Advanced Topics in Cybersecurity 1 Project

Lorenzo Cian

June 2022

Contents

1 Description of the attack **1**

2 Compiling and running the code **4**

This project consists of an implementation of a differential attack on the block cipher KLEIN [2], based on the ideas of [1] and [3].

1 Description of the attack

The attack on $R \geq 3$ rounds of KLEIN works as follows:

- Let us consider the truncated differential of Figure 1. First we generate $C \times p^{-1}$ pairs, where C is an arbitrary constant and p^{-1} is the probability for the truncated differential to hold for $R-1$ rounds. We filter the pairs and keep only the good pairs, i.e. the ones which exhibit the desired pattern at the end of round $R-1$. We can verify if this happens by applying InverseMixColumn on the difference of the ciphertexts. We keep the pair if and only if this value has only lower nibbles possibly active, i.e. the higher nibbles have zero difference.

- For each good pair, we guess the nibbles of the first round key (which corresponds to the master key) corresponding to the four active nibbles in the input difference, XOR them with the corresponding nibbles in the plaintexts, then apply the KLEIN Sbox to these values. By Proposition 1 of [3], we know that the input difference at round 2 will be the one we expect if and only if the most significant bits of the values we just computed are all 0

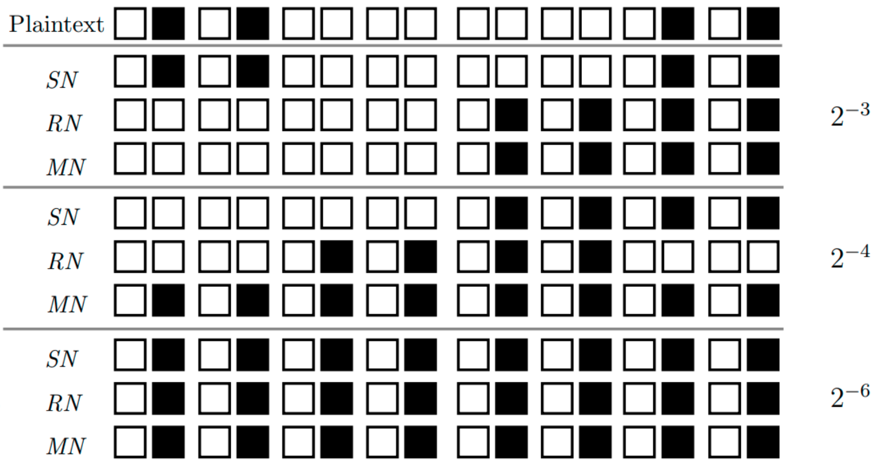

Figure 1: The first truncated differential. White squares indicate inactive nibbles, black squares indicate possibly active nibbles. On the right, the corresponding probability for the differential to hold for each round is shown. The following rounds all have probability 2^{-6}. Figure taken from Fig. 5, case IV of [3].

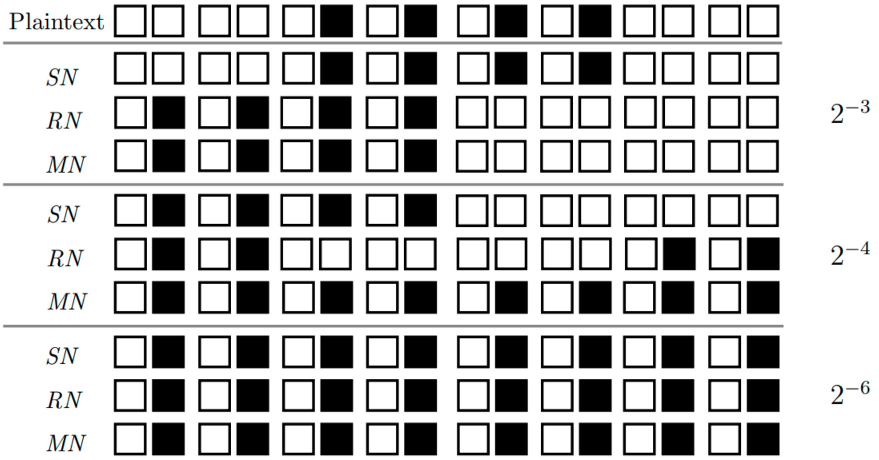

Figure 2: The second truncated differential. White squares indicate inactive nibbles, black squares indicate possibly active nibbles. On the right, the corresponding probability for the differential to hold for each round is shown. The following rounds all have probability 2^{-6}. Figure modified from Fig. 5, case IV of [3].

or all 1. Therefore, we can increment the counters only for the guesses that satisfy this condition. After trying all possible values for the target nibbles for a given good pair, we disable the candidates that do not have the highest counter (i.e. they will be ignored when processing the following good pairs for the same differential). This process will allow us to recover the lower nibbles of bytes $0, 1, 6, 7$ of the master key.

