrow \{0,1\}^{48}, x \longmapsto x \oplus K_i$ adds (xor) the round key K_i to the input.
  - $S : \{0,1\}^{48} \longrightarrow \{0,1\}^{32}$ splits the input $(a_1, \ldots, a_{48}) \in \{0,1\}^{48}$ into 6-bit blocks and for each $j = 1, \ldots, 8$ substitutes $(a_{6j-5}, \ldots, a_{6j}) \in \{0,1\}^6$ with the corresponding 4-bit value obtained from Table 1.
  - $P \in S_{2^{32}}$ is a permutation on 32-bit strings as specified in [1].
- In each round, the 64-bit input is split into a left half $L_i \in \{0,1\}^{32}$ and a right half $R_i \in \{0,1\}^{32}$. Then the value $L'_i := F'_{K_i}(R_i) \oplus L_i$ is computed, where $\oplus$ is addition in $\{0,1\}^{48}$. The output of round i for $i \in \{1, \ldots, 15\}$ is (R_i, L'_i). In the last round there is no swap, that is, the value (L'_{16}, R_{16}) is output.

Table 1. The substitution function $S : \{0,1\}^6 \longrightarrow \{0,1\}^4$ of DESL is given by this S-box from [2]; $(a_1, \ldots, a_6) \in \{0,1\}^6$ is mapped to the 4-bit binary representation of the table entry in row no. a_1a_6 and column no. $a_2a_3a_4a_5$ (both interpreted as binary representation of a number in $\{0, \ldots, 3\}$ resp. $\{0, \ldots, 15\}$).

14	5	7	2	11	8	1	15	0	10	9	4	6	13	12	3
5	0	8	15	14	3	2	12	11	7	6	9	13	4	1	10
4	9	2	14	8	7	13	0	10	12	15	1	5	11	3	6
9	6	15	5	3	8	4	11	7	1	12	2	0	14	10	13

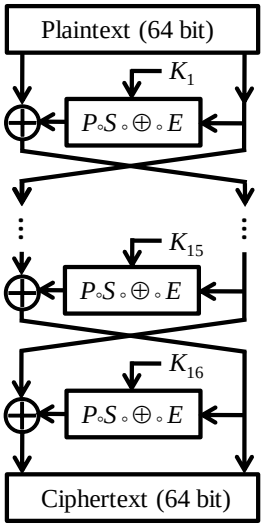

Figure 1. Data Encryption Standard Lightweight extension (DESL) overview.

For the group-theoretic part of our discussion of DESL, we make use of an observation about DES by Davio et al. [13] which has also been exploited in [7]. Namely, we rewrite DESL as shown in Figure 2, that is, by applying P^{-1} respectively P before the first round and after the last round, we combine E and P into a single function EP such that P no longer has to be applied after the application of the S-box. The composition of and E and P is given in Table 2.

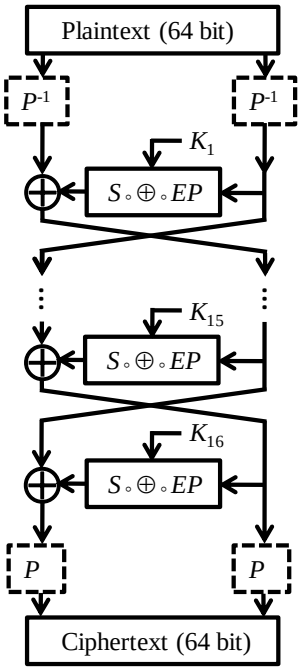

Figure 2. Equivalent description of DESL with the permutation P being applied before the expansion function E.

Table 2. The function $EP : \{0,1\}^{32} \longrightarrow \{0,1\}^{48}$, mapping $(a_1, \ldots, a_{32})$ to $a_{EP(1)}, \ldots, a_{EP(32)}$ where $EP(j)$ is the j-th entry in the table, reading from left to right, top to bottom (e.g., $EP(7) = 21$).

25	16	7	20	21	29
21	29	12	28	17	1
17	1	15	23	26	5
26	5	18	31	10	2
10	2	8	24	14	32
14	32	27	3	9	19
9	19	13	30	6	22
6	22	11	4	25	16

2.2. Multiple Right-Hand Sides (MRHS)

DESL, DES, and many other block ciphers can be modeled as series of polynomial equations over the binary field $\mathbb{F}_2$, therewith suggesting algebraic attacks as a possible attack vector. MRHS offers an alternative to algebraic attacks using SAT solvers or Gröbner bases. Instead of working with ordinary polynomials, equations are represented in a different way, which for several block ciphers, including DESL and DES, can be derived conveniently. For a detailed discussion of MRHS, we refer to Raddum and Semaev's work [8]. Here we restrict ourselves to an informal review of those aspects needed for our application. In particular, we do not discuss specifics of the implementation of the algorithm and refer to [8] (Section 6) for more details (cf. also [12,14]).

2.2.1. Basic Terminology

For a column vector $x = (x_1\ x_2\ \ldots\ x_y)^{\mathrm{T}} \in \mathbb{F}_2^y$, a $k \times y$ binary matrix A of rank k, and column vectors $b_1, b_2, \ldots, b_s \in \mathbb{F}^k$ consider the following type of equation:

$$Ax = b_1, b_2, \ldots, b_s. \tag{1}$$

We refer to such an equation as an *MRHS system of linear equations* with *right hand sides* $b_1, b_2, \ldots, b_s$. By a *solution* to (1) we mean a vector in $\mathbb{F}_2^y$ satisfying at least one particular linear system of equations $Ax = b_i$. The set of *all solutions to* (1) is obtained by forming the union of the solutions to the individual systems $Ax = b_i$ ($1 \leq i \leq s$). To work with MRHS systems of linear equations, we juxtapose the above column vectors b_i to form a matrix L and rewrite Equation (1) as $Ax = [L]$. The pair (A, L) is called a *symbol*, and when writing equations, the brackets around L emphasize that we are not working with an ordinary equation of matrices.

For example, the following is an MRHS system of linear equations:

$$\begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x1 \\ x2 \\ x3 \\ x4 \\ x5 \end{pmatrix} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

and algebraically, it corresponds to the nonlinear equation

$$x_1x_4 + x_1x_2 + x_2x_4 + x_2 + x_3 + x_4 + 1 = 0.$$

Given a system of symbols

$$\begin{array}{rrcl} S_1: & A_1x & = & [L_1] \\ & & \vdots & \\ S_n: & A_nx & = & [L_n] \end{array}, \tag{2}$$

a solution to such a system is defined in the obvious way: it is a vector $x \in \mathbb{F}_2^y$ satisfying all of the underlying n MRHS systems of linear equations, and the goal of the procedure discussed next is to identify all solutions of (2).

2.2.2. Solving a System of Symbols

There are three main components to MRHS: *agreeing*, *gluing*, and *extracting equations*. Since memory is finite in any actual implementation of the algorithm, it may also happen that we have to guess variables, and sometimes an equation symbol is made use of. Each of these parts is discussed below, and we start with a description of the main components.

