Supplementary Material

Experiments on round-reduced key-recovery attacks.

According to Eq. (12), in our attack, the data to store is at least $2^{n/2}/pq$, where n is the block size. So it is difficult for us to perform experimental attacks on typical block ciphers. But it is rather straightforward to verify the attack on a cipher with 32-bit block size. So we make experiments on round-reduced Simon32, whose block size is 32. The round function of Simon32/63 is given in Figure 1. We give an example on `Simon32/64` using a 6-round distinguisher with probability 2^{-2} in single-key setting. Appending 1-round E_b and 3-round E_f, we attack 10-round `Simon32/64` as Table 1, where ΔX_r is the input difference in round r $(0 \leq r \leq 10)$.

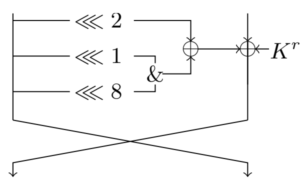

Fig. 1: rth Round function of Simon32/64

Table 1: The 10-round rectangle attack on `Simon32/64`.

ΔX_0	0100 0000 0000 0000 ?000 0000 0?00 0001
$\Delta X_1(\alpha)$	0000 0000 0000 0000 0100 0000 0000 0000
...	...
$\Delta X_7(\delta)$	0100 0000 0000 0000 0000 0000 0000 0000
ΔX_8	?000 0000 0?00 0001 0100 0000 0000 0000
ΔX_9	0?00 000? ?000 01?? ?000 0000 0?00 0001
ΔX_{10}	?000 0??? 0?01 ???? 0?00 000? ?000 01??

This experiment follows the single-key model in the submitted paper. As shown in Figure 1, the XORing of the first round K_0 can be placed at the input of left branch, hence, we just use the $\Delta X_1 = \alpha$ to collect the data, i.e., $m_b = 0$. We choose the expected number of right quartets $s = 4$, and construct 2^{18} pairs (P_1, P_2) satisfying the input difference α. The time of generating pair is 2^{18}. We invert the final round without guessing any key bits to derive ΔX_9 as filters to generate quartets. After the filter, the number of the remaining quartets is $2^{18+17-2\cdot 25} = 2^{-15}$. The time complexity to generate the quartets is 2^{18} with 2^{18} memory. The subkeys involved in E_f are 15 bits. We construct 2^{15} key counters. Using the remaining quartets, we first guess $K_9[0, 3-9, 12-15]$ to check whether

$\mathit{\Delta} X_8$ satisfies the (?000 0000 0?00 0001 0100 0000 0000 0000) for both (C_1, C_3) and (C_2, C_4). $2^{-15+12-2\cdot 5} = 2^{-13}$ quartets remain. Then we guess $K_8[5, 7, 14]$ to check whether $\mathit{\Delta} X_8$ satisfies δ difference. There are $2^{-13} \cdot 2^{3-2\cdot 2} = 2^{-14}$ quartets remain. So the time complexity of the key recovery process is $2^{-15} \cdot 2^{12} + 2^{-13} \cdot 2^3 = 2^{-3}$. This is because the first filter process delete most quartets.

In total, the data complexity is 2^{18} and the memory complexity is $\cdot 2^{18} + 2^{15} \approx 2^{18.17}$. The time complexity is also 2^{18} (We don't make experiments on the exhaustive search process). Set $h = 4$, and the success probability is 97.6%.

Experiments result. Testing with 100 different mater keys, if the right key candidate is in the top 2^{15-h} key counters, we consider the attack succeeds to gain a $h = 4$-bit advantage than the exhaustive search. The experiment need about 1 minute on one computer and the success rate is 100%. The code of the experiment can be found in `https://github.com/key-guess-rectangle/key-guess-rectangle`

References