An Analysis of Serpent-p and Serpent-p-ns

Orr Dunkelman
Computer Science department, Technion, Haifa 32000, Israel

February 2, 1999

Abstract

Serpent is the AES candidate with the largest safety margins. In this paper we try to analyze how these security margins affect the strength of the cipher. We adopt a simplified Linear Transformation (proposed by the designers of Serpent as their preliminary candidate), and show that even this weakened variant requires at least 2^{121} known plaintext to be attacked. We conclude that Serpent is secure, even with much smaller number of rounds.

1 Introduction

Serpent [1, 2] is one of the more promising AES candidate, due to large safety margins. It was introduced by Eli Biham, Ross J. Anderson and Lars R. Knudsen. The Serpent family has two variants: original published variant Serpent-0 [1], and the AES submission variant Serpent-1 [2]. By convention when we say just Serpent we mean Serpent-1. In the history remarks in [1, 2] a preliminary linear transformation is mentioned, which rotates three of the words in the bitslice implementation, rather than performing a more complex mixing. In this paper we study two variants of Serpent:

1. Serpent with the rotation linear transformation and the S-boxes of Serpent-0 (derived from the S-boxes of DES). We call this variant Serpent-p.

2. Serepnt with this rotation linear transformation and the new S-boxes of Serpent-1. We call this variant Serpent-p-ns.

We study these variants in the hope that their analysis shed light on the design and strength of Serpent.

In this paper we analyze these variants using several cryptanalytic techniques, including differential cryptanalysis, cryptanalysis using impossible differentials, and linear cryptanalysis. It appears that both variants are secure against differential cryptanalysis, and cryptanalysis using impossible differentials. However, we could attack 32 rounds of both variants using linear cryptanalysis, given almost all possible plaintext/ciphertext pairs. We expect that, the relation between Serpent-0 and Serpent-1 is similar the relation between Serpent-p and Serpent-p-ns.

We call a s-box whose input difference is non-zero an *active s-box.* We call a bit which has non-zero difference, *an active bit. Active entry* is an entry in the difference distribution table, which has non-zero value. We denote hexadecimal numbers with subscript x and binary with subscript b (e.g., $61 = 3D_x = 111101_b$). In this work '?' denotes unknown values of bits or nibbles, depending on the base of the number.

This paper is organized as follows: In Section 2 we describe Serpent-p and Serpent-p-ns. In Section 3 we describe our differential attacks on the variants. In Section 4 we gathered several observations on the S-boxes of Serpent-p-ns and Serpent. In Section 5 we describe cryptanalysis using impossible differentials. In Section 6 we describe linear attacks on this variants, and finally in Section 7 we summarize this work.

2 A Description of Serpent-p

We adopt the standard non-bitsliced notations from [1, 2], and take the names of the s-boxes from there.

Serpent-p encrypts 128-bit blocks under keys of 128, 192 and 256 bits. Given a plaintext - P :

$$\begin{aligned} \hat{B}_0 &= IP(P) \\ \hat{B}_{i+1} &= R_i(\hat{B}_i) \\ C &= IP^{-1}(\hat{B}_{64}) \\ R_i(X) &= \mathrm{Rot}_i(\hat{\mathcal{S}}_i(X \oplus \hat{K}_i)) \qquad i = 0, \ldots, 62 \\ R_i(X) &= \hat{\mathcal{S}}_i(X \oplus \hat{K}_i) \oplus \hat{K}_{64} \quad i = 63 \end{aligned}$$

After each of the 64 rounds, we use Rot_i, which is a set of rotations defined as (0,1,3,7) for even i's and (0,5,13,22) for odd i's. This means that for even i's the first bit in each nibble is rotated to the left by 0 nibbles, the second by 1 (e.g., from nibble 3 to nibble 4) , etc. The s-boxes are 4-bit to 4-bit permutations. In each round the same s-box is used 32 times, while different s-boxes are used in the various rounds. The s-boxes are derived from the S-boxes of DES [10], giving 32 s-boxes in total.

Serpent-p uses the key schedule of Serpent:

1. Expand the key length to 256 bits by appending one bit '1' and as many '0' as required.
2. Define $w_{-8}, \ldots, w_{-1}$ to be the eight 32-bit words composed from those 256 bits.
3. For $i = 0, \ldots, 64$ define $w_i := (w_{i-8} \oplus w_{i-5} \oplus w_{i-3} \oplus w_{i-1} \oplus \phi \oplus i) <<< 11$,
   where ϕ is the golden ratio - $9E3779B9_x$.
4. For $i = 0, \ldots, 64$ define $K_i = (k_{4i}, k_{4i+1}, k_{4i+2}, k_{4i+3}) = S_{n(i)}(w_{4i}, w_{4i+1}, w_{4i+2}, w_{4i+3})$, where $n(i) = (3 - i) \bmod 32$.

We define Serpent-p-ns as a variant of Serpent-p with the following modifications:

1. Serpent-p-ns uses the same s-boxes as Serpent-1.
2. Serpent-p-ns has 32 rounds. Therefore in the encryption process i is between 0 to 31, and in the key schedule algorithm i is between 0 to 32. Also for Serpent-p-ns $n(i) = (3 - i) \bmod 8$.

Therefore, Serpent-p-ns differs from Serpent-1 only in the linear transformation. We expect that the relation between Serpent-p and Serpent-p-ns is very similar to the relation between Serpent-0 and Serpent, as the differences between Serpent-0 and Serpent are composed from these two differences together with another difference in the key scheduling.

3 Differential Attacks

Looking at the rotations set, we observe that in both variants Serpent-p and Serpent-p-ns;

1. Each characteristic has 31 counterparts. This observation is derived from the fact that the 32 parallel s-boxes in each round are the same. Therefore, if we rotate both the input difference and the output difference of a characteristic by the same multiple of a nibble, the resulting differences form another characteristic with the same probability.
2. Due to the previous observation, we can redefine the notation of iterative characteristics to be differential characteristics which have the same input and output differences or whose output difference value is the input difference rotated by a multiple of a nibble.

