Differential Factors Revisited: Corrected Attacks on PRESENT and SERPENT

Cihangir Tezcan$^{1,2,3(\boxtimes)}$

[1] Department of Mathematics, Middle East Technical University, Ankara, Turkey
cihangir@metu.edu.tr

[2] CYDES Laboratory, Department of Cyber Security, Institute of Informatics,
Middle East Technical University, Ankara, Turkey

[3] Department of Cryptography, Institute of Applied Mathematics,
Middle East Technical University, Ankara, Turkey

Abstract. Differential factors, which prevent the attacker to distinguish some of the guessed keys corresponding to an active S-box during a differential attack on a block cipher, are recently introduced at Lightsec 2014 and used to reduce the time complexities of the previous differential-linear attacks on Serpent. Key recovery attacks generally consists of two parts: Key guess using the distinguisher and exhaustive search on the remaining key bits. Thus, we show that differential factors can reduce the time complexity of the former and increase the latter since the attacker does not need to guess the keys which cannot be distinguished. As an example for the latter, we show that the best known differential attack on Present overlooked its six differential factors and the corrected attack actually requires a time complexity increased by a factor of 64. Moreover, we show that differential factors also reduce data complexity of the differential attacks since less number of pairs are required to distinguish the correct key when the key space is reduced. This reduction in data complexity also reduces the time complexity. By using Serpent's differential factors, we further reduce the data and time complexity of the differential-linear attacks on this cipher to obtain the best attacks.

Keywords: S-box · Differential factor · Serpent · Present

1 Introduction

Confusion layer of symmetric cryptography algorithms mostly consists of substitution boxes (S-boxes) and in order to provide better security against known attacks, S-boxes are selected depending on their cryptographic properties. Low non-linear and differential uniformity [24] provide resistance against linear [21]

C. Tezcan—This work was supported by The Scientific and Technological Research Council of Turkey (TÜBİTAK) under the grant 115E447 titled "Quasi-Differential Factors and Time Complexity of Block Cipher Attacks". Part of this work was done when the author was visiting Department of Electrical Engineering, ESAT/COSIC, Katholieke Universiteit Leuven, Belgium.

T. Güneysu et al. (Eds.): LightSec 2015, LNCS 9542, pp. 21–33, 2016.
DOI: 10.1007/978-3-319-29078-2_2

and differential cryptanalysis [6], respectively and most of the time these are the only properties designers focus on. However, it is shown that resistance against algebraic [11] and cube [12] attacks can be obtained by high algebraic degree and branch number. Moreover, lack of undisturbed bits [28] provides resistance against truncated [17], impossible [2], and improbable [27] differential cryptanalysis. It was shown in [20] that undisturbed bits are actually linear structures in coordinate functions. Thus, it is better to avoid linear structures to get better security against these kind of attacks. Resistance against side-channel attacks like differential power analysis [18] can be obtained depending on the number of shares [7] in threshold implementations. Implementation invariant resistance against these attacks can be obtained by S-boxes with a low transparency order [25] but low transparency order is not sufficient alone to directly achieve a satisfying level of security [10].

Recently it was shown in [29] that S-boxes may have parameters called differential factors which does not change the output difference of an S-box when they are XORed with the input pair. Thus, some counters of the guessed keys in a differential variant attack become the same, which prevents the attacker from fully capturing the attacked round keys. This may benefit the attacker because reduction in the attacked key space reduces the time complexity of many attacks. For instance, the 10, 11, and 12-round differential-linear attacks of [13] on Serpent [1] tries to capture 40, 48, and 160 bits of the key, respectively. However, it was shown in [29] that these attacks can only obtain advantages of 38, 46, and 157 bits on the key due to differential factors and these attacks can actually be performed with time complexities reduced by a factor of 4, 4, and 8, respectively.

Most of the statistical attacks on blocks ciphers consists of two steps: Capturing partial information about the key via distinguishers and obtaining the remaining key bits via exhaustive search. We note that although differential factors reduce the time complexity of the former, they increase the time complexity of the latter. In this work we use this observation to correct the differential attack of [31] on Present [9] which due to six differential factors requires a time complexity of 2^{70} memory accesses instead of 2^{64} memory accesses as it is claimed in [31].

Moreover, we show that differential factors also reduces the data complexity of differential attacks since the reduction in the key space allows us to use less number of pairs to distinguish the correct key. This observation also reduces the memory required to store the key counters and time complexity since the attack procedure is repeated for every data. We use our findings to obtain best differential-linear attacks on Serpent by reducing the data and time complexity of the previous attacks.