Agreeing

The basic idea of an agreeing phase is to remove columns b in a right hand side L_i if no solution of $A_i x = b$ can be a solution to the system (2). To achieve this, pairwise *agreeing* of symbols is employed. Namely, let $S_i : A_i x = [L_i]$ and $S_j : A_j x = [L_j]$ be two symbols; we say that S_i and S_j agree if for every $b \in L_i$, there exists a $b' \in L_j$ such that the linear system

$$\begin{pmatrix} A_i \\ A_j \end{pmatrix} x = \begin{pmatrix} b \\ b' \end{pmatrix} \tag{3}$$

is consistent, and, vice versa, for each $b' \in L_j$ there exists a $b \in L_i$ such that (3) is consistent.

In a situation where S_i and S_j do not agree, we remove those columns b from L_i for which the linear system $A_i x = b$ is inconsistent with $A_j x = [L_j]$. Dually, those columns b' from L_j are removed, for which $A_j x = b'$ is inconsistent with $A_i x = [L_i]$. Different strategies can be used to realize this basic idea, but for our purposes it is not necessary to go into further detail on this.

However, it is important to note that if two symbols S_h and S_i agree but S_i and S_j disagree, columns may be deleted in one or both of L_i and L_j. After this happens, it may well happen that S_h does not agree with either of the modified symbols, and it becomes necessary to *re-agree* S_h with them. During the latter agreement, columns from L_h may have to be deleted, and so on, possibly resulting in a chain reaction of column deletions. To ensure that a system of symbols reaches a pairwise-agreed state, we perform the *Agreeing1 algorithm* in Figure 3 (see [8] (Section 3.1)).

While the symbols in a System (2) do not pairwise agree,

1. Find S_i and S_j which do not agree.
2. Agree S_i and S_j.

Figure 3. Agreeing1 algorithm.

Gluing

When a system of symbols is in a pairwise-agreed state, we may choose to apply a different operation: The *gluing* of two symbols $S_i = (A_i, L_i)$ and $S_j = (A_j, L_j)$ results in a new symbol $Bx = [L]$ whose set of solutions is the set of common solutions to $A_i x = [L_i]$ and $A_j x = [L_j]$. After having formed this new symbol, it is inserted into the system at hand and the two symbols S_i and S_j which formed (B, L) are no longer necessary and are removed from the system.

Gluing a matrix L_i of width s_i with a matrix L_j of width s_j may yield a matrix L with as many as $s_i \cdot s_j$ columns. In an implementation, computing certain glues might therefore turn out to be infeasible, and one restricts to gluing only pairs of symbols where the number of columns in the resulting symbol does not exceed a certain threshold.

Once several glues have been performed, the symbols in the resulting system will usually no longer be pairwise-agreed, so the algorithm in Figure 3 can be run again, initiating another round of agreeing and gluing. The eventual goal of iterated agreeing and gluing steps is to obtain a system of symbols which consists of a single symbol.

Extracting Equations

From a given symbol $S : Ax = [L]$ we can try to extract *unique right-hand side (URHS)* equations, and if this is done, the resulting linear equations are placed in a dedicated symbol S_0 to which we refer as an *equation symbol*. The equation symbol is checked for consistency and size. The A-part of S_0 has the same number of columns as the A-parts of the other symbols, but its L-part has only one column. The equation symbol is not considered a proper part of the system (2) and does not take part in the Agreeing1 algorithm, nor is it removed after being glued to a symbol in the system. However, various implementations will involve S_0 in an agreement or gluing step. Furthermore, information from guessing variables may also be reflected by S_0.

Guessing Variables

It may happen that all symbols in a system are pairwise-agreed, no new URHS equations can be extracted, and no pair of symbols can be glued without exceeding the threshold. Lacking a better alternative, in such a situation one can guess the (one-bit) value of a variable. Before performing a guess, the system of symbols—to which we will refer as the *state*—is stored. After the guess has been made, pairwise agreeing, gluing, and equation extraction are performed as normal. If after some steps the state, again, does not allow for any new URHS equation to be computed or pair of symbols to be glued, the state is saved again, and we guess the value of another variable.

Obviously a guess for a variable can be incorrect, and this discovery manifests as follows: during the agreement of two symbols, all right-hand sides of at least one of the symbols get removed, indicating that the system has no solution. When this happens, the state can be rolled back to a previously saved state, so that a different guess can be made.

3. The Group Generated by DESL's Round Functions

In this section we show that the round functions of DESL generate the same group as the round functions of DES. The main part of the argument is to establish 3-transitivity of the group generated by DESL's round functions. To present the (somewhat technical) proof it will be convenient to introduce some notation.

3.1. Notation

The inputs for the S-box of DESL are bitstrings of length 6, outputting bit strings of length 4, as detailed in Table 1. The bitstring inputs are obtained by dividing a 48 bit string into eight blocks of equal length. To refer to the latter, given $a \in \{0,1\}^{48}$, we set $[a]_j := (a_i)_{i=6j-5}^{6j} \quad (j = 1,\ldots,8)$. Analogously, for $a \in \{0,1\}^{32}$, we write $[a]_j := (a_i)_{i=4j-3}^{4j}$ $(j = 1,\ldots,8)$ for the selection of 4-bit blocks. It will be clear from the context when we are dealing with 48-bit, respectively 32-bit values. Finally, as manifested in the balanced Feistel structure, splitting a bitstring of even length into two halves is a common operation in DESL, and for $(a_1,\ldots,a_{2m}) \in \{0,1\}^{2m}$ we define $a_L := (a_i)_{i=1}^{m} \in \{0,1\}^m$ and $a_R := (a_i)_{i=m+1}^{2m} \in \{0,1\}^m$.

Furthermore, for ease of readability, we will often represent bitstrings by the decimal number they represent in binary (again, the length of the bitstring will always be clear from the context). Accordingly, we write $A_{2^{64}}$ and $S_{2^{64}}$ for the alternating and symmetric group respectively on $\{0,1\}^{64}$. Given a set of permutations Π, we denote by $\langle\Pi\rangle$ the group generated by them. Specifically we are interested in the group G generated by the round functions F_K of DESL, where K ranges over all

possible values in $\{0,1\}^{48}$. As in Wernsdorf's analysis of DES in [7], we ignore any restrictions imposed by the key schedule and allow the round keys to be chosen freely.

Using the description and notation from Section 2.1, for a given round key $K \in \{0,1\}^{48}$ we can represent $F_K \in S_{2^{64}}$ as

$$\begin{array}{rccc} F_K: & \{0,1\}^{32} \times \{0,1\}^{32} & \longrightarrow & \{0,1\}^{32} \times \{0,1\}^{32} \\ & (a,b) & \longmapsto & \left(b, ([a]_i \oplus S([K]_i \oplus [EP(b)]_i))_{i=1}^{8}\right) \end{array}.$$

We can therefore state our result in terms of these functions, proving that

$$G = \left\langle \{F_K \in S_{2^{64}} | K \in \{0,1\}^{48}\} \right\rangle = A_{2^{64}}.$$

3.2. Establishing 3-Transitivity of G

Before proving the main result, we will prove some previous lemmas.