We searched for iterative characteristics of Serpent-p and of Serpent-p-ns. In the following we represent characteristics as lists of the active s-boxes in each round, denoting the *path* of the development of the differences during encryption. For example, the differential characteristic $1_x \rightarrow C_x$ in the first round, is represented as $(0) \rightarrow (3,7)$, where 0 denotes that S-box 0 is active in the first round, and (3,7) denotes that the output of those S-box cause differences in the inputs of S-boxes 3 and 7 in the next round. In the search phase we ignored the actual probability of the characteristic. We assume that almost all the entries in the difference distribution tables are active, i.e., all entries have non-zero value except of those entries with one bit difference in the input and one bit difference in the output, which are always impossible. We later verify if the resultant characteristics hold in the variants we study.

We distinguish between three kinds of paths, based on whether they exploit the various rotations in the two sets. We first describe three paths which exploit some properties of the first rotation set (0,1,3,7), to which we call O1, O2 and O3.

Paths O1 and O2 yield the following iterative characteristics:

Path	First Round		Second Round		The Path
	Active s-boxes	Characteristics	Active s-boxes	Characteristics	
O1:	0	$I \rightarrow 6_x$	1	$3_x \rightarrow J$	
	1	$I \rightarrow 3_x$	2	$3_x \rightarrow J$	$(0,1,2) \rightarrow (1,2,3) \rightarrow (1+x, 2+x, 3+x)$
	2	$I \rightarrow 3_x$	3	$6_x \rightarrow J$	for any $x \in \{0,5,13,22\}$
O2:	0	$I \rightarrow C_x$	3	$6_x \rightarrow J$	
	2	$I \rightarrow 6_x$	5	$6_x \rightarrow J$	$(0,2,4) \rightarrow (3,5,7) \rightarrow (3+x, 5+x, 7+x)$
	4	$I \rightarrow 6_x$	7	$C_x \rightarrow J$	for any $x \in \{0,5,13,22\}$

When path O3 is

$$(0,1,4) \rightarrow (1,4,7) \rightarrow (1+x, 4+x, 7+x) \rightarrow (4+x, 7+x, 8+x) \rightarrow$$

$$(4+x+y, 7+x+y, 8+x+y) \rightarrow (7+x+y, 8+x+y, 11+x+y) \rightarrow$$

$$(7+x+y+z, 8+x+y+z, 11+x+y+z) \qquad (\text{Where } x,y,z \in \{0,5,13,22\})$$

Path O3 yields the iterative characteristics:

Round	First active s-box	Characteristicss	Second active s-box	Characteristicss	Third active s-box	Characteristicss
1	0	$I \rightarrow A_x$	1	$I \rightarrow 5_x$	4	$I \rightarrow 5_x$
2	1	$3_x \rightarrow J$	4	$5_x \rightarrow J$	7	$C_x \rightarrow J$
3	$1+x$	$J \rightarrow C_x$	$4+x$	$J \rightarrow 5_x$	$7+x$	$J \rightarrow 3_x$
4	$4+x$	$5_x \rightarrow K$	$7+x$	$5_x \rightarrow K$	$8+x$	$A_x \rightarrow K$
5	$4+x+y$	$K \rightarrow C_x$	$7+x+y$	$K \rightarrow 3_x$	$8+x+y$	$K \rightarrow 5_x$
6	$7+x+y$	$5_x \rightarrow L$	$8+x+y$	$3_x \rightarrow L$	$11+x+y$	$C_x \rightarrow L$

Where I, J, K and L represent one bit difference ($1_x, 2_x, 4_x$ or 8_x), and J determine the factor x in the rotation (K determine y, and L determine z). J determaine x in the following manner: when $J = 1_x$ $x = 0$, and when $J = 2_x$ $x = 5$, etc. Therefore, we might have 288 iterative characteristics.

We now check whether the required entries in the S-boxes' difference distribution tables, are active, and what their probabilities are. Table 10 summarizes the probabilities.

Consulting Table 10, we can see that O1 and O2 are not practical on Serpent-p, due to the fact that they have only one round at most. In O3, in S_{15}, for J= 2_x, we can connect between the first two rounds and the next two, but in S_{17} the fourth round characteristic exist for K=1_x, while in S_{18} we need K=4_x. As a result we have a 4-round characteristic with probability of $2^{-30.2}$. We also find the path which starts at S_{15} (fourth-round characteristic) and continues till S_{21}. This 7-round characteristic has the probability of $2^{-57.4}$. Of course, the same characteristics appear 32 rounds afterward (under the same S-boxes).

We found one path which exploits some properties of the second rotation set (0,5,13,22), to which we call T. Path T yields the following iterative characteristics:

Path	First Round		Second Round		The Path
	Active s-boxes	Characte-ristics	Active s-boxes	Characte-ristics	
T:	0	$9_x \to J$	0	$I \to 3_x$	
	5	$3_x \to J$	5	$I \to 3_x$	$(0,5,10) \to (0,5,10) \to (0+x, 5+x, 10+x)$
	10	$3_x \to J$	10	$I \to 9_x$	for any $x \in \{0,1,3,7\}$

where I and J represent one bit difference $(1_x, 2_x, 4_x, 8_x)$, and J determine the factor x in the rotation.

We now check whether the required entries in the s-boxes' difference distribution table, are active, and what their probabilities are. In appendix B, in Table 10 we put the summary of the probabilities.

Looking at the required input differences for continuing the iterative characteristic, we find that for path T, we can not create a characteristic with more than 2 rounds. It may look like we can, but due to the need of different active bits, a two-round characteristic is the longest we reached. However, we can find a 4-round characteristic using O3. It starts in s-box 16 ($I = 8_x$), and using the differences $8_x, 2_x, 2_x, 1_x, 1_x$ (e.g., the input in the first round is 8_x which become 2_x later, etc.).