2 Differential Factors

Definition 1 ([29]). *Let S be a function from $\mathbb{F}_2^n$ to $\mathbb{F}_2^m$. For all $x, y \in \mathbb{F}_2^n$ that satisfy $S(x) \oplus S(y) = \mu$, if we also have $S(x \oplus \lambda) \oplus S(y \oplus \lambda) = \mu$, then we say that the S-box has a* differential factor *λ for the output difference μ. (i.e. μ remains invariant for λ).*

PRESENT's S-box is given as an example in Table 1 which has $\lambda = 1$ as a differential factor for $\mu = 5$.

Table 1. PRESENT's S-box ordered in pairs where the output difference is $\mu = 5$. Note that XOR of any pair with $\lambda = 1$ gives another pair that has output difference $\mu = 5$.

x	5	1	E	C	F	D	8	2	B	7	4	0	6	A	3	9
S(x)	0	5	1	4	2	7	3	6	8	D	9	C	A	F	B	E

The following theorem shows that the number of differential factors of an S-box is the same with the number of differential factors of its inverse. Moreover, it also provides the differential factors of the inverse S-box when we know the differential factors of the S-box. Hence, there is no need to check the differential factors of the inverse of S-boxes. This theorem is useful in practice since inverse of an S-box is used for decryption in substitution permutation networks.

Theorem 1 **([29]).** *If a bijective S-box S has a differential factor λ for an output difference μ, then S^{-1} has a differential factor μ for the output difference λ.*

Moreover, differential factors for the same μ form a vector space.

Theorem 2 **([29]).** *If λ_1 and λ_2 are differential factors for an output difference μ, then $\lambda_1 \oplus \lambda_2$ is also a differential factor for the output difference μ. i.e. All differential factors λ_i for μ form a vector space.*

Differential factors are observed mostly in small S-boxes. For instance, 73.3 % of all 3×3 bijective S-boxes contain differential factors. Moreover, a list of ciphers and hash functions whose 4×4 S-boxes contain differential factors are provided in [29].

2.1 Differential Factors and Time Complexity

We start by recalling the definition of advantage.

Definition 2 **([26]).** *If an attack on an m-bit key gets the correct value ranked among the top r out of 2^m possible candidates, we say the attack obtained an $(m - log(r))$-bit* advantage *over exhaustive search.*

Differential attacks on block ciphers use a differential as a distinguisher and the attack is performed by adding a few more rounds on the top or bottom of this differential. Pairs that may satisfy this differential are partially encrypted or decrypted under the possible subkeys and counters of these keys are incremented when the differential is satisfied. In a one round attack, one can obtain these counters just by looking at a precomputed table. However, more complicated attacks may require to repeat partial encryptions under every possible subkey. In these cases, differential factors reduce the time complexity of this step as follows.

Theorem 3 ([29]). *In a block cipher let an S-box S contain a differential factor λ for an output difference μ and the partial round key k is XORed with the input of S. If an input pair provides the output difference μ under a partial subkey k', then the same output difference is observed under the partial subkey $k' \oplus \lambda$. Therefore, during a differential attack involving the guess of a partial subkey corresponding to the output difference μ, the advantage of the cryptanalyst is reduced by* 1 *bit and the time complexity of this key guess step is halved.*

Proof. In a differential attack for any key k', k' and $k' \oplus \lambda$ would get the same number of hits since λ is a differential factor. Hence the attacker cannot distinguish half of the guessed keys with the other half. Therefore during the key guessing step, the attacker does not need to guess half of the keys. Thus, the time complexity of this step is halved. □

From Theorems 2 and 3 we obtain the following Corollary.

Corollary 1 ([29]). *During a differential attack involving the guess of a partial subkey corresponding to the output difference μ of an S-box that has a vector space of differential factors of dimension r for μ, the advantage of the cryptanalyst is reduced by r bits and the time complexity of the key guess step is reduced by a factor of 2^r.*

Most of the statistical attacks on blocks ciphers first tries to capture partial information about the secret key and then the full key is obtained by exhaustive search. Thus, if possible, the attacker tries to balance these two steps to obtain the optimal time complexity for the attack. Although differential factors reduce the time complexity of the former, they increase the time complexity of the latter. We provide our first observation in Corollary 2.

Corollary 2. *Differential factors reduce the time complexity of capturing partial information about the key which uses differentials but they increase the time complexity of the exhaustive search for obtaining the remaining key bits.*

Thus, the attacker should take into account differential factors when trying to balance the time complexities of these two parts. We show the importance of Corollary 2 in Sect. 4 by proving that Wang's differential attack on Present is actually wrong and the corrected attack requires 2^{70} memory accesses instead of 2^{64} as it is claimed in [31].