Lemma 1. *The round functions of DESL generate a subgroup of $A_{2^{64}}$ that acts transitively on $\{0,1\}^{64}$.*

Proof. Verifying the transitivity of G is straightforward, and the work of Even and Goldreich [15] ensures that G is contained in the alternating group. □

As an intermediate step, we will show the transitivity of $G_0 := \{g \in G |\ g(0) = 0\}$ on $\{0,1\}^{64} \setminus \{(0,\ldots,0)\}$ and transitivity of $G_{0,d} := \{g \in G | g(0) = 0 \text{ and } g(d) = d\}$ on $\{0,1\}^{64} \setminus \{(0,\ldots,0), d\}$, where $d := (\delta_{31,i})_{i=1}^{64}$ has a single non-zero entry at the 31st position.

Before doing so, let us have a closer look at G_0 and $G_{0,d}$:

In view of the Feistel structure of DESL, it is perhaps not very surprising that we deal with pairs of round functions when exploring the transitivity of G_0 and $G_{0,d}$. We define four sets of key pairs, where the last two depend on the auxiliary value $d' := (0,0,0,1,0,0) \in \{0,1\}^6$:

$$\begin{array}{rcl} M & := & \{(k,k') \in \{0,1\}^6 \times \{0,1\}^6 | S(k) = S(k')\} \\ \mathbb{M} & := & \{(K,K') \in \{0,1\}^{48} \times \{0,1\}^{48} | \forall\, j \in \{1,\ldots,8\} : ([K]_j, [K']_j) \in M\} \\ M_{d'} & := & \{(k,k') \in M | S(k \oplus d') = S(k' \oplus d')\} \\ \mathbb{M}_{d'} & := & \{(K,K') \in \mathbb{M} | ([K]_4, [K']_4) \in M_{d'}.\} \end{array}$$

The elements in G we are mainly interested in are of the form $F^L_{K,K'} := F^{-1}_{K'} F_K$ or $F^R_{K,K'} := F_{K'} F^{-1}_K$ with the key pair (K,K') being chosen from $\mathbb{M}$. For input pairs $(a,b) \in \{0,1\}^{32} \times \{0,1\}^{32}$ we have

$$\begin{array}{rcl} F^L_{K,K'}(a,b) & = & ([a]_1 \oplus S([K]_1 \oplus [EP(b)]_1) \oplus S([K']_1 \oplus [EP(b)]_1), \ldots, \\ & & [a]_8 \oplus S([K]_8 \oplus [EP(b)]_8) \oplus S([K']_8 \oplus [EP(b)]_8), b) \text{ and} \\ F^R_{K,K'}(a,b) & = & (a, [b]_1 \oplus S([K]_1 \oplus [EP(a)]_1) \oplus S([K']_1 \oplus [EP(a)]_1), \ldots, \\ & & [b]_8 \oplus S([K]_8 \oplus [EP(a)]_8) \oplus S([K']_8 \oplus [EP(a)]_8)). \end{array}$$

In other words, when evaluating $F^L_{(K,K')}(a,b)$, the right half of the input does not vary and its left half is XORed with the value $(S([K]_i \oplus [EP(b)]_i) \oplus S([K']_i \oplus [EP(b)]_i))_{i=1}^{8}$ to the left half of the input.

For $F^R_{(K,K')}$ the situation is similar, with the left half of the input being stabilized.

The following proposition helps in understanding the effect of repeatedly applying a map of the form $F^R_{K,K'}$, respectively $F^L_{K,K'}$.

Proposition 1. *The functions $F^L_{K,K'}$ and $F^R_{K,K'}$ defined above satisfy the following:*

(a) $\forall (K,K') \in \mathbb{M} : F^L_{K,K'} \in G_{0,d}$ *and* $F^R_{K,K'} \in G_0$.

(b) $\forall (K,K') \in \mathbb{M}_{d'} : F^L_{K,K'} \in G_{0,d}$ *and* $F^R_{K,K'} \in G_{0,d}$.

(c) Let $n \in \mathbb{N}$. *Then, for all* $(K_1,K_1'),\ldots,(K_n,K_n') \in \mathbb{M}$ *and for all* $(a,b) \in \{0,1\}^{32} \times \{0,1\}^{32}$, *the following hold:*

$F^R_{K_1,K_1'} \circ \cdots \circ F^R_{K_n,K_n'}(a,b) =$

$$\Big(a, \quad [b]_1 \oplus \bigoplus_{i=1}^{n} (S([K_i]_1 \oplus [EP(a)]_1) \oplus S([K_i']_1 \oplus [EP(a)]_1)), \ldots,$$
$$[b]_8 \oplus \bigoplus_{i=1}^{n} (S([K_i]_8 \oplus [EP(a)]_8) \oplus S([K_i']_8 \oplus [EP(a)]_8))\Big)$$

and, analogously,

$F^L_{K_1,K_1'} \circ \cdots \circ F^L_{K_n,K_n'}(a,b) =$

$$\Big([a]_1 \quad \oplus \quad \bigoplus_{i=1}^{n} (S([K_i]_1 \oplus [EP(b)]_1) \oplus S([K_i']_1 \oplus [EP(b)]_1)), \ldots,$$
$$[a]_8 \quad \oplus \quad \bigoplus_{i=1}^{n} (S([K_i]_8 \oplus [EP(b)]_8) \oplus S([K_i']_8 \oplus [EP(b)]_8)), \qquad b\Big).$$

Proof. The proof is immediate from the definition of $F^L_{K,K'}$ and $F^R_{K,K'}$. □

To understand better which values can be obtained in the left and right 32-bit halves of the output through repeated application of a map of the form $F^R_{K,K'}$ (respectively $F^L_{K,K'}$), given some 64-bit input, it is helpful to take a look at some $\mathbb{F}_2$-vector subspaces of $\mathbb{F}_2^4$:

Lemma 2. *For* $y \in \{0,1\}^6 \setminus \{(0,0,0,0,0,0)\}$ *let*

$$U(y) := \langle S(k \oplus y) \oplus S(k' \oplus y) \,|\, (k,k') \in M \rangle \subseteq \mathbb{F}_2^4$$

be the $\mathbb{F}_2$-vector space spanned by $\{S(k \oplus y) \oplus S(k' \oplus y) \,|\, (k,k') \in M\}$.

Similarly, denote by $U_{d'}(y)$ *the $\mathbb{F}_2$-vector space*

$$U_{d'}(y) := \langle S(k \oplus y) \oplus S(k' \oplus y) \,|\, (k,k') \in M_{d'} \rangle.$$

Then, the following statements hold:

(a) $\forall\, y \in \{0,1\}^6 \setminus \{(0,0,0,0,0,0),(0,0,0,0,0,1)\} : U(y) = \{0,1\}^4$.

(b) $U(0,0,0,0,0,1) = \{0,2,4,6,8,10,12,14\}$.

(c) $\forall\, y \in \{2,6,17,18,21,22,41,45,49,53,58,62\} : U_{d'}(y) = \{0,1\}^4$.

(d) $\forall\, y \in \{0,1\}^6 \setminus \{(0,0,0,1,0,0)\} : U_{d'}(y) \neq \{0\}$.