The mixed paths, i.e., those that exploit both sets of rotations, described after even rounds (first rotation set is (0,1,3,7)): and yield the following iterative characteristics:

Due to the fact that 6_x difference in the input cannot become one-bit difference in the output paths A,D,F,I and J, can have no more than one-round characteristic.

As shown in appendix B in Table 11, it is easy to see that paths C,E,G,H and K have no more than one-round characteristic. Path B, has 2-round characteristic with the probability of $2^{-16.4}$.

We now take the longest characteristic (4-round) and try to add rounds. At the beginning we can add 4 rounds with probability of $2^{-43.8}$. While in the end, we are able to add 3 rounds in the end with the probability of $2^{-30.4}$, giving us a 11-round characteristic (described in Table 2) with a probability of $2^{-104.6}$. This can be used to attack differentially 13 rounds of Serpent-p from round 11 to round 23, using the following algorithm:

1. Given the input difference, choose the actual value in the beginning of the one before last round of the attack (last round of characteristic) (in 11 s-boxes).

2. For each of the 16 bits, try all the possibly values in the first round output (in 11 s-boxes - S_3,S_6,S_{11},S_{13},S_{15},S_{17},S_{19},S_{20},S_{21},S_{23},S_{27}).

3. Obtaining a structure in the size of 2^{44} plaintexts, which give us 2^{87} pairs which become into the input difference of the differential.

4. Take 2^{20} such structures (totally giving 2^{64} plaintext) structure and decrypt them.

5. Discard all pairs which differ in the 6 s-boxes (S_5,S_9,S_{15},S_{20},S_{24},S_{29})in the beginning which suppose to have zero difference. Doing so, out of 2^{107} pairs, only 2^{83} pairs remain. Due to the fact that only 0.4 of the entries has non-zero value, given the output difference of the first round (the beginning of the characteristic), in the other 27 s-boxes, only 0.4^{27} of them might become the needed difference. This means that only one out of $2^{35.7}$ suggestions survives. Therefore, we are left with $2^{48.7}$ pairs.

6. About $2^{38.7}$ of those pairs have the same last round subkey. Of which about $2^{2.4}$ are right pairs.

7. If we take the subkey having about 4 pairs suggesting it, we found the first round subkey (44 bits) and 44 bits of the one before last round subkey.

8. Applying this attack with the offset of 1 nibbles (to the left) shall require 2 times the data, and will give the last two rounds subkeys (besides 44-bits).

Path	First Round		Second Round		Path
	Active s-boxes	Characteristics	Active s-boxes	Characteristics	
A:	0	$1_x \rightarrow 6_x$	1	$3_x \rightarrow 2_x$	
	1	$1_x \rightarrow 5_x$	3	$5_x \rightarrow 1_x$	$(0,1,3) \rightarrow (1,3,4) \rightarrow (3,4,6)$
	3	$2_x \rightarrow 3_x$	4	$6_x \rightarrow 1_x$	
B:	0	$1_x \rightarrow A_x$	1	$3_x \rightarrow 4_x$	
	1	$1_x \rightarrow 9_x$	7	$9_x \rightarrow 1_x$	$(0,1,7) \rightarrow (1,7,8) \rightarrow (7,8,14)$
	7	$4_x \rightarrow 3_x$	8	$A_x \rightarrow 1_x$	
C:	0	$8_x \rightarrow C_x$	3	$5_x \rightarrow 1_x$	
	3	$1_x \rightarrow 9_x$	7	$9_x \rightarrow 1_x$	$(0,3,7) \rightarrow (3,7,10) \rightarrow (0,3,7)$
	7	$1_x \rightarrow 5_x$	10	$C_x \rightarrow 8_x$	
D:	0	$8_x \rightarrow C_x$	3	$6_x \rightarrow 1_x$	
	2	$8_x \rightarrow A_x$	7	$A_x \rightarrow 8_x$	$(0,2,6) \rightarrow (3,7,9) \rightarrow (3,29,31)$
	6	$1_x \rightarrow 6_x$	9	$C_x \rightarrow 8_x$	
E:	0	$1_x \rightarrow C_x$	3	$A_x \rightarrow 2_x$	
	3	$1_x \rightarrow 3_x$	4	$3_x \rightarrow 1_x$	$(0,3,4) \rightarrow (3,4,7) \rightarrow (4,7,8)$
	4	$2_x \rightarrow A_x$	7	$C_x \rightarrow 1_x$	
F:	0	$3_x \rightarrow 4_x$	3	$4_x \rightarrow 6_x$	
	8	$5_x \rightarrow 1_x$	8	$1_x \rightarrow 5_x$	$(0,8,13) \rightarrow (3,8,16) \rightarrow (8,16,21)$
	13	$6_x \rightarrow 4_x$	16	$4_x \rightarrow 3_x$	
G:	0	$A_x \rightarrow 1_x$	0	$1_x \rightarrow A_x$	
	5	$9_x \rightarrow 1_x$	5	$1_x \rightarrow 9_x$	$(0,5,15) \rightarrow (0,5,22) \rightarrow (5,22,27)$
	15	$3_x \rightarrow 8_x$	22	$8_x \rightarrow 3_x$	
H:	0	$C_x \rightarrow 1_x$	0	$1_x \rightarrow C_x$	
	10	$5_x \rightarrow 4_x$	13	$4_x \rightarrow 9_x$	$(0,10,19) \rightarrow (0,13,22) \rightarrow (3,13,22)$
	19	$9_x \rightarrow 4_x$	22	$4_x \rightarrow 5_x$	
I:	0	$C_x \rightarrow 8_x$	7	$8_x \rightarrow C_x$	
	15	$6_x \rightarrow 1_x$	15	$1_x \rightarrow A_x$	$(0,15,24) \rightarrow (7,15,24) \rightarrow (5,20,29)$
	24	$A_x \rightarrow 1_x$	24	$1_x \rightarrow 6_x$	
J:	0	$6_x \rightarrow 2_x$	1	$2_x \rightarrow C_x$	
	9	$A_x \rightarrow 1_x$	9	$1_x \rightarrow A_x$	$(0,9,17) \rightarrow (1,9,18) \rightarrow (14,23,31)$
	17	$C_x \rightarrow 2_x$	18	$2_x \rightarrow 6_x$	
K:	0	$9_x \rightarrow 4_x$	3	$4_x \rightarrow 5_x$	
	13	$C_x \rightarrow 1_x$	13	$1_x \rightarrow C_x$	$(0,13,23) \rightarrow (3,13,26) \rightarrow (3,16,26)$
	23	$5_x \rightarrow 4_x$	26	$4_x \rightarrow 9_x$	