2.2 Differential Factors and Data Complexity

Statistical attacks use a distinguisher which is observed with different probabilities p_0 and p for the correct key and the wrong keys, respectively. For instance, the attacker uses N plaintext pairs in differential attack and counts the times each subkey satisfies this distinguisher. The correct key is expected to be above some threshold T since we have $p_0 > p$. Thus, the number of hits a wrong (right) subkey gets can be seen as a random variable of a binomial distribution with parameters N and p (p_0). We denote the non-detection error probability, which is the probability of the counter for the right subkey to be less than T, by p_{nd}; and the false alarm error probability, which is the probability of the counter for a wrong subkey to be higher than or equal to T, by p_{fa}.

Theorem 4. *Differential factors reduce the key space for the key guess process and therefore reduce the data complexity of the attack. Thus, memory required to keep the counters for the guessed keys also reduces. Reduction in the data complexity may also reduce the time complexity depending on the attack.*

Proof. The amount of required plaintext pairs N to perform the attack with the desired success probability depends also on the number of wrong keys. Because they determine the number of binomial distributions from which we try to distinguish the correct key. Since the existence of the differential factors reduces the wrong subkey space, the number of pairs required to perform the attack also reduces. Thus, memory required to keep the counters for the guessed keys also reduces. Moreover, the attack procedure is repeated for every pair in most of the attacks. Therefore, this reduction in the data complexity further reduces the time complexity. □

When differential factors were introduced in [29], their effect on the data and memory complexity were overlooked. By using differential factors that appear in the differential-linear attacks on Serpent, we reduce the data complexity of these attacks in Sect. 5. Since the data and time complexities of these attacks are directly proportional, we further reduce the time complexities of these attacks. Moreover, we reduce the data and memory complexity of the differential attack on Present in Sect. 4 using Theorem 4.

Success probability of differential attacks are generally calculated easily using Selçuk's formula [26] and it is used in the original Present attack. However, in this work we use Blondeau-Gérard-Tillich algorithm [8] since it is valid for both differential and differential-linear attacks. This algorithm takes p, p_0, p_{nd}, and p_{fa} as input and provides N and T as output.

3 PRESENT and SERPENT

Present [9] is a 31-round SPN (Substitution Permutation Network) type block cipher with block size of 64 bits that supports 80 and 128-bit secret key. It has been internationally standardized by ISO/IEC 29192-2:2012 [16] as a lightweight block cipher. Round function of Present, which is depicted in Fig. 1, is same for both versions of Present and consists of standard operations such as subkey XOR, substitution and permutation: At the beginning of each round, 64-bit input of the round function is XORed with the subkey. Just after the subkey XOR, 16 identical 4 × 4-bit S-boxes are used in parallel as a non-linear substitution layer and finally a permutation is performed so as to provide diffusion.

Serpent [1] was designed by Anderson, Biham and Knudsen in 1998. It was submitted to the AES contest and became one of the five finalists. It has a block

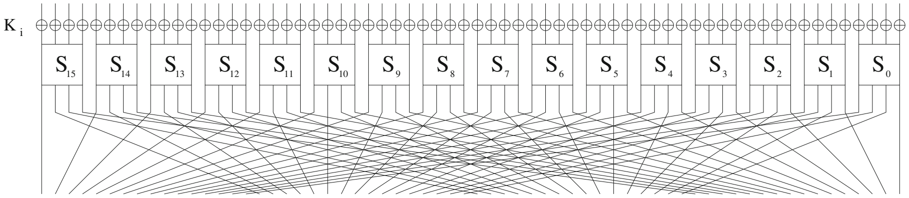

Fig. 1. Round function of PRESENT

size of 128 bits and accepts any key size of length 0 to 256 bits. It is a 32-round SPN, where each round consists of key mixing, a layer of S-boxes and a linear transformation.

The 128-bit input value before round i is denoted by $\hat{B}_i$, $i \in \{0, \ldots, 31\}$. Each $\hat{B}_i$ is composed of four 32-bit words X_0, X_1, X_2, X_3 where X_0 is the leftmost word.

Three round operations are specified as follows:

1. Key Mixing: At each round R_i, a 128-bit subkey K_i is XORed with the current intermediate data $\hat{B}_i$.
2. S-boxes: At each round, R_i uses a single S-box S_j, where $i \equiv j \pmod 8$ and $i \in \{0, \ldots, 31\}$, 32 times in parallel. In this paper, we use the bitsliced version of SERPENT. For example, in the first round the first copy of S_0 takes the least significant bits from X_0, X_1, X_2, X_3 and returns the output to the same bits. Thus, we obtain 32 4-bit slices referred as b_i's, where $i \in \{0, \ldots, 31\}$ and b_0 is the right most slice.
3. Linear Transformation: The four 32-bit words X_0, X_1, X_2, X_3 are linearly mixed by the following linear operations:

$$\begin{aligned}
X_0 &:= X_0 \lll 13\\
X_2 &:= X_2 \lll 3\\
X_1 &:= X_1 \oplus X_0 \oplus X_2\\
X_3 &:= X_3 \oplus X_2 \oplus (X_0 \ll 3)\\
X_1 &:= X_1 \lll 1\\
X_3 &:= X_3 \lll 7\\
X_0 &:= X_0 \oplus X_1 \oplus X_3\\
X_2 &:= X_2 \oplus X_3 \oplus (X_1 \ll 7)\\
X_0 &:= X_0 \lll 5\\
X_2 &:= X_2 \lll 22\\
\hat{B}_{i+1} &:= X_0, X_1, X_2, X_3
\end{aligned}$$

   where $\lll$ denotes the left rotation operation and $\ll$ denotes the left shift operation.