Proof. The proof is by direct computation, e.g., using a programming language like Python [16]. □

Remark 1. *Bringing the notation in Lemma 2 to use, from Proposition 1 we obtain the following statements which for the case* $U([EP(a)]_i) = \{0,1\}^4$ *(respectively* $U([EP(b)]_k) = \{0,1\}^4$*) may be regarded as "hinting at transitivity":*

- *For $i = 1, \ldots, 8$ let $u_i \in U([EP(a)]_i)$ be a bitstring. Then, there exist $(K_1, K_1'), \ldots, (K_n, K_n') \in \mathbb{M}$ such that $F^R_{K_1,K_1'} \circ \cdots \circ F^R_{K_n,K_n'}(a,b) = (a, [b]_1 \oplus u_1, \ldots, [b]_8 \oplus u_8)$ for all $(a,b) \in \{0,1\}^{32} \times \{0,1\}^{32}$.*
- *For $i = 1, \ldots, 8$ let $u_i \in U([EP(b)]_i)$ be a bitstring. Then, there exist $(K_1, K_1'), \ldots, (K_n, K_n') \in \mathbb{M}$ such that $F^L_{K_1,K_1'} \circ \cdots \circ F^L_{K_n,K_n'}(a,b) = ([a]_1 \oplus u_1, \ldots, [a]_8 \oplus u_8, b)$ for all $(a,b) \in \{0,1\}^{32} \times \{0,1\}^{32}$.*
- *For $i \in \{1, \ldots, 8\} \setminus \{4\}$ let $u_i \in U([EP(a)]_i)$ be a bitstring and let $u_4 \in U_{d'}([EP(a)]_4)$. Then, there exist $(K_1, K_1'), \ldots, (K_n, K_n') \in \mathbb{M}_{d'}$ such that $F^R_{K_1,K_1'} \circ \cdots \circ F^R_{K_n,K_n'}(a,b) = (a, b_1 \oplus u_1, \ldots, b_8 \oplus u_8)$ for all $(a,b) \in \{0,1\}^{32} \times \{0,1\}^{32}$.*
- *For $i \in \{1, \ldots, 8\} \setminus \{4\}$ let $u_i \in U([EP(b)]_i)$ be a bitstring and let $u_4 \in U_{d'}([EP(b)]_4)$. Then there exist $(K_1, K_1'), \ldots, (K_n, K_n') \in \mathbb{M}_{d'}$ such that $F^L_{K_1,K_1'} \circ \cdots \circ F^L_{K_n,K_n'}(a,b) = (a_1 \oplus u_1, \ldots, a_8 \oplus u_8, b)$ for all $(a,b) \in \{0,1\}^{32} \times \{0,1\}^{32}$.*

Therefore, if we know that the equality $U([EP(a)]_k) = \{0,1\}^4$ holds for some $1 \leq k \leq 8$, then for each bitstring $c \in \{0,1\}^4$ we can find a sequence of key pairs $(K_1, K_1'), \ldots, (K_n, K_n') \in \mathbb{M}$ with

$$\left[\left[F^R_{K_1,K_1'} \circ \cdots \circ F^R_{K_n,K_n'}(a,b)\right]_R\right]_k = c.$$

For instance, we can choose pairs $(K_1, K_1'), \ldots, (K_n, K_n')$ with $([K_j]_k, [K_j']_k) \in M$ corresponding to the linear combination of $c \oplus [b]_k$, and the rest of the positions being 0. This ensures that all (K_j, K_j') are contained in $\mathbb{M}$, and if $U_{d'}([EP(a)]_k) = \{0,1\}^4$ or $k \neq 4$, we can also ensure $(K_1, K_1'), \ldots, (K_n, K_n') \in \mathbb{M}_{d'}$.

Similarly, in case $U([EP(b)]_k)$ contains all bitstrings of length 4, we can obtain a sequence of key pairs with

$$\left[\left[F^L_{K_1,K_1'} \circ \cdots \circ F^L_{K_n,K_n'}(a,b)\right]_L\right]_k = c.$$

The subsequent lemmata enable us to argue that $G_{0,d}$ acts transitively on $\{0,1\}^{64} \setminus \{0,d\}$. In other words, we prove that for all $x, y \in \{0,1\}^{64} \setminus \{0,d\}$ the equivalence $x \sim y$ holds, where $x \sim y \iff \exists g \in G_{0,d} : g(x) = y$. The proofs exploit in particular the transitivity of $\sim$.

Lemma 3. *Let $e := (1,0,1,\ldots,1) \in \{0,1\}^{32}$ be the 32-bit vector which has a single 0-entry at the second position and 1-entries everywhere else, and let $(z,z') \in \{0,1\}^{32} \times \{0,1\}^{32}$ be arbitrary. Then $(e,z) \sim (e,z')$.*

Proof. Let $(z,z') \in \{0,1\}^{32} \times \{0,1\}^{32}$ be arbitrary, but fixed. From Table 2 we see that

$$[EP(e)]_i = \begin{cases} (1,1,1,1,1,1) & \text{, if } i \in \{1,2,3,6,7,8\} \\ (1,1,1,1,1,0) & \text{, if } i = 4 \\ (0,1,1,1,1,1) & \text{, if } i = 5 \end{cases}$$

Hence, by properties (a) and (c) of Lemma 2 we obtain $U((EP(e))_i) = \{0,1\}^4$ for all $i = 1, \ldots, 8$ as well as $U_{d'}((EP(e))_4) = \{0,1\}^4$.

Therefore, because of Remark 1 for $c = (z_1', z_2', z_3', z_4')$ we get:

$(e,z) \sim (e, (z_1', z_2', z_3', z_4', z_5, \ldots, z_{32}))$, since $(e, (z_1', z_2', z_3', z_4', z_5, \ldots, z_{32})) = F^R_{K^1,K^{1'}} \circ \cdots \circ F^R_{K^n,K^{n'}}(e,z)$, for the corresponding $(K^i, K^{i'})$, $i \in \{1, \ldots, n\}$.

Analogously, since $U((EPe)_2) = \{0,1\}^4$, we can obtain:

$$(e, (z_1', z_2', z_3', z_4', z_5, \ldots, z_{32})) \sim (e, (z_1', \ldots, z_8', z_9, \ldots, z_{32})).$$

If we continue carrying out the same procedure, since all the subspaces considered are $\{0,1\}^4$, we can finally see that $(e,z) \sim (e,z')$. □

Lemma 4. $\forall\ a \in \{0,1\}^{64} \setminus \{0,d\}\ , \exists\ a' \in \{0,1\}^{64} \setminus \{0,d\} : a' \sim a$ *and* $\exists\ i \in \{1,\ldots,32\} \setminus \{2,5,10,18,26,31\} : a'_i = 1$.

Proof. If $\exists\ i \in \{1,\ldots,32\} \setminus \{2,5,10,18,26,31\} : a_i = 1$, then we obtain the lemma with $a' := a$.