Table 1: Paths in Serpent-p and Serpent-p-ns

Round	S-box/Difference
12	$0/3_x$, $3/8_x$, $6/1_x$, $7/C_x$, $11/3_x$, $12/B_x$
12	$13/C_x$, $17/B_x$, $18/C_x$, $27/3_x$
13	$3/1_x$, $6/C_x$, $7/B_x$, $13/3_x$, $18/B_x$, $30/4_x$
14	$3/B_x$, $7/1_x$, $8/A_x$, $11/6_x$
15	$10/C_x$, $11/5_x$, $14/C_x$
16	$0/8_x$, $1/8_x$, $4/8_x$
17	$1/3_x$, $4/5_x$, $7/C_x$
18	$6/2_x$, $9/2_x$, $12/2_x$
19	$9/5_x$, $12/5_x$, $13/A_x$
20	$9/1_x$, $12/1_x$, $13/1_x$
21	$9/1_x$, $12/5_x$, $13/3_x$, $15/4_x$, $16/C_x$
22	$3/8_x$, $10/1_x$, $12/1_x$, $15/3_x$, $16/1_x$, $20/2_x$
End	$3/1_x$, $6/4_x$, $11/2_x$, $13/2_x$, $15/1_x$, $17/A_x$
End	$19/8_x$, $20/1_x$, $21/2_x$, $23/8_x$, $27/8_x$

Table 2: 11-round differential characteristic of Serpent-p (the input for the round is given)

9. For each of the possibilities for those bits - we can calculate the real key.

We take structures, which differ at the 11 s-boxes. Therefore, each structure should give us 2^{87} pairs. The needed work load is 2^{64} decryptions. This will recover 128-, 192- and 256-bit keys.

It worth to note that Serpent-p might have the following possible weaknesses:

1. In the difference distribution table of Serpent-p, there is one entry with the value 10 (out of 16) in S_{29}, and all but S_{19} have entries with 6's and 8's.
2. There are many structures in the difference distribution tables. For example, in S0, the input difference 8_x always becomes into the form of $1???_b$. In S_5, the input difference 7_x *always* becomes as output difference of the form $???0_b$. The same is in S_6, when $A_x \to 0???_x$ with probability of 1.
3. Due to the high value entries, there are lots of zero entries. This might help when cryptanalyzing using the impossible differential method.
4. We found that one-bit difference affects at most 4 bits in the next round, 16 in the second round and 64 in the third round. However the 64 bits enter only 30 out of the 32 s-boxes.
5. The above property gives us 4-round impossible differential, when the input difference is in the first nibble, the 72nd bit and the 124th bit can not be active after 4 rounds.

As for Serpent-p-ns, we gathered the results about Serpent-p-ns are given in appendix C, in Table 12.

We took T's longest characteristic (7-round) and add as much rounds as we could. We can add 2-round characteristic in the beginning, with the probability of 2^{-18}. We also can add 3-round in the end with the probability of 2^{-25}, totally giving 2^{-104} probability for the 12-round characteristic described in Table 2. Using this characteristic, we can analyze 12 rounds of Serpent-p-ns, using the above algorithm, with minor modifications which are:

- 10 s-boxes have non-zero difference instead of 11, therefore we use structures of 2^{40} ciphertexts. Each structure gives 2^{79} pairs, and we need 2^{27} such in order to get 2^{106} pairs.
- We check if 12 s-boxes in the first round have zero difference, and in the other 20 we have the chance of $1/0.4^{20}$ to be candidate for right pair. Totally leaving one pair out of 2^{74}.
- For each subkey, we can identify in the last round 11 s-boxes which are active, meaning that out of the 2^{106} pairs, only 2^{32} suppose to remain, from which only one subkey value is about to be suggested by 4 pairs. We find therefore also 20 subkey bits of the one before last round.

Round	S-box/Difference
1	$3/9_x$, $8/E_x$, $13/E_x$, $25/A_x$, $30/A_x$
2	$3/B_x$, $25/9_x$, $30/B_x$
3	$0/8_x$, $5/8_x$, $10/8_x$
4	$0/9_x$, $5/3_x$, $10/3_x$
5	$0/1_x$, $5/1_x$, $10/1_x$
6	$0/9_x$, $5/3_x$, $10/3_x$
7	$0/1_x$, $5/1_x$, $10/1_x$
8	$0/9_x$, $5/3_x$, $10/3_x$
9	$0/1_x$, $5/1_x$, $10/1_x$
10	$0/3_x$, $5/3_x$, $10/9_x$
11	$0/1_x$, $5/1_x$, $17/8_x$
12	$0/1_x$, $5/3_x$, $18/4_x$, $22/A_x$, $30/4_x$
END	$0/1_x$, $1/2_x$, $3/4_x$, $5/1_x$, $18/1_x$, $19/2_x$
END	$22/1_x$, $23/4_x$, $30/1_x$, $31/2_x$

Table 3: 12-round differential characteristic of Serpent-p-ns

- Doing so another 3 times, for the shift of 1,4,10 nibbles to the left will give all the last round subkey and all but 56 key-bits of the round before last.