32-round SERPENT cipher may be described by the following equations:

$$\hat{B}_0 := P \qquad \hat{B}_{i+1} := R_i(\hat{B}_i),\ i \in \{0, \ldots, 31\} \qquad C := \hat{B}_{32}$$

where

$$R_i(X) = LT(\hat{S}_i(X \oplus K_i)),\ i \in \{0, \ldots, 30\}$$
$$R_{31}(X) = \hat{S}_{31}(X \oplus K_{31}) \oplus K_{32}$$

and $\hat{S}_i$ is the application of the S-box $S_{(i\ (mod\ 8))}$ 32 times in parallel, and LT is the linear transformation.

In this paper, we use P, S, I to the denote output of the permutation layer, output of the substitution layer, and input of a round, respectively.

Differential factors of Present and Serpent's S-boxes are provided in Table 2.

Table 2. Differential factors of Present and Serpent's S-boxes

S-box	0123456789ABCDEF	λ	μ
Present	C56B90AD3EF84712	1_x	5_x
Present	C56B90AD3EF84712	F_x	F_x
Serpent S_0	38F1A65BED42709C	4_x	4_x
Serpent S_0	38F1A65BED42709C	D_x	F_x
Serpent S_1	FC27905A1BE86D34	4_x	4_x
Serpent S_1	FC27905A1BE86D34	F_x	E_x
Serpent S_2	86793CAFD1E40B52	2_x	1_x
Serpent S_2	86793CAFD1E40B52	4_x	D_x
Serpent S_6	72C5846BE91FD3A0	6_x	2_x
Serpent S_6	72C5846BE91FD3A0	F_x	F_x

4 Differential Attacks on PRESENT

The best known differential attack on Present is obtained in [31] by adding two rounds to the bottom of the 24 different 14-round differentials which has different input and same output difference. These differentials hold with probability $p = 2^{-62}$ and Δ_1 is an example for these differentials

$$\Delta_1 : 0700000000000700 \rightarrow_{14r} 0000000900000009$$

This differential attack captures 32 bits of the key with a time complexity of $2^{33.18}$ 2-round Present encryptions, a data complexity of 2^{64} chosen plaintexts, and a memory complexity of 2^{32} 6-bit counters. This part of the attack works with a success probability of 99.9999939 % and then the remaining 48 bits are obtained via exhaustive search which requires 2^{48} 16-round Present encryptions or equivalently 2^{64} memory accesses.

It is claimed that these 14-round characteristics activates two S-boxes at the round 15 and due to the undisturbed bits of the S-box, it activates at most six S-boxes instead of eight in round 16. If activated, the input difference of these S-boxes must be 1. PRESENT's S-box has a differential factor $\lambda = 1$ for $\mu = 5$. Thus, the inverse of the S-box has a differential factor $\lambda = 5$ for $\mu = 1$ by Theorem 1. Since $\mu = 1$ coincides with the input difference of these six S-boxes, the advantage of this attack is actually 26 bits instead of 32 bits. This theoretical result can easily be observed experimentally by performing this attack by removing the first few rounds of the 14-round differential so that it remains within our computational power. This attack is summarized in Table 3.

Table 3. 16-round differential-linear attack of [31]. Output differences μ that contain differential factors, which is $\lambda = 1$ for the inverse S-box, are shown in bold.

Rounds	Differences in bits															
	x_{15}	x_{14}	x_{13}	x_{12}	x_{11}	x_{10}	x_9	x_8	x_7	x_6	x_5	x_4	x_3	x_2	x_1	x_0
$X_{1,I}$	0000	0111	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000	0111	0000	0000
14-Round Differential $\varDelta_1$																
$X_{14,P}$	0000	0000	0000	0000	0000	0000	0000	1001	0000	0000	0000	0000	0000	0000	0000	1001
$X_{15,S}$	0000	0000	0000	0000	0000	0000	0000	???0	0000	0000	0000	0000	0000	0000	0000	???0
$X_{15,P}$	0000	000?	0000	**000?**	0000	**000?**	0000	**000?**	0000	**000?**	0000	**000?**	0000	0000	0000	0000
$X_{16,S}$	0000	????	0000	????	0000	????	0000	????	0000	????	0000	????	0000	0000	0000	0000

This observation reduces the time complexity of the first part of the attack to $2^{27.18}$ 2-round PRESENT encryptions and the memory complexity to 2^{26} 6-bit counters. However, the time complexity of exhaustive search for the remaining bits of the key is 2^{54} 16-round PRESENT encryptions or equivalently 2^{70} memory accesses. Therefore, the correct time complexity of Wang's differential attack on PRESENT [31] is 2^{70} memory accesses, instead of 2^{64}.