Otherwise, we distinguish two cases:

- If $\exists\ i \in \{33,\ldots,64\} : a_i = 1$:

  Then $\exists\ l \in \{1,\ldots,8\}$ such that $[EP(a)^{64}_{i=33}]_l \neq 0$:

  - If $[EP(a)^{64}_{i=33}]_l \neq 1$, then $U([EP(a)^{64}_{i=33})]_l) = \{0,1\}^4$. Therefore, because of Remark 1, we can show $a' = F^L_{K^1,K^{1'}} \circ \cdots \circ F^L_{K^n,K^{n'}}(a)$ such that $([a']_L)_j = 1$ for $j \in \{4l-3,\ldots,4l\}$. Thus, $\exists\ i \in \{1,\ldots,32\} \setminus \{2,5,10,18,26,31\} : a'_i = 1$.
  - If $[EP(a)^{64}_{i=33}]_l = 1$, then $U([EP(a)^{64}_{i=33}]_l) = \{0,2,4,6,8,10,12,14\}$. With an argument similar to the previous one, we can get an element $a' = F^L_{K^1,K^{1'}} \circ \cdots \circ F^L_{K^n,K^{n'}}(a)$, such that $(a'_L)_i = 1$ for $i \in \{4l-3,\ldots,4l-1\}$. Therefore, $\exists\ i \in \{1,\ldots,32\} \setminus \{2,5,10,18,26,31\} : a'_i = 1$.

- If $\forall\ i \in \{33,\ldots,64\} : a_i = 0$.

  Since $a \neq 0$, then $\exists\ i \in \{1,\ldots,32\} : a_i = 1$. Therefore, $\exists\ l \in \{1,\ldots,8\}$ such that $[EP(a)^{32}_{i=1}]_l \neq 0$ and, like before (but using "right-functions") we prove that we can get an element $a' = F^R_{K^1,K^{1'}} \circ \cdots \circ F^R_{K^n,K^{n'}}(a)$, where $(K^i, K^{i'}) \in \mathbb{M}_{d'}$, such that $\exists\ i \in \{33,\ldots,64\} : a'_i = 1$. Notice that in this case the pairs $(K^i, K^{i'})$ must be not only in $\mathbb{M}$, but in $\mathbb{M}_{d'}$, so that $a \sim a'$ (Proposition 1(b)).

  - If $l \neq 4$
    - If $(EP(a)^{32}_{i=1})_l \neq 1$, then $U([EP(a)^{32}_{i=1})]_l) = \{0,1\}^4$.
      Therefore, because of Remark 1, we can have $a' = F^R_{K^1,K^{1'}} \circ \cdots \circ F^R_{K^n,K^{n'}}(a)$, where $(K^i, K^{i'}) \in \mathbb{M}_{d'}$, with $a'_i = 1$ for some $i \in \{33,\ldots,64\}$.
    - If $[EP(a)^{32}_{i=1}]_l = 1$, then $U([EP(a)^{32}_{i=1})]_l) = \{0,2,4,6,8,10,12,14\}$. With the same argument as before, we can get an element $a' = F^R_{K^1,K^{1'}} \circ \cdots \circ F^R_{K^n,K^{n'}}(a)$, such that $a'_i = 1$ for $i = 32 + j$, where $j \in \{4l-3,\ldots,4l-1\}$.
  - If $l = 4$: Since $a \neq d$, according to Table 2, $(EPa)_4 \neq (0,0,0,1,0,0)$. Therefore, we have $U_{d'}((EPa)_4) \neq 0$ (Lemma 2(d)) and we can obtain, as in the previous cases, an element $a' := F^L_{K^1,K^{1'}} \circ \cdots \circ F^R_{K^n,K^{n'}}(a) \sim a$, with $a'_i = 1$ for some $i \in \{33,\ldots,64\}$.

  Hence, this case is traced back to the case $\exists\ i \in \{33,\ldots,64\} : a_i = 1$ and the proof is complete. □

Lemma 5. $\forall\ a' \in \{0,1\}^{64} \setminus \{0,d\} : a' \sim a$ *and* $\exists\ i \in \{1,\ldots,32\} \setminus \{2,5,10,18,26,31\} : a'_i = 1$, $\exists\ a'' \in \{0,1\}^{64} \setminus \{0,d\} : a'' \sim a'$ *and* $\forall\ i \in \{1,\ldots,32\} \setminus \{13,\ldots,16\} : a''_i = e_i$.

Proof. If $\forall\ i \in \{1,\ldots,32\} \setminus \{13,\ldots,16\} : a''_i = e_i$, then we immediately obtain the Lemma with $a'' := a'$.

Otherwise, we choose an index $j \in \{1,\ldots,32\} \setminus \{2,5,10,18,26,31\} : a'_j = 1$ and we will prove that $\exists\ a^0 \in \{0,1\}^{64} \setminus \{0,d\} : a^0 \sim a'$, $[a^0]_L = [a']_L$ *and* $\forall\ i \in I(j) : (a^0)_{32+i} = 1$, where the sets $I(j)$ are defined in Figure 4.

j	$I(j)$
1	$\{5,\ldots,12\}\setminus\{8\}$
3 or 27	{21,…, 24}
4 or 11	{29,…, 32}
6	{25,…, 32}
7 or 20	{1,…, 4}
8 or 24	{17,…, 20}
9	{21,…, 29}
12 or 28	{5,…, 8}
13 or 30	{25,…, 28}
14	{17,…, 24}
15 or 23	{9,…,12}
16	$\{1,\ldots,4\}\cup\{29,\ldots,31\}$
17	{5,…, 12}
19	$\{21,\ldots,28\}\setminus\{24\}$
21	{1,…, 8}
22	$\{25,\ldots,32\}\setminus\{28\}$
25	$\{1,\ldots,4\}\cup\{29,\ldots,32\}$
29	$\{1,\ldots,8\}\setminus\{4\}$
32	$\{17,\ldots,24\}\setminus\{20\}$

Figure 4. Definition of $I(j)$.

We define $a^0 := F^R_{K^1,K^{1'}} \circ F^R_{K^2,K^{2'}} \circ \cdots \circ F^R_{K^n,K^{n'}}(a')$, with $(K^i, K^{i'}) \in \mathbb{M}_{d'}$. Therefore, $[a^0]_L = [a']_L$, and we will see that if $(K^i, K^{i'}), i \in \{1,\ldots,n\}$ have been chosen appropriately, we can have $(a^0)_{32+i} = 1, \forall\, i \in I(j)$.

For $j = 1$:

According to Table 2, $[EP(a')_L]_2 \neq 0$ and $[EP(a')_L]_3 \notin \{0,1\}$, since the corresponding positions for a'_1 are 12 and 14, which are in blocks 2 and 3. Therefore, we have:

- If $[EP(a')_L]_2 \neq 1$, then $U([EP(a')_L]_2) = \{0,1\}^4$. Hence, because of Remark 1, $\exists\ (K^i, K^{i'}) \in \mathbb{M}_{d'}$ such that $[[a^0]_R]_2 = [F^L_{K^1,K^{1'}} \circ F^L_{K^2,K^{2'}} \circ \cdots \circ F^L_{K^n,K^{n'}}(a')]_2 = (1,1,1,1)$. Therefore, $(a^0)_{32+i} = 1$ for all $i \in \{5,\ldots,8\}$.