The required number of chosen plaintexts is $2^{40} \cdot 2^{26} \cdot 4 = 2^{68}$.

We also tried to use only one s-box (in order to reduce hardware/software complexity) and found out that there are s-box in whose difference distribution table, we can find iterative characteristic. For example, Serpent-p-ns S_3 has for path O1, the probability of 2^{-6} and 2^{-9} per round (depend on its number). This will allow us to create a 17-round differential characteristic with the probability of 2^{-126}.

4 Observations on Serpent's S-boxes

Serpent uses the same s-boxes as Serpent-p-ns. The only difference between the two is the linear transformation (LT) of Serpent that replaces the rotations. Therefore, any problem in Serpent's s-boxes might be also used to analyze Serpent-p-ns, and vice versa.

There are several observations that could be used while attacking Serpent (and Serpent-p-ns):

1. In three out of 8 of Serpent's s-boxes, the input difference 4_x, can become the output difference with the probability of 1/4.

2. There are many places where given an active bit, and arbitrary bit, the output is the same under the same probability, such in S_2 where 4_x and C_x, has probability of 1/4 to become D_x. This can be handy while trying to use truncated differential attack [8].

3. Another case where arbitrary bit lead to the same output is in S_3 where both 2_x and 3_x can become F_x with probability 1/4.

4. In S0 the difference 6_x has probability 1(!) to become $0???_b$. This property means means that the first bit is *always* zero. Such thing can allow an attacker to find a differential and keep some bit at zero difference (reducing the number of unknown bits in the next rounds).

5. It is easy to find other input differences which have an output difference of the form $1???_b$, or any other bit set.

6. There are many cases where a connection between the output differences is known. For example in S0 : $4_x \rightarrow 1?ab_b$, where $a \oplus b = 1$.

We now bring the difference distribution table statistics of Serpent's s-boxes in Table 4.

	Number of			
S-box	0's	2's	4's	16's
0	159	72	24	1
1	157	76	22	1
2	160	70	25	1
3	153	84	18	1
4	153	84	18	1
5	153	84	18	1
6	159	72	24	1
7	153	84	18	1

Table 4: Difference Distribution Table Statistics of Serpent's (and Serpent-p-ns') S-boxes

5 Attacks using Impossible Differential

The impossible differential method [5, 7] discards wrong subkey values, using differentials with probability 0. If a key suggests such differential, the subkey is wrong and can be discarded. Serpent-p and Serpent-p-ns, have 4-round impossible differentials.

If the first rotation used is (0,1,3,7) we find a 4-round impossible differential. The input difference 00...00k_x (where $k \in \{1_x, \ldots, F_x\}$) can never cause after 4 rounds a difference in which bits 72 and 124 are active. Using the (0,5,13,22) rotation first, we find an impossible differential in which the input difference 00..00k_x and after 4 rounds the output difference has bits 36,96 and 124 active.

An 8-round Serpent-p (or Serpent-p-ns) can be attacked using this method, as follows:

We use the impossible differential in rounds 2-5. In the first 2 rounds we check 5 s-boxes (the one which affects the first bit from round 1, and four in round 0 that activate the s-box from round 1). We also check the s-box which bit 124 enters in round 6, and the 4 s-boxes that the 124th bit affects later in round 7. We, therefore, check 10 s-boxes, which has 40 subkey bits.

If we take sufficiently many pairs, we can recover the key using the following algorithm:

1. For each subkey of the second round, we check all the pairs which end with the output difference 8_x. We have 8 such pairs, and using 16 table-lookups we can find all of them.
2. For each possible pair, we can find the 8 possible inputs in each of the 4 s-boxes we check.
3. After obtaining a pairs ciphertexts, we try all possible key values in the 5 s-box in rounds 6,7. Decrypting a pair under those subkeys, will help us discard all the subkeys that suggest that the 124th bit is different.

Each of the pairs which was found in the above algorithm, has the probability of 2^{-21} to discard a subkey. Meaning that in average 2^{21} pairs will discard a fraction of $1 - 1/e$ out of the subkeys.

The complexity to the above algorithm is given by the expression $2^4 \cdot 2^4 \cdot 2^{13} \cdot 2^{20} \cdot k$ when the first 2^4 is for 16 possible subkeys, times 16 table look-ups, which result with 2^{13} possible pairs for the first round, times 2^{20} trying all the last two rounds subkeys, times k, the times we do so. Afterward we perform exhaustive search on all the remaining possibilities, which requires $2^{128}/e^k$ steps. For 8 rounds, the best time complexity is given for $k \approx 56$, which is $2^{47.21}$. It requires about $2^{26.81}$ chosen plaintexts. We obtain 40 subkey bits. Now we can extend the attack for other subkeys. We can now get subkeys of another 8 s-boxes with time complexity of $2^{39.21}$, with additional $2^{18.81}$ chosen plaintexts. An unoptimized attack which performs this 32 times requires $2^{44.21}$ steps and $2^{26.81}$ texts.

With the same algorithm (extended one round in the end), we can attack both Serpent-p and Serpent-p-ns, with $2^{24.74}$ texts and $2^{109.67}$ steps.

Round	S-box	Known bits	Round	S-box	Known bits	Round	S-box	Known
6	0	F_x	16	14	1_x	26	5	1_x
7	1	2_x	17	15	2_x	27	8	4_x
8	1	1_x	18	15	1_x	28	21	4_x
9	8	8_x	19	15	1_x	29	24	4_x
10	30	8_x	20	5	8_x	30	14	4_x
11	31	2_x	21	8	4_x	31	21	8_x
12	31	1_x	22	13	2_x	32	21	1_x
13	0	2_x	23	14	2_x	33	22	2_x
14	13	4_x	24	4	8_x	34	22	1_x
15	14	2_x	25	5	8_x	35	22	1_x
End	3	4_x		12	8_x		22	1_x

Table 5: Linear approximation of Serpent-p

6 Linear Attack

We found a 30-round linear approximation of Serpent-p, with the probability of $1/2 - 2^{-56.41}$. The characteristics is from the round number 6 (using S_6) till round 35 (using S_3). It is based on one-bit to one-bit relations over 28 rounds. We describe it using Matsui's [9] notation as : $\{0, 1, 2, 3\} \rightarrow \{2, 39, 76\}$.