Another correction we make for this attack is due to Theorem 4. The original attack uses the whole codebook and achieves a success probability of 0.999999939. However, the original attack tries to capture 32 bits of the key. Thus, we need $p_{fa} \leq 2^{-33}$ to have only the correct key counter above the threshold T. Since the six differential factors used in the attack reduces the key space for the key guess process, we can choose $p_{fa} = 2^{-27}$ to prevent any wrong key to get a counter higher than T. Using the BLONDEAU-GÉRARD-TILLICH algorithm with parameters $p = 2^{-64}$, $p_0 = 24 \cdot 2^{-62}$, $p_{nd} = 1 - 0.999999939$, and $p_{fa} = 2^{-27}$ shows that

Table 4. Comparison of Wang's original differential attack on PRESENT and our corrected one. *MA - Memory Accesses, b - bits, CP - Chosen Plaintexts.*

	Rounds	Data	Time	Memory	Success	Reference
Original	16	2^{64} CP	2^{64} MA	$6 \cdot 2^{32}$ b	99.9999939 %	[31]
Corrected	16	$2^{63.58}$ CP	2^{70} MA	$6 \cdot 2^{26}$ b	99.9999939 %	Sect. 4

this attack can be performed with $2^{63.58}$ data complexity to achieve the success probability of the original attack. This change reduces the memory required for the guessed key counters to $6 \cdot 2^{26}$ bits from $6 \cdot 2^{32}$ bits. These corrections are summarized in Table 4.

5 Differential-Linear Attacks on SERPENT

The most successful differential-linear attacks on Serpent were provided by Dunkelman *et al.* in [13] for 10, 11, and 12 rounds for the key sizes 128, 192, and 256, respectively. These attacks combine the 3-round differential

$$\Delta : 0000000000000000000000040050000 \rightarrow 0??00?000?000000000?00?0??0??0?0$$

Table 5. 12-round differential-linear attack of [13]. Output differences μ that contain differential factors, which are $\mu = 4$ and $\mu = E$ for S_1 and $\mu = 4$ for S_0, are shown in bold. Undisturbed bits are shown in italic.

Input	X_0:	????	????	0???	0???	????	????	????	00??
	X_1:	????	????	0???	0???	????	????	????	00??
	X_2:	????	????	0???	0???	????	???*1*	????	00??
	X_3:	????	????	0???	0???	????	????	????	00??
S_0	X_0:	??0?	00?0	0**0**00	0?00	00?0	000**0**	00??	00??
	X_1:	??0?	????	0**0**?0	0???	0???	???**0**	0?00	0000
	X_2:	000?	00??	0**?**?0	0?00	??00	?00**1**	0?00	0000
	X_3:	?0??	?0??	0**0**??	0???	??0?	0??**0**	?001	0000
LT	X_0:	?000	0000	0000	0??0	0?00	?000	0000	0000
	X_1:	?000	0000	0000	0??0	0?00	?000	0000	0000
	X_2:	?000	0000	0000	0??0	0?00	?000	0000	0000
	X_3:	?000	0000	0000	0*1*?0	0?00	*1*000	0000	0000
S_1	X_0:	0000	0000	0000	0100	0**0**00	**0**000	0000	0000
	X_1:	1000	0000	0000	0010	0**1**00	**0**000	0000	0000
	X_2:	0000	0000	0000	0000	0**1**00	**1**000	0000	0000
	X_3:	0000	0000	0000	0010	0**1**00	**0**000	0000	0000
LT	X_0:	0000	0000	0000	0000	0000	0000	0001	0000
	X_1:	0000	0000	0000	0000	0000	0000	0000	0000
	X_2:	0000	0000	0000	0000	0000	0000	1001	0000
	X_3:	0000	0000	0000	0000	0000	0000	0000	0000
9-Round Differential-Linear Characteristic $\Delta \circ \Lambda$									
Last Round									

Table 6. Summary of attacks on Serpent. Our observations on differential factors in Theorem 4 convert the attacks of [29] to the best attacks for this cipher. *En - Encryptions, MA - Memory Accesses, B - bytes, AC - Adaptive Chosen Plaintexts, CP - Chosen Plaintexts, KP - Known Plaintexts.*

