**Linear Cryptanalysis for a given S-Box using SPN cipher:**

The linear approximation for the given S-box is as follows: rows represent input sum in hexadecimal and columns represent output sum in hexadecimal.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **A** | **B** | **C** | **D** | **E** | **F** |
| **0** | 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **1** | 0 | 0 | 2 | -2 | 0 | 4 | 2 | 2 | 0 | 0 | 2 | -2 | 4 | 0 | -2 | -2 |
| **2** | 0 | -2 | 4 | 2 | -4 | 2 | 0 | -2 | -2 | 0 | -2 | 0 | -2 | 0 | -2 | 0 |
| **3** | 0 | -2 | -2 | 0 | 0 | 2 | -2 | 4 | -2 | 0 | 0 | -2 | -2 | -4 | 0 | 2 |
| **4** | 0 | -2 | -2 | 0 | -2 | 0 | 0 | -2 | -2 | 0 | 4 | 2 | 0 | -2 | 2 | -4 |
| **5** | 0 | -2 | 0 | -2 | -2 | -4 | 2 | 0 | -2 | 0 | -2 | 0 | 4 | -2 | 0 | 2 |
| **6** | 0 | 0 | -2 | 2 | -2 | 2 | 0 | 0 | 4 | 4 | -2 | 2 | 2 | -2 | 0 | 0 |
| **7** | 0 | 0 | 0 | 0 | 2 | 2 | -2 | -2 | -4 | 4 | 0 | 0 | 2 | 2 | 2 | 2 |
| **8** | 0 | -4 | 2 | -2 | 0 | 0 | 2 | 2 | 2 | 2 | 0 | 0 | -2 | 2 | 4 | 0 |
| **9** | 0 | 0 | 4 | 0 | 0 | 0 | -4 | 0 | 2 | -2 | 2 | 2 | 2 | -2 | 2 | 2 |
| **A** | 0 | -2 | -2 | -4 | 0 | 2 | -2 | 0 | 0 | -2 | -2 | 4 | 0 | 2 | -2 | 0 |
| **B** | 0 | 2 | 0 | -2 | -4 | -2 | -4 | 2 | 0 | 2 | 0 | -2 | 0 | 2 | 0 | -2 |
| **C** | 0 | -2 | 0 | 2 | 2 | 0 | -2 | 0 | 0 | -2 | -4 | -2 | 2 | 0 | 2 | -4 |
| **D** | 0 | 2 | 2 | -4 | 2 | 0 | 0 | -2 | 0 | 2 | -2 | 0 | -2 | -4 | 0 | -2 |
| **E** | 0 | 4 | 0 | 0 | -2 | 2 | 2 | 2 | -2 | -2 | -2 | 2 | 0 | 0 | 4 | 0 |
| **F** | 0 | 0 | 2 | 2 | 2 | -2 | 0 | 4 | -2 | 2 | 0 | 4 | 0 | 0 | -2 | -2 |

**Note:** This is the linear approximation table depicting number of matches of input sum and output sum in hexadecimal MINUS 8 - in accordance with the Hey's tutorial.

We wrote a program in Java called "spn.java" to generate this table.

1. We already have computed the information required to generate linear approximations in Answer 1. Using those values, we choose the following approximations:

2. Here, let represent the 16 bit block of bits at the input of round i of S boxes and

represent the bit of block . Similarly, for output, let it be and respectively. Let represent the subkey block of bits XOR-ed at input to round i. Thus, where P is block of 16 plaintext bits.

**Using linear approximation for Round 1:**

...........(2)

Probability =

**Using linear approximation for Round 2:**

...........(2a)

Probability =

As , thus,

................(2b)

Probability =

Combining (2) and (2b),

= 0 ............................(3)

Probability by Piling Lemma = =

Thus, linear bias =

**Using linear approximation for Round 3:**

, Probability = , Thus, linear bias =

, Probability = , Thus, linear bias =

Now,

and

Thus, ...............................(4)

Probability by Piling Lemma:

. Thus, linear bias =

Combining (3) and (4) to incorporate for all S-box approximations and calculating the probability by Piling Lemma, we get the combined probability as:

Thus, combined Bias =

**Cryptanalysis:**

**LINEAR APPROXIMATION USED:**

1. For the given S-box, we choose the following approximations for the spn cipher:

2. Here, let represent the 16 bit block of bits at the input of round i of S boxes and

represent the bit of block . Similarly, for output, let it be and respectively. Let represent the subkey block of bits XOR-ed at input to round i. Thus, where P is block of 16 plaintext bits.

**Using linear approximation for Round 1:**

...........(2)

Probability =

**Using linear approximation for Round 2:**

...........(2a)

Probability =

As , thus,

................(2b)

Probability =

Combining (2) and (2b),

= 0 ............................(3)

Probability by Piling Lemma = =

Thus, linear bias =

**Using linear approximation for Round 3:**

, Probability = , Thus, linear bias =

, Probability = , Thus, linear bias =

Now,

and

Thus, ..........................(4)

Probability by Piling Lemma:

. Thus, linear bias =

Combining (3) and (4) to incorporate for all S-box approximations and calculating the probability by Piling Lemma, we get the combined probability as:

Thus, combined Bias =

The linear approximations combining (3) and (4) is as follow:

Now, noting that:

Thus,   
 = 0

where

is either 0 or 1 depending on the key of the cipher. Since is fixed, we note that

= 0 .....................(5) must hold with probability or (1 - ) = depending on whether respectively.

The Linear expression (5) affects the inputs to S-boxes and .

**Thus, the target partial key bits are [].**

Method we used to find the partial target key:

1. We wrote a code in Java which reads in the given .json fie consisting of all cipertext-plaintext pairs for the given cipher.

2. For every possible value of [] and [] and for every given ciphertext-plaintext pair, we check if Equation (5) holds true.

3. We count the number of such matches for each value of [], [] and ciphertext-plaintext pair.

4. We find the probability of this count(dividing by the total number of possibilities of ciphertext-plaintext which is given to be 65536) and find the probability bias(the difference from 1/2). We take the absolute value while finding the probability bias as negative or positive bias depends on value of the key.

5. For all such possibilities, we find the maximum difference from 1/2.

6. For the linear approximation we chose, the maximum difference turns out to be around 0.013 and both the keys turn out to be 15.

That is, bit key values are **[] = [1111] = 0xF and [[1111] = 0xF.**

7. This is almost equal to the probability bias we had calculated from the linear approximation we chose which is .

8. Thus, the probability bias as per our linear approximation and the code match.

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**Reference:**

1. **Hey's tutorial** (Linear and Differential Cryptanalysis by Howard M. Heys)

**Link:** https://www.engr.mun.ca/~howard/PAPERS/ldc\_tutorial.pdf

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/