The approximation (given in the format of S-boxes/known input bits) is in Table 5

We attack using the following algorithm:

1. Initialize $32 \cdot 2^{40}$ counters for 40 key bits (s-box 0 in round 6, the four which enter in round 5, s-box 19 in round 35, and the four it outputs to 0,9,19,20 in round 36). We do so 32 times in order to find all the first and second round subkeys at once due to property 1 mentioned before.

2. We calculate for each input and output the corresponding subkey values (meaning given the input we can find the subkey which it suggest).

3. For any plaintext/ciphertext which is arriving - update the 32 counters which correspond to its value.

4. Do so until there is a high chance that one counter is appears to be outnumbering the rest (after about 2^{116} known plaintexts). This is probably the right subkey (40 subkey-bits). Now discard counters which not correspond for the right value and continue, till you got the whole first and second round subkeys.

Incrementing 32 counters is about the same work as one round of Serpent-p and thus the counting procedure does not affect the time complexity of 2^{116} using 2^{116} known plaintexts.

For Serpent-p-ns, we used the same technique, and found the next two characteristics with the probability of $1/2 + 2^{-59}$. Both are from round 1 to round 30. The first characteristic is: $\{0, 1, 3\} \rightarrow \{2, 116, 121\}$. and the second characteristic is: $\{0, 2, 3\} \rightarrow \{2, 116, 121\}$. Running almost the same algorithm as above (now using 2 counters for each 40 sub-keys which improve the identification), we can find the whole key using 2^{121} known plaintext-ciphertext. The approximations (given in the format of S-boxes/known input bits) is in Table 6

7 Summary

In Table 7 we gathered the best cryptanalitic results. It Appears that Serpent-p and Serpent-p-ns are secure against the attacks involving differentials, such differential attack and cryptanalysis using

Round	S-box	Known bits	Round	S-box bits	Known	Round bits	S-box	Known
1	0	B_x/D_x	11	7	2_x	21	14	2_x
2	5	2_x	12	12	2_x	22	27	4_x
3	6	2_x	13	13	2_x	23	30	4_x
4	11	2_x	14	26	4_x	24	3	2_x
5	12	2_x	15	29	4_x	25	3	1_x
6	25	4_x	16	2	2_x	26	8	2_x
7	28	4_x	17	2	1_x	27	9	2_x
8	1	2_x	18	7	2_x	28	14	2_x
9	1	1_x	19	8	2_x	29	15	2_x
10	6	2_x	20	13	2_x	30	28	4_x
End	28	1_x		29	2_x		31	4_x

Table 6: Linear approximation of Serpent-p-ns

Type of Attack	Serpent-p Rounds	Serpent-p Number of Texts	Serpent-p Time Complexity	Serpent-p-ns Rounds	Serpent-p-ns Number of Texts	Serpent-p-ns Time Complexity
Differential	11-23	2^{64}	2^{64}	0-13	2^{68}	2^{68}
Imp. Diff.	0-7	$2^{26.81}$	$2^{47.21}$	0-7	$2^{26.81}$	$2^{47.21}$
Imp. Diff.	0-8	$2^{24.74}$	$2^{109.67}$	0-8	$2^{24.74}$	$2^{109.67}$
Linear	5-36	2^{116}	2^{116}	0-31	2^{121}	2^{121}

Table 7: Attacks on Serpent-p's family

impossible differentials. However, we found that linear attack do work, though the requirement for almost all the possible plaintext/ciphertext pairs.

In conclusion, it appears that Serpent is secure, and 16 rounds of Serpent might be secure more than Triple-DES, as required from AES.

8 Acknowledgement

I would like to thank Eli Biham for his great assistance during my work on this paper.

References

[1] Ross J. Anderson, Eli Biham, Lars R. Knudsen, *Serpent: A Proposal for the Advanced Encryption Standard*. Available at: `http://www.cs.technion.ac.il/~biham/Reports/Serpent`.

[2] Eli Biham, Ross J. Anderson, Lars R. Knudsen, *Serpent: A New Block Cipher Proposal*, Proceedings of Fast Software Encryption - FSE '98, Springer LNCS Vol. 1372 pp. 222–238.

[3] Eli Biham, Alex Biryukov, *An Improvement of Davies' Attack on DES*, Proceedings of Eurocrypt '94, Springer LNCS 950, Journal of Cryptology, vol. 10, No. 3, pp.195-206,1997. Available at `http://cs.technion.ac.il/~biham/publications.html`.

[4] Eli Biham, Adi Shamir, *Differential Cryptanalysis of the Data Encryption Standard* (Springer 1993).

[5] Eli Biham, Alex Biryukov, Adi Shamir, *Cryptanalysis of Skipjack Reduced to 31 Rounds using Impossible Differentials*,Technion Computer Science Department technical report CS0947,1998. available at: `http://www.cs.technion.ac.il/~biham/Reports/Skipjack`.