# Rounds	Attack Type	Key Size	Data	Time	Memory	Advantage	Success	Reference
6	Meet-in-the-middle	256	512 KP	2^{247} En	2^{246} B	-	-	[19]
6	Differential	All	2^{83} CP	2^{90} En	2^{40} B	-	-	[19]
6	Differential	All	2^{71} CP	2^{103} En	2^{75} B	-	-	[19]
6	Differential	192, 256	2^{41} CP	2^{163} En	2^{45} B	124	-	[19]
7	Differential	256	2^{122} CP	2^{248} En	2^{126} B	128	-	[19]
7	Improbable	All	$2^{116.85}$ CP	$2^{117.57}$ En	2^{113} B	112	99.9 %	[30]
7	Differential	All	2^{84} CP	2^{85} MA	2^{56} B	-	-	[4]
10	Rectangle	192, 256	$2^{126.3}$ CP	$2^{173.8}$ MA	$2^{131.8}$ B	80	-	[5]
10	Boomerang	192, 256	$2^{126.3}$ AC	$2^{173.8}$ MA	2^{89} B	80	-	[5]
10	Differential-Linear	All	$2^{101.2}$ CP	$2^{115.2}$ En	2^{40} B	40	84 %	[13]
10	Differential-Linear	All	$2^{101.2}$ CP	$2^{113.2}$ En	2^{40} B	38	84 %	[29]
10	**Differential-Linear**	**All**	$2^{100.55}$ **CP**	$2^{112.55}$ **En**	2^{40} **B**	**38**	**84 %**	**Sect.** 5
11	Linear	256	2^{118} KP	2^{214} MA	2^{85} B	140	78.5 %	[3]
11	Multidimensional Linear[a]	All	2^{116} KP	$2^{107.5}$ En	2^{108} B	48	78.5 %	[23]
11	Multidimensional Linear[b]	All	2^{118} KP	$2^{109.5}$ En	2^{104} B	44	78.5 %	[23]
11	Nonlinear	192, 256	$2^{120.36}$ KP	$2^{139.63}$ MA	$2^{133.17}$ B	118	78.5 %	[22]
11	Filtered Nonlinear	192, 256	$2^{114.55}$ KP	$2^{155.76}$ MA	$2^{146.59}$ B	132	78.5 %	[22]
11	Differential-Linear	192, 256	$2^{121.8}$ CP	$2^{135.7}$ En	2^{76} B	48	84 %	[13]
11	Differential-Linear	192, 256	$2^{121.8}$ CP	$2^{133.7}$ En	2^{76} B	46	84 %	[29]
11	**Differential-Linear**	**192, 256**	$2^{120.8}$ **CP**	$2^{132.7}$ **En**	2^{76} **B**	**46**	**84 %**	**Sect.** 5
12	Multidimensional Linear[c]	256	2^{116} KP	$2^{237.5}$ En	2^{125} B	174	78.5 %	[23]
12	Differential-Linear	256	$2^{123.5}$ CP	$2^{249.4}$ En	$2^{128.5}$ B	160	84 %	[13]
12	Differential-Linear	256	$2^{123.5}$ CP	$2^{246.4}$ En	$2^{128.5}$ B	157	84 %	[29]
12	**Differential-Linear**	**256**	$2^{122.45}$ **CP**	$2^{244.35}$ **En**	$2^{128.5}$ **B**	**156**	**84 %**	**Sect.** 5

[a] [22] shows that this attack requires $2^{125.81}$ KP and $2^{101.44}$ En $+2^{114.13}$ MA.
[b] [22] shows that this attack requires $2^{127.78}$ KP and $2^{97.41}$ En $+2^{110.10}$ MA.
[c] [22] shows that this attack requires $\geq 2^{125.81}$ KP $2^{229.44}$ En $+2^{242.13}$ MA.

that has an experimental probability of 2^{-7} with the 6-round linear approximation

$$\Lambda : 20060040000000100100000000000000 \rightarrow 00001000000000005000010000100001$$

of [3] that has bias $q = 2^{-27}$. By performing experiments on the first four rounds of this 9-round differential-linear distinguisher, it was shown in [13] that for the full distinguisher, the probability of pairs to have the same parity in the masked outputs is $1/2 + 2^{-57.75}$. The 11-round attack adds one round to the top of this distinguisher and one round to the bottom. The 12-round attack adds an extra round to the top, which is provided in Table 5. Since the time complexity of the 11-round attack exceeds the exhaustive search of 128 bits, the 10-round attack removes the last round of the distinguisher so that it becomes applicable to Serpent with 128-bit keys. These attacks partially encrypt the top rounds under every possible subkey to obtain the input difference of Δ. Then the last round is decrypted to check the parity of the correct pairs which is actually performed by using precomputed lookup tables.