- If $[EP(a')_L]_2 = 1$, then $U([EP(a')_L]_2) = \{0,2,4,6,8,10,12,14\}$. With a similar argument, $\exists\ (K^i, K^{i'}) \in \mathbb{M}_{d'}$ such that $[[a^0]_R]_2 = [F^L_{K^1,K^{1'}} \circ F^L_{K^2,K^{2'}} \circ \cdots \circ F^L_{K^n,K^{n'}}(a')]_2 = (1,1,1,0)$. Therefore, $(a^0)_{32+i} = 1$ for all $i \in \{5,\ldots,7\}$.

Since $[EP(a')_L]_3 \notin \{0,1\}$, then $U([EP(a')_L]_3) = \{0,1\}^4$ and therefore $\exists (K^i, K^{i'}) \in \mathbb{M}_{d'}$ such that $[[a^0]_R]_3 = [F^L_{K^1,K^{1'}} \circ F^L_{K^2,K^{2'}} \circ \cdots \circ F^L_{K^n,K^{n'}}(a')]_3 = (1,1,1,1)$. Therefore, $(a^0)_{32+i} = 1$ for all $i \in \{9,\ldots,12\}$.

Thus, considering the composition of the functions involved, we obtain a^0 such that $(a^0)_{32+i} = 1\ , \forall\, i \in \{5,\ldots,12\}\setminus\{8\}$.

A similar argument applies to the other values of $j \in \{1,\ldots,32\}\setminus\{13,\ldots,16\}$.

Now, we will see that $\exists\ a^1 \in \{0,1\}^{64}\setminus\{0,d\} : a^1 \sim a^0\ ,\ [a^1]_R = [a^0]_R\ and\ \forall\ i\ \in J(j) : (a^1)_i = e_i$, where the sets $J(j)$ are defined in Figure 5.

We define $a^1 := F^L_{K^1,K^{1'}} \circ \cdots \circ F^L_{K^n,K^{n'}}(a')$, with $(K^i, K^{i'}) \in \mathbb{M}$. Therefore, $[a^0]_R = [a']_R$, and we will see that choosing adequate elements $(K^i, K^{i'})$, we can have $(a^1)_i = e_i, \forall\, i \in J(j)$.

For $j = 1$, $I(1) = \{5,\ldots,12\}\setminus\{8\}$:

According to Table 2, let us see which positions $EP(([a^0]_R)_i)$ are in for the different values of $i \in I(1)$. We can see $EP(([a^0]_R)_5)$ is in position 18 (block 3) and 20 (block 4), $EP(([a^0]_R)_6)$ is in position 41 (block 7) and 43 (block 8), $EP(([a^0]_R)_7)$ is in position 3 (block 1), $EP(([a^0]_R)_9)$ is in position 35 and 37 (blocks 6 and 7), $EP(([a^0]_R)_{10})$ is in position 23 and 25 (block 4 and 5), $EP(([a^0]_R)_{11})$ is in position 45 (block 8), and $EP(([a^0]_R)_{12})$ is in position 9 (block 2).

j	$(\{1,\ldots,32\} \setminus \{13,\ldots,16\}) \setminus J(j)$
1 or 17	$\{12\}$
3 or 27	$\{21,\ldots,24\} \cup \{28\}$
4 or 11	$\{4\} \cup \{9,\ldots,12\} \cup \{20\} \cup \{29,\ldots,32\}$
6 or 22	$\{20\}$
7 or 20	$\{1,\ldots,4\} \cup \{8\} \cup \{25,\ldots,28\}$
8 or 24	$\{17,\ldots,20\} \cup \{24\} \cup \{29,\ldots,32\}$
9	$\{28\}$
12 or 28	$\{5,\ldots,8\} \cup \{12\} \cup \{21,\ldots,24\}$
13 or 30	$\{17,\ldots,20\} \cup \{25,\ldots,28\}$
14 or 32	$\{24\}$
15 or 23	$\{1,\ldots,4\} \cup \{9,\ldots,12\}$
16 or 25	$\{4\}$
19	$\{17,\ldots,20\} \cup \{28\}$
21 or 29	$\{8\}$

Figure 5. Definition of $J(j)$.

In all blocks j, for $j \in \{1,\ldots,8\} \setminus \{3\}$, we have $[EP[a^0]_R]_j \notin \{0,1\}$ and then $U([EP[a^0]_R]_j) = \{0,1\}^4$. Therefore, as discussed in the previous proofs, $\exists (K^i, K^{i'}) \in \mathbb{M}$ such that $[[a^1]_L]_j := [F^L_{K^1,K^{1'}} \circ F^L_{K^2,K^{2'}} \circ \cdots \circ F^L_{K^n,K^{n'}}(a')]_j = [e]_j \; \forall \, j \in \{1,\ldots,8\} \setminus \{3\}$. For block 3, we have $[EP[a^0]_R]_3 = 1$, therefore $\exists \, (K^i, K^{i'}) \in \mathbb{M}$ such that $(a^1)_i := (F^L_{K^1,K^{1'}} \circ F^L_{K^2,K^{2'}} \circ \cdots \circ F^L_{K^n,K^{n'}}(a'))_i = e_i \; \forall \, i \in \{9,\ldots,11\}$.

Therefore, the only position we cannot assure is equal to e is $i = 12$, therefore $J(1)^c = \{12\}$.

For the rest of the indices j, we use similar arguments to compute sets $J(j)$.

- If $j \in \{1,6,9,14,16,17,21,22,25,29,32\}$, the set $(\{1,\ldots,32\} \setminus \{13,\ldots,16\}) \setminus J(j)$ has only one element. Therefore, as $((a^1)_L)_i = e_i \; \forall \; i \in$J(j), $[EP(a^1_L)]_i \notin \{0,1\} \; \forall \; i \in \{1,\ldots,8\} \setminus \{4\}$, so $U([EP(a^1_L)]_i) = \{0,1\}^4$. Therefore, choosing appropriate $(K^i, K^{i'}) \in \mathbb{M}_{d'}$ we get $a^2 := F^R_{K^1,K^{1'}} \circ \cdots \circ F^R_{K^n,K^{n'}}(a^1)$, such that $([a^2]_R)_i = e_i \; \forall \, i \in \{1,\ldots,32\} \setminus \{13,\ldots,16\}$ (Remark 1).

  Therefore, we have $[EP(a^2_R)]_i \notin \{0,1\} \; \forall \, i \in \{1,\ldots,8\} \setminus \{4\}$, so $U([EP(a^2_L)]_i) = \{0,1\}^4$. Now, choosing adequate $(K^i, K^{i'}) \in \mathbb{M}_{d'}$, we can have $a^3 := F^L_{K^1,K^{1'}} \circ \cdots \circ F^L_{K^n,K^{n'}}(a^2)$, such that $(a^3)_i = e_i \; \forall \, i \in \{1,\ldots,32\} \setminus \{13,\ldots,16\}$. Therefore, for $a'' := a^3$ we have the desired result.