S_0:	3, 8, 15, 1, 10, 6, 5, 11, 14, 13, 4, 2, 7, 0, 9, 12
S_1:	15, 12, 2, 7, 9, 0, 5, 10 ,1, 11, 14, 8, 6, 13, 3, 4
S_2:	8, 6, 7, 9, 3, 12, 10, 15, 13, 1, 14, 4, 0, 11, 5, 2
S_3:	0, 15, 11, 8, 12, 9, 6, 3, 13, 1, 2, 4, 10, 7, 5, 14
S_4:	1, 15, 8, 3, 12, 0, 11, 6, 2, 5, 4, 10, 9, 14, 7, 13
S_5:	15, 5, 2, 11, 4, 10, 9, 12, 0, 3, 14, 8, 13, 6, 7, 1
S_6:	7, 2, 12, 5, 8, 4, 6, 11, 14, 9, 1, 15, 13, 3, 10, 0
S_7:	1, 13, 15, 0, 14, 8, 2, 11, 7, 4, 12, 10, 9, 3, 5, 6

Table 8: Serpent-p S-boxes

[6] Eli Biham, Alex Biryukov, Orr Dunkelman, Eran Richardson, Adi Shamir, *Initial Observations on Skipjack: Cryptanalysis of SkipJack-3XOR*, Proceedings of SAC '98, pp. 367-380, 1998. Available at: `http://www.cs.technion.ac.il/~biham/Reports/Skipjack`.

[7] Lars R. Knudsen, *DEAL - A 128-bit Block Cipher*, University of Bergen, Department of Informatics, Technical report 151, 1998. available at: `http://www.ii.uib.no/~larsr/newblock.html`.

[8] Lars R. Knudsen, *Truncated and Higher Order Differentials*, Proceedings of Fast Software Encryption-Second International Workshop, Springer LNCS Vol. 1008, pp. 196–211.

[9] M. Matsui, *Linear Cryptanalysis Method for DES Cipher"*, Advances in Crtyptology – Eurocrypt '93, Springer LNCS Vol. 765 pp. 386-397.

[10] National Bureau of Standards. *Data encryption standard.* Federal Information Processing Standard (FIPS), Publication 46, National Bureau of Standards, U.S. Department of Commerce, Washington D.C., January 1977. available at: `http://www.csrc.nist.gov/fips/`.

A The S-boxes of Serpent-p and of Serpent-p-ns

The S-boxes of Serpent-p-ns (and of Serpent-1), are given in Table 8, and the S-boxes of Serpent-p are given in Table 9.

B Probabilities Tables of Serpent-p

We now describe how to read the following tables: If one wants to know what is the possibility of a round in any of the characteristic, in the table you have the bound. One has to check in the difference distribution tables, whether the characteristics holds for the wanted input/output (i.e., what is the value of the unknown bit). Also one should keep in mind that paths O1 and O2 come in pairs of even round - odd round and thus need no connection, while trying to connect between odd rounds to the one after them , one should check whether there is a bit which is the output of the odd round and the input of the even round. In O3, one should keep in mind that odd rounds are connected to the even rounds in the line under (like the bolded values). In T, the rules of odd and even are replaced, and the rest of the paths has no connection whatsoever to arbitrary bits.

C Probabilities Tables of Serpent-p-ns

We gathered the results from Serpent-p-ns in Table 12.

S_0:	14, 4, 13, 1, 2, 15, 11, 8, 3, 10, 6, 12, 5, 9, 0, 7
S_1:	0, 15, 7, 4, 14, 2, 13, 1, 10, 6, 12, 11, 9, 5, 3, 8
S_2:	4, 1, 14, 8, 13, 6, 2, 11, 15, 12, 9, 7, 3, 10, 5, 0
S_3:	15, 12, 8, 2, 4, 9, 1, 7, 5, 11, 3, 14, 10, 0, 6, 13
S_4:	15, 1, 8, 14, 6, 11, 3, 4, 9, 7, 2, 13, 12, 0, 5, 10
S_5:	3, 13, 4, 7, 15, 2, 8, 14, 12, 0, 1, 10, 6, 9, 11, 5
S_6:	0, 14, 7, 11, 10, 4, 13, 1, 5, 8, 12, 6, 9, 3, 2, 15
S_7:	13, 8, 10, 1, 3, 15, 4, 2, 11, 6, 7, 12, 0, 5, 14, 9
S_8:	10, 0, 9, 14, 6, 3, 15, 5, 1, 13, 12, 7, 11, 4, 2, 8
S_9:	13, 7, 0, 9, 3, 4, 6, 10, 2, 8, 5, 14, 12, 11, 15, 1
S_{10}:	13, 6, 4, 9, 8, 15, 3, 0, 11, 1, 2, 12, 5, 10, 14, 7
S_{11}:	1, 10, 13, 0, 6, 9, 8, 7, 4, 15, 14, 3, 11, 5, 2, 12
S_{12}:	7, 13, 14, 3, 0, 6, 9, 10, 1, 2, 8, 5, 11, 12, 4, 15
S_{13}	:13, 8, 11, 5, 6, 15, 0, 3, 4, 7, 2, 12, 1, 10, 14, 9
S_{14}:	10, 6, 9, 0, 12, 11, 7, 13, 15, 1, 3, 14, 5, 2, 8, 4
S_{15}:	3, 15, 0, 6, 10, 1, 13, 8, 9, 4, 5, 11, 12, 7, 2, 14
S_{16}:	2, 12, 4, 1, 7, 10, 11, 6, 8, 5, 3, 15, 13, 0, 14, 9
S_{17}:	14, 11, 2, 12, 4, 7, 13, 1, 5, 0, 15, 10, 3, 9, 8, 6
S_{18}:	4, 2, 1, 11, 10, 13, 7, 8, 15, 9, 12, 5, 6, 3, 0, 14
S_{19}:	11, 8, 12, 7, 1, 14, 2, 13, 6, 15, 0, 9, 10, 4, 5, 3
S_{20}:	12, 1, 10, 15, 9, 2, 6, 8, 0, 13, 3, 4, 14, 7, 5, 11
S_{21}:	10, 15, 4, 2, 7, 12, 9, 5, 6, 1, 13, 14, 0, 11, 3, 8
S_{22}:	9, 14, 15, 5, 2, 8, 12, 3, 7, 0, 4, 10, 1, 13, 11, 6
S_{23}:	4, 3, 2, 12, 9, 5, 15, 10, 11, 14, 1, 7, 6, 0, 8, 13
S_{24}:	4, 11, 2, 14, 15, 0, 8, 13, 3, 12, 9, 7, 5, 10, 6, 1
S_{25}:	13, 0, 11, 7, 4, 9, 1, 10, 14, 3, 5, 12, 2, 15, 8, 6
S_{26}:	1, 4, 11, 13, 12, 3, 7, 14, 10, 15, 6, 8, 0, 5, 9, 2
S_{27}:	6, 11, 13, 8, 1, 4, 10, 7, 9, 5, 0, 15, 14, 2, 3, 12
S_{28}:	13, 2, 8, 4, 6, 15, 11, 1, 10, 9, 3, 14, 5, 0, 12, 7
S_{29}:	1, 15, 13, 8, 10, 3, 7, 4, 12, 5, 6, 11, 0, 14, 9, 2
S_{30}:	7, 11, 4, 1, 9, 12, 14, 2, 0, 6, 10 ,13, 15, 3, 5, 8
S_{31}:	2, 1, 14, 7, 4, 10, 8, 13, 15, 12, 9, 0, 3, 5, 6, 11