It was claimed that these attacks can capture 40, 48, and 160 bits of the subkey. Later it was shown in [29] that these attacks overlooked the differential factors of Serpent's S-boxes S_0 and S_1 and the actual advantages are 38, 46, and 157 bits, respectively. Since the attack procedure is repeated for every guess of the subkey bits, existence of differential factors also reduced the time complexities of these attacks by a factor of 4, 4, and 8, respectively.

However, we can further improve these attacks using Theorem 4. We also not that a differential factor was overlooked in the 12-round attack of [29] and therefore the advantage of the attack is actually 156 bits, not 157. Since the differential factors used in the attacks reduce the key space to 38, 46, and 156 bits, we choose the false alarm probability for these attacks in Blondeau-Gérard-Tillich algorithm as $p_{fa} = 2^{-39}$, $p_{fa} = 2^{-47}$, and $p_{fa} = 2^{-157}$, respectively. This analysis shows that these attacks can actually be performed with data complexities $2^{100.55}$, $2^{120.8}$, and $2^{122.45}$ instead of $2^{101.2}$, $2^{121.8}$, and $2^{123.5}$ respectively. Since the data and time complexities of these attacks are directly proportional, we further reduce the time complexities of these attacks to $2^{112.55}$, $2^{132.7}$, and $2^{244.35}$ from $2^{113.2}$, $2^{133.7}$, and $2^{246.4}$, respectively. The attacks on Serpent are summarized in Table 6.

6 Conclusion

Many attacks on ciphers require data, time, and memory complexities that are beyond our computational powers. Thus, experiments on the reduced versions of these theoretical attacks are vital to check the validity in practice. For instance, it was believed that the key bits corresponding to active S-boxes in a differential attack could be fully captured in a differential attack. However, differential factors which are introduced in Lightsec 2014 show that this is not always the case. Differential factors were used to correct the differential-linear attacks on Serpent and the resulting attacks have reduced time complexities. Key recovery attacks generally consists of two parts and in this work we show that differential factors reduce the time complexity of the key guess using a distinguisher step but increase the time complexity of exhaustive search on the remaining key bits step. As an example, we show that the best differential attack on Present in the literature overlooked the differential factors and the attack actually requires 2^{70} memory accesses instead of 2^{64}. Hence, differential factors affect the attacker adversely if the exhaustive search step of the attack requires time complexity more than the key guess step.

Moreover, we further investigate the effects of differential factors and observe that existence of differential factors in an attack reduces the memory complexity required for the key counters and the data complexity. This is because differential factors reduce the size of the key space for the key guess part of the attack which allows the attacker to distinguish the correct key from the wrong ones with a reduced number of data. The reduction in the data complexity may result in a similar reduction in the time complexity since data and time complexities are directly proportional in most of the attacks. Using these observations, we further reduce the data and time complexities of the best differential-linear attacks on Serpent to obtain the best attacks for this cipher. Moreover, we show that the differential attack on Present actually requires less data and memory complexity.

References

1. Biham, E., Anderson, R., Knudsen, L.R.: Serpent: a new block cipher proposal. In: Vaudenay, S. (ed.) FSE 1998. LNCS, vol. 1372, pp. 222–238. Springer, Heidelberg (1998)
2. Biham, E., Biryukov, A., Shamir, A.: Cryptanalysis of skipjack reduced to 31 rounds using impossible differentials. J. Cryptology **18**(4), 291–311 (2005)
3. Biham, E., Dunkelman, O., Keller, N.: Linear cryptanalysis of reduced round serpent. In: Matsui, M. (ed.) FSE 2001. LNCS, vol. 2355, pp. 16–27. Springer, Heidelberg (2002)
4. Biham, E., Dunkelman, O., Keller, N.: The rectangle attack - rectangling the serpent. In: Pfitzmann, B. (ed.) EUROCRYPT 2001. LNCS, vol. 2045, pp. 340–357. Springer, Heidelberg (2001)
5. Biham, E., Dunkelman, O., Keller, N.: New results on boomerang and rectangle attacks. In: Daemen, J., Rijmen, V. (eds.) FSE 2002. LNCS, vol. 2365, pp. 1–16. Springer, Heidelberg (2002)
6. Biham, E., Shamir, A.: Differential cryptanalysis of DES-like cryptosystems. J. Cryptology **4**(1), 3–72 (1991)
7. Bilgin, B., Nikova, S., Nikov, V., Rijmen, V., Stütz, G.: Threshold implementations of all 3 × 3 and 4 × 4 S-boxes. In: Prouff, E., Schaumont, P. (eds.) CHES 2012. LNCS, vol. 7428, pp. 76–91. Springer, Heidelberg (2012)
8. Blondeau, C., Gérard, B., Tillich, J.P.: Accurate estimates of the data complexity and success probability for various cryptanalyses. Des. Codes Crypt. **59**(1–3), 3–34 (2011)
9. Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J.B., Seurin, Y., Vikkelsoe, C.: PRESENT: an ultra-lightweight block cipher. In: Paillier, P., Verbauwhede, I. (eds.) CHES 2007. LNCS, vol. 4727, pp. 450–466. Springer, Heidelberg (2007)
10. Chakraborty, K., Sarkar, S., Maitra, S., Mazumdar, B., Mukhopadhyay, D., Prouff, E.: Redefining the transparency order. Cryptology ePrint Archive, Report 2014/367 (2014)
11. Courtois, N.T., Pieprzyk, J.: Cryptanalysis of block ciphers with overdefined systems of equations. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 267–287. Springer, Heidelberg (2002)
12. Dinur, I., Shamir, A.: Cube attacks on tweakable black box polynomials. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 278–299. Springer, Heidelberg (2009)