- We can do the same with the differential of Figure 2 to recover the lower nibbles of bytes $2, 3, 4, 5$ of the master key.
- The higher nibbles of the master key can be recovered by exhaustive search.

The complexity of the attack is:

- data: $O(2 \times 2 \times C \times p^{-1})$, as we need to collect $C \times p^{-1}$ many plaintext and ciphertext pairs for each differential;
- time: $O(2 \times 2 \times C \times p^{-1} + 2^{32})$, as we need to apply the process of generating, encrypting using the oracle and verifying if the pair is good or not for a total of $C \times p^{-1}$ pairs for each of the two differentials, then we need $O(2^{32})$ computations (assuming we successfully reduced the search space for the lower nibbles to just a single choice) to recover the higher nibbles via exhaustive search.

  I have found $C = 12$ to work best in practice, as it is the smallest value that allows us to reduce the search space for the lower nibbles of the master key to just 1 choice almost every time. With this choice of C, the time complexity will become: $O(48 \times p^{-1} + 2^{32})$;
- memory: constant.

So, by computing p (approximately) as explained above and in [3] and assuming we choose $C = 12$, we obtain the following results with the method explained above:

- for 4 rounds of KLEIN, $p = 2^{-13}$, thus the cipher can be broken with data complexity $O(2 \times 2 \times 12 \times 2^{13})$, which is less than $O(2^{19})$ and time complexity less than $O(2^{32})$ (dominated by the exhaustive search for the higher nibbles);
- for 5 rounds of KLEIN, $p = 2^{-19}$, thus the cipher can be broken with data complexity $O(2 \times 2 \times 12 \times 2^{19})$, which is less than $O(2^{25})$ and time complexity less than $O(2^{32})$ (dominated by the exhaustive search for the higher nibbles);
- for 6 rounds of KLEIN, $p = 2^{-25}$, thus the cipher can be broken with data complexity $O(2 \times 2 \times 12 \times 2^{25})$, which is less than $O(2^{31})$ and time complexity less than $O(2^{32})$ (dominated by the exhaustive search for the higher nibbles);

- breaking $R \geq 7$ rounds of KLEIN will require data and time complexity greater than $O(2^{32})$.

2 Compiling and running the code

If you are using gcc, then you can compile the attack by moving to the directory `src` and using the command:

```
gcc attack.c common.c klein.c speedklein64.h kleinSbox.h config.h -lm.
```

As the running time of the attack can be high, it is suggested to enable additional optimization flags, such as: `-march=native -O3`. You can also add the flag `-fopenmp` if you have OpenMP installed, as that will make the exhaustive search much faster by using all of the cores available on your machine.
Then run it by executing the compiled file.
The number of rounds and the master key can be changed by editing lines 6 and 10-11, respectively, of the file `src/config.h`.

References

[1] Jean-Philippe Aumasson, María Naya-Plasencia, and Markku-Juhani O. Saarinen. Practical attack on 8 rounds of the lightweight block cipher klein. In Daniel J. Bernstein and Sanjit Chatterjee, editors, *Progress in Cryptology – INDOCRYPT 2011*, pages 134–145, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[2] Zheng Gong, Svetla Nikova, and Yee Wei Law. Klein: A new family of lightweight block ciphers. In Ari Juels and Christof Paar, editors, *RFID. Security and Privacy*, pages 1–18, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[3] Virginie Lallemand and María Naya-Plasencia. Cryptanalysis of klein. In Carlos Cid and Christian Rechberger, editors, *Fast Software Encryption*, pages 451–470, Berlin, Heidelberg, 2015. Springer Berlin Heidelberg.