  Hence, we have seen that the lemma holds if $a'_j = 1$ for $j \in \{1,6,9,14,16,17,21,22,25,29,32\}$.
- For indices $j \in \{1,\ldots,32\} \setminus \{2,5,10,18,26,31\}$, we have $J(j) \cap \{1,6,9,14,16,17,21,22,25,29,32\} \neq \emptyset$. Therefore, we are in the case where $\exists \, j \in \{1,6,9,14,16,17,21,22,25,29,32\}$ such that $(a^1)_i = 1$, and carrying out the same procedure as the one to get a^3 from a', we get a'' satisfying $(a'')_i = e_i \; \forall i \in \{1,\ldots,32\} \setminus \{13,\ldots,16\}$.

□

Lemma 6. $\forall \, a'' \in \{0,1\}^{64} \setminus \{0,d\} \; : a''_i = e_i \; \forall \, i \in \{1,\ldots,32\} \setminus \{13,\ldots,16\}$, $\exists \, z \in \{0,1\}^{32} : a'' \sim (e,z)$.

Proof. According to Table 2, $[(EP(a)_L)]_4$ corresponds to positions 26, 5, 18, 31, and 2. Since $\{2,5,10,18,26,31\} \cap \{13,\ldots,16\} = \emptyset$, we know $(a''_L)_i = e_i, \; \forall \, i \in \{2,5,10,18,26,31\}$. Therefore, $[(EP(a)_L)]_4 = (1,1,1,1,1,0) = 62$ and because of Lemma 2 (c), $U([EP((a'')_L)]_j) = \{0,1\}^4$. Thus, considering appropriate $(K^i, K^{i'})$, we get $(e,z) = F^L_{K^1,K^{1'}} \circ \cdots F^L_{K^n,K^{n'}}(a'')$, for some $z \in \{0,1\}^{32}$. □

Corollary 1. $\forall a \in \{0,1\}^{64} \setminus \{0,d\} \; \exists \, z \in \{0,1\}^{32} : a \sim (e,z)$.

Proof. Considering the chain $a \sim a' \sim a'' \sim (e,z)$, where these elements are as described in the previous lemmata, the result follows. □

Corollary 2. *$G_{0,d}$ is transitive on $\{0,1\}^{64} \setminus \{0,d\}$.*

Proof. Let $a, a' \in \{0,1\}^{64} \setminus \{0, d\}$, by Lemma 6 and Corollary 1, $\exists\ z, z' \in \{0,1\}^{32} : a \sim (e,z) \sim (e,z') \sim a'$. □

Corollary 3. *G_0 is transitive on $\{0,1\}^{64} \setminus \{0\}$.*

Proof. Because of Corollary 1, it is enough to show that $\exists g \in G_0$ such that $g(d) \neq d$.

Note that since $g \in G_0$, then $g(d) \neq 0$.

Let $(K, K') \in \mathbb{M} \setminus \mathbb{M}_{d'}$, then $S(K) = S(K')$ and $S(K \oplus d') \neq S(K' \oplus d')$. Therefore, $F^R_{K,K'}(d) = (d_L, d_R \oplus) S(K \oplus d') \oplus S(K' \oplus d') \neq d$, and $F^R_{K,K'} \in G_0$. □

Lemma 7. *If G_0 is transitive on $\{0,1\}^{64} \setminus \{(0,\ldots,0)\}$ and $G_{0,d}$ is transitive on $\{0,1\}^{64} \setminus \{(0,\ldots,0), d\}$, then G is 3-transitive on $\{0,1\}^{64}$.*

Proof. It follows immediately from [17] (Theorem 9.1). □

Once we have shown that G is a 3-transitive subgroup of $A_{2^{64}}$, it is not particularly difficult to verify that G is actually equal to the alternating group on 2^{64} points.

Theorem 1. *The round functions of DESL generate the alternating group, i.e., $G = A_{2^{64}}$.*

Proof. We refer to the proof of Theorem 1 in [7], since the same proof applies here. □

4. Applying MRHS to DESL and DES

The previous section focuses on a structural group-theoretic property which does not take the actual number of DESL rounds into account. Subsequently, we studied an algebraic attack against reduced and full round versions of DESL and compared the behavior of the attack with the situation for DES. The underlying question is, to what extent does the modified S-box change the complexity of an algebraic attack?

4.1. Symbol Creation for DESL

Since the structure of DES and DESL is the same, the process for creating the A-parts of MRHS symbols for DESL is the same as that for DES, which is described nicely in [12] (pp. 50–53). The only difference is that the L-part of each symbol will not correspond to a DES S-box, but instead to the DESL S-box. This L-part is given as

$$\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & F & F & F & F & F & F & F & F \\
0 & 0 & 0 & 0 & F & F & F & F & 0 & 0 & 0 & 0 & F & F & F & F \\
0 & 0 & F & F & 0 & 0 & F & F & 0 & 0 & F & F & 0 & 0 & F & F \\
0 & F & 0 & F & 0 & F & 0 & F & 0 & F & 0 & F & 0 & F & 0 & F \\
3 & 3 & 3 & 3 & 3 & 3 & 3 & 3 & 3 & 3 & 3 & 3 & 3 & 3 & 3 & 3 \\
5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 5 \\
8 & 5 & E & 3 & 6 & 9 & 6 & 9 & 6 & 6 & 9 & 9 & A & C & 3 & 5 \\
E & 9 & 4 & 3 & 1 & 6 & F & 8 & 9 & 7 & 2 & C & 6 & C & 9 & 3 \\
8 & B & D & 6 & 7 & 4 & 8 & 3 & 1 & E & 6 & 1 & C & 9 & 3 & E \\
6 & 9 & 9 & A & 5 & 9 & 6 & 6 & 6 & 5 & 6 & 9 & 5 & A & A & 9
\end{bmatrix},$$

where each entry is written as standard hex notation to save space. Note that the top six rows correspond to each of the possible inputs to an S-box, and the bottom four rows correspond to the output of the S-box. For example, if the input to the S-box is 000000, then the output is 1110, both being readable from the first column of this matrix. If the input is 000001, then the output is 0101, both being readable from the second column. Further, if the input is 000010, the output is 0101, and if the input is 000011, the output is 0000.

4.2. Results

For serious ciphers, very often the first MRHS action cycle of agreeing, gluing, and equation extracting (that is, until a guess is called for) will not be sufficient to discover the key, so guesses of the key variables must be committed. Naturally, the fewer guesses required, the better an attack is deemed to be. We give the name δ to the number of key bits we must guess before we discover the whole key through an MRHS attack.

For our attacks, we use a machine called Blue with the following specifications: two quad-core Xeon E5520 2.26 GHz processors (though only one core was used), 24 GB of RAM, using Windows 7 Server (Standard Edition). The ciphertext was `0123456789ABCDEF`, and the key was the first 56 bits of the SHA-1 hash of "Katalina" (without quotes).

Under these conditions, DESL was attacked on Blue, varying both the number of rounds of the cipher and the threshold of MRHS. The results are summarized in Table 3, with the note that the threshold listed is actually the base 2 logarithm of the actual threshold, so we always choose a power of 2 for the number of columns each L-part is allowed to grow to.

Table 3. DESL δ on Blue, for varying rounds and thresholds.