13. Dunkelman, O., Indesteege, S., Keller, N.: A differential-linear attack on 12-round serpent. In: Chowdhury, D.R., Rijmen, V., Das, A. (eds.) INDOCRYPT 2008. LNCS, vol. 5365, pp. 308–321. Springer, Heidelberg (2008)
14. Eisenbarth, T., Öztürk, E. (eds.): LightSec 2014. LNCS, vol. 8898. Springer, Heidelberg (2015)
15. Helleseth, T. (ed.): EUROCRYPT 1993. LNCS, vol. 765. Springer, Heidelberg (1994)
16. ISO/IEC 29192–2:2012: Information technology - security techniques - lightweight cryptography - part 2: Block ciphers (2011)
17. Knudsen, L.R.: Truncated and higher order differentials. In: Preneel, B. (ed.) FSE 1994. LNCS, vol. 1008, pp. 196–211. Springer, Heidelberg (1995)
18. Kocher, P.C., Jaffe, J., Jun, B.: Differential power analysis. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 388–397. Springer, Heidelberg (1999)
19. Kohno, T., Kelsey, J., Schneier, B.: Preliminary cryptanalysis of reduced-round Serpent. In: AES Candidate Conference, pp. 195–211 (2000)
20. Makarim, R.H., Tezcan, C.: Relating undisturbed bits to other properties of substitution boxes. In: Eisenbarth and Öztürk [14], pp. 109–125
21. Matsui, M.: Linear cryptoanalysis method for DES cipher. In: Helleseth [15], pp. 386–397
22. McLaughlin, J., Clark, J.A.: Filtered nonlinear cryptanalysis of reduced-round serpent, and the wrong-key randomization hypothesis. In: Stam, M. (ed.) IMACC 2013. LNCS, vol. 8308, pp. 120–140. Springer, Heidelberg (2013)
23. Nguyen, P.H., Wu, H., Wang, H.: Improving the algorithm 2 in multidimensional linear cryptanalysis. In: Parampalli, U., Hawkes, P. (eds.) ACISP 2011. LNCS, vol. 6812, pp. 61–74. Springer, Heidelberg (2011)
24. Nyberg, K.: Differentially uniform mappings for cryptography. In: Helleseth [15], pp. 55–64
25. Prouff, E.: DPA attacks and S-boxes. In: Gilbert, H., Handschuh, H. (eds.) FSE 2005. LNCS, vol. 3557, pp. 424–441. Springer, Heidelberg (2005)
26. Selçuk, A.A.: On probability of success in linear and differential cryptanalysis. J. Cryptology **21**(1), 131–147 (2008)
27. Tezcan, C.: The improbable differential attack: cryptanalysis of reduced round CLEFIA. In: Gong, G., Gupta, K.C. (eds.) INDOCRYPT 2010. LNCS, vol. 6498, pp. 197–209. Springer, Heidelberg (2010)
28. Tezcan, C.: Improbable differential attacks on PRESENT using undisturbed bits. J. Comput. Appl. Math. **259**, Part B(0), 503–511 (2014)
29. Tezcan, C., Özbudak, F.: Differential factors: improved attacks on SERPENT. In: Eisenbarth and Öztürk [14], pp. 69–84
30. Tezcan, C., Taskin, H.K., Demircioglu, M.: Improbable differential attacks on serpent using undisturbed bits. In: Poet, R., Rajarajan, M. (eds.) Proceedings of the 7th International Conference on Security of Information and Networks, p. 145. ACM, New York (2014)
31. Wang, M.: Differential cryptanalysis of reduced-round PRESENT. In: Vaudenay, S. (ed.) AFRICACRYPT 2008. LNCS, vol. 5023, pp. 40–49. Springer, Heidelberg (2008)