Table 9: Serpent-p S-boxes

Path	S-box number										
	0	1	2	3	4	5	6	7	8	9	10
O1	-5.8	-	-9	-	-5.8	-	-9	-	-9	-	-
O2	-7.4	-	-5.8	-	-9	-	-	-	-9	-	-
O3 rounds											
1-2	-	-8	-	-	-8	-8	-	-	-7.4	-6.8	-
3-4	-	-7.4	-	-	-	-9	-	-	-	-8	-
5-6	-	-8	-	-	-	-8	-	-	-	-6.8	-
T	-9	-	-9	-9	-8	-9	-9	-7	-9	-	-7

Path	S-box number										
	11	12	13	14	15	16	17	18	19	20	21
O1	-	-8	-	-5.8	-	-9	-	-9	-	-9	-
O2		-	-	-7.4	-	-9	-	-9	-	-9	-
O3 rounds											
1-2	-	-9	-7.4	-9	-7.4	**−7.4**	**−8**	-9	-8	-9	-
3-4	-9	-	-7	-	-7	-	-7.4	**−8**	**−9**	-7	-7
5-6	-	-	-7.4	-	-7.4	-	-8	-8	-8	-7	-
T	-	-	-7	-	-7.4	-9	-	-9	-8	-8	-7

Path	S-box number									
	22	23	24	25	26	27	28	29	30	31
O1	-5.8	-	-5.8	-	-	-	-9	-	-	-
O2	-7.4	-	-	-	-7	-	-	-	-5.8	-
O3 rounds										
1-2	-	-9	-9	-7.4	-	-	-9	-9	-7	-9
3-4	-	-7	-	-5.8	-	-7	-9	-9	-	-9
5-6	-	-9	-	-7.4	-	-	-9	-9	-	-9
T	-7	-	-8	-7	-	-7	-7	-8	-	-6

Table 10: Paths Probabilities in Serpent-p given in log_2

	S-box number										
Path	0	1	2	3	4	5	6	7	8	9	10
B	-8	-	-	-	-	-	-	-	-	-	-
C	-	-	-7	-	-	-	-	-	-	-	-
E	-	-8	-	-	-	-	-	-	-	-	-
G	-	-	-7.4	-	-8	-	-	-	-	-	-
H	-	-	-	-	-	-	-	-9	-	-	-8
K	-	-	-	-	-	-	-	-9	-	-	-8

	S-box number										
Path	11	12	13	14	15	16	17	18	19	20	21
B	-	-	-7.4	-9	-	-	-	-9	-9	-	-
C	-	-	-	-	-	-	-	-9	-	-8	-
E	-	-	-	-	-	-	-	-	-	-	-
G	-	-	-	-	-	-	-	-	-	-	-
H	-	-	-	-	-7	-	-	-	-	-	-
K	-	-	-	-	-7	-	-	-	-	-	-

	S-box number									
Path	22	23	24	25	26	27	28	29	30	31
B	-	-	-	-	-	-	-9	-	-	-
C	-	-	-	-	-6.4	-	-	-	-	-
E	-	-9	-	-	-	-9	-	-	-	-
G	-	-	-	-	-	-	-	-	-	-
H	-	-	-	-	-	-	-	-	-	-
K	-	-	-	-	-	-	-	-	-	-

Table 11: Paths Probabilities in Serpent-p given in log_2

	S-box number								Longest	Prob-
Path	-	1	2	3	4	5	6	7	Path	ability
O1	-9	-9	-	-7	-7	-9	-9	-7	$S_6 \rightarrow S_2$	-34
O2	-9	-9	-	6	-8	-9	-	-9	$S_7 \rightarrow S_1$	-27
O3 rounds										
1-2	-	-9	-9	-8	-9	-	-9	-	$S_2 \rightarrow S_5$	-34
3-4	-9	-	-	-	-8	-9	-	-7		
5-6	-9	-9	-	-8	-8	-	-	-		
T	$\mathbf{-9}$	$\mathbf{-9}$	-	$\mathbf{-9}$	$\mathbf{-8}$	$\mathbf{-9}$	$\mathbf{-9}$	$\mathbf{-7}$	$S_3 \rightarrow S_1$	-61
A	-	-	-	-9	-	-	-	-		-9
B	-	-	-	-	-	-	-9	-		-9
C	-	-	-	-8	-	-	-9	-		-8
D	-	-	-	-	-	-	-	-8		-8
E	-9	-	-	-	-	-	-	-		-9
F	-	-	-	-	-8	-	-	-		-8
G	-	-	-	-	-	-9	-	-9		-9
H	-	-	-	-	-	-	-9	-9		-18
I	-9	-9	-	-	-	-7	-	-		-18
J	-	-	-	-	-	-	-	-		-
K	-	-	-	-	-	-	-9	-9		-18

Table 12: Paths Probabilities in Serpent-p-ns given in log_2