	Rounds of DESL						
Threshold	**4**	**6**	**8**	**10**	**12**	**14**	**16**
20	0	34	36	36	40	38	40
21	0	34	39	37	39	39	42
22	0	33	39	37	38	43	38
23	0	33	38	45	46	48	46

We can see from this data that four rounds of DESL could be handled in the initial turn of an MRHS attack, but things became more complicated with more rounds. For more than six rounds it was not at all guaranteed that an increased threshold would actually help with the computation. Only for twelve rounds did we see an improvement with increased threshold, but once we moved to a threshold of 23, δ increased dramatically.

By way of contrast, DES was attacked on Blue varying the number of rounds and threshold. The results are summarized in Table 4.

Table 4. DES δ on Blue, varying rounds and thresholds.

	Rounds of DES						
Threshold	**4**	**6**	**8**	**10**	**12**	**14**	**16**
20	1 (+1)	35 (+1)	36 (+0)	36 (+0)	41 (+1)	41 (+3)	40 (+0)
21	0 (+0)	35 (+1)	39 (+0)	37 (+0)	39 (+0)	40 (+1)	39 (−3)
22	0 (+0)	32 (−1)	39 (+0)	37 (+0)	38 (+0)	40 (−3)	38 (+0)
23	0 (+0)	33 (+0)	39 (+1)	43 (−2)	46 (+0)	48 (+0)	46 (+0)

Overall, DESL was about as secure as DES from an MRHS perspective, though there were two occasions where DESL required three more bits to guess before recovering the entire key.

We remark in passing that it was conjectured by Schoonen in [12] (Hypothesis 5.1) that for 7–16 rounds of DES, δ would always be 56 minus the (base 2 logarithm of the) threshold, but Table 4 makes it plain that this was not the case.

5. Conclusions

Unlike DES, the DES Lightweight extension (DESL) uses a single S-box. The security of DESL against a number of common types of attacks has already been argued in the literature. In this work

we establish that the round functions of DESL generate the same permutation group as the round functions of DES, namely, the alternating group on 2^{64} points. Moreover, based on our work, DESL appeared to offer comparable resistance to MRHS-based algebraic attacks as DES. Therefore, from these algebraic points of view, DESL has no disadvantage compared to DES, and the structural properties of DESL remain an interesting cryptanalytic topic of study.

Author Contributions: Individual contributions to this article: conceptualization, K.M, R.S., and A.S.C.; methodology, K.M., R.S., and A.S.C.; validation, K.M., R.S., and A.S.C.; formal analysis, A.S.C.; software, K.M. and R.S.; investigation, K.M., R.S., and A.S.C.; resources, R.S. and A.S.C.; writing—original draft preparation, K.M., R.S., and A.S.C.; writing—review and editing, K.M., R.S., and A.S.C.; project administration, R.S. and A.S.C.; funding acquisition, R.S. and A.S.C.

Funding: This research was funded in part by the NATO Science for Peace and Security Programme under grant G5448 and through research project MTM2017-83506-C2-2-P by the Spanish MICINN.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Daley, W.M.; Kammer, R.G. Data Encryption Standard (DES). In *Federal Information Processing Standards Publication*; National Institute of Standards and Technology: Gaithersburg, MD, USA 1999.
2. Leander, G.; Paar, C.; Poschmann, A.; Schramm, K. New Lightweight DES Variants. In *Fast Software Encryption, 14th International Workshop, FSE 2007*; Lecture Notes in Computer Science; Biryukov, A., Ed.; International Association for Cryptologic Research, Springer: New York, NY, USA, 2007; Volume 4593, pp. 196–210.
3. Priyanka, A.A.; Saibal, K.P. A Survey of Cryptanalytic Attacks on Lightweight Block Ciphers. *Int. J. Comput. Sci. Inf. Technol. Secur.* **2012**, *2*, 472–481.
4. Sun, S.; Hu, L.; Qiao, K.; Ma, X.; Shan, J.; Song, L. Improvement on the Method for Automatic Differential Analysis and Its Application to Two Lightweight Block Ciphers DESL and LBlock-s. In Proceedings of the 2015 10th International Workshop on Security Advances in Information and Computer Security, IWSEC, Nara, Japan, 26–28 August 2015; pp. 97–111.
5. Hatzivasilis, G.; Fysarakis, K.; Papaefstathiou, I.; Manifavas, C. A review of lightweight block ciphers. *J. Cryptogr. Eng.* **2018**, *8*, 141–184. [CrossRef]
6. Ji, F.; Zhang, W.; Ding, T. Improving Matsui's Search Algorithm for the Best Differential/Linear Trails and its Applications for DES, DESL and GIFT. Cryptology ePrint Archive, Report 2019/1190. 2019. Available online: http://eprint.iacr.org/2019/1190 (accessed on 14 November 2019).
7. Wernsdorf, R. The One-Round Functions of the DES Generate the Alternating Group. In *Advances in Cryptology—EUROCRYPT '92*; Lecture Notes in Computer Science; Rueppel, R.A., Ed.; Springer: New York, NY, USA, 1993; Volume 658, pp. 99–112.
8. Raddum, H.; Semaev, I. Solving Multiple Right Hand Sides linear equations. *Des. Codes Cryptogr.* **2008**, *49*, 147–160. [CrossRef]
9. Geiselmann, W.; Matheis, K.; Steinwandt, R. PET SNAKE: A Special Purpose Architecture to Implement an Algebraic Attack in Hardware. In *Transactions on Computational Science X*; Lecture Notes in Computer Science; Springer: New York, NY, USA, 2010; Volume 6340, pp. 298–328.
10. Håvard, R.; Zajac, P. MRHS solver based on linear algebra and exhaustive search. *J. Math. Cryptol.* **2018**, *12*, 143–157.
11. Zajac, P. Upper bounds on the complexity of algebraic cryptanalysis of ciphers with a low multiplicative complexity. *Des. Codes Cryptogr.* **2017**, *82*, 43–56. [CrossRef]
12. Schoonen, A.C.C. Multiple Right-Hand Side Equations. Master's Thesis, Department of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands, 2008. Available online: http://alexandria.tue.nl/extra1/afstversl/wsk-i/schoonen2008.pdf (accessed on 14 November 2019).
13. Davio, M.; Desmedt, Y.; Fosséprez, M.; Govaerts, R.; Hulsbosch, J.; Neutjens, P.; Piret, P.; Quisquater, J.J.; Vandewalle, J.; Wouters, P. Analytical Characteristics of the DES. In *Advanves in Cryptology—CRYPTO '83*; Chaum, D., Ed.; Plenum Press: New York, NY, USA, 1984; pp. 171–202.

14. Raddum, H. MRHS Equation Systems. In *Selected Areas in Cryptography—SAC 2007*; Lecture Notes in Computer Science; Carlisle Adams, A.M., Wiener, M., Eds.; Springer: New York, NY, USA, 2007; Volume 4876, pp. 232–245.
15. Even, S.; Goldreich, O. DES-Like Functions Can Generate the Alternating Group. *IEEE Trans. Inf. Theory* **1983**, *29*, 863–865. [CrossRef]
16. Foundation, P.S. Python Programming Language—Offical Website. 2010. Available online: http://www.python.org (accessed on 14 November 2019).
17. Wielandt, H. *Finite Permutation Groups*; Academic Press: Cambridge, MA, USA, 1964.