Practical Forgery Attacks on SCREAM and iSCREAM

Siang Meng Sim and Lei Wang

Division of Mathematical Sciences, School of Physical and Mathematical Science,
Nanyang Technological University, Singapore

ssim011@e.ntu.edu.sg, wang.lei@ntu.edu.sg

Abstract. In this short article, we describe a practical forgery attack on the authenticated encryption mode SCREAM and iSCREAM, which have been submitted to the CAESAR competition. Our attack needs only 2 queries. The weakness of these two modes comes from the operations on the last plaintext block and on the tag generation.

Key words: authenticated encryption, CAESAR, SCREAM, iSCREAM, cryptanalysis, authenticity

Side-Channel Resistant Authenticated Encryption with Masking (SCREAM) is one of the CAESAR competition submissions [3], which is based on Liskov et al.'s Tweakable Authenticated Encryption (TAE) [1] with new designed tweakable block ciphers. iSCREAM is an involution version of SCREAM. In this paper, we will mainly describe the attack on SCREAM, but we stress that it is also applicable to iSCREAM since they share exactly the same mode.

Our contributions. In this paper, we present a simple and practical forgery attack on SCREAM and iSCREAM by exploiting the zero padding rule for non-full last plaintext block in the algorithm. First, we query a nonce, an associated data and a plaintext, whose last block is not full and moreover consists of only zero bits, to the encryption and receive a ciphertext and an authentication tag. Next, by modifying the last non-full block of the received ciphertext, we can create a successful forgery by making only one query to the decryption oracle.

1 Description of SCREAM

The encryption mode of SCREAM uses tweakable block cipher Scream, a variant of the LS-designs introduced in [2], in the Tweakable Authenticated Encryption (TAE) proposed Liskov et al. [1]. Since our attack is independent of Scream, we focus our description on the authenticated encryption mode. Nevertheless, we need to know that Scream takes three 128-bit inputs, a block of the plaintext P_i, a secret key K and a tweak T. The master key for the SCREAM is the same for all block ciphers, with a variation of the tweak. Using a nonce N, optional associated data A and plaintext P, SCREAM produces a ciphertext C and a tag TAG. For the decryption, a nonce N, an associated data A, a ciphertext C and a tag TAG are used to recover a plaintext P and another tag TAG^* is generated. If the tags TAG and $TAG*$ do not match, the SCREAM algorithm returns a null output $\perp$. Otherwise, it returns P.

There are three main steps in the encryption mode - processing the associated data, encrypting the plaintext and generating the tag, which can also be seen in Figure 1. Firstly, the associated data is divided into 128-bit blocks, each block is encrypted through Scream and the output values are XORed to get a authentication value $auth$. If the last block is not full, it is padded with single 1 bit followed by 0 bits until it is a full block. Secondly, the plaintext is also divided into 128-bit blocks, each block P_i (except the last block) is encrypted with Scream and the output ciphertext block C_i is stored. For the last cipher block, the length of the last block of plaintext $|P_{m-1}|$ is encrypted and the output is truncated to the same length as P_{m-1} and XORed with P_{m-1} to produce C_{m-1}. Lastly, the plaintext blocks are XORed to get a checksum $\sum$, where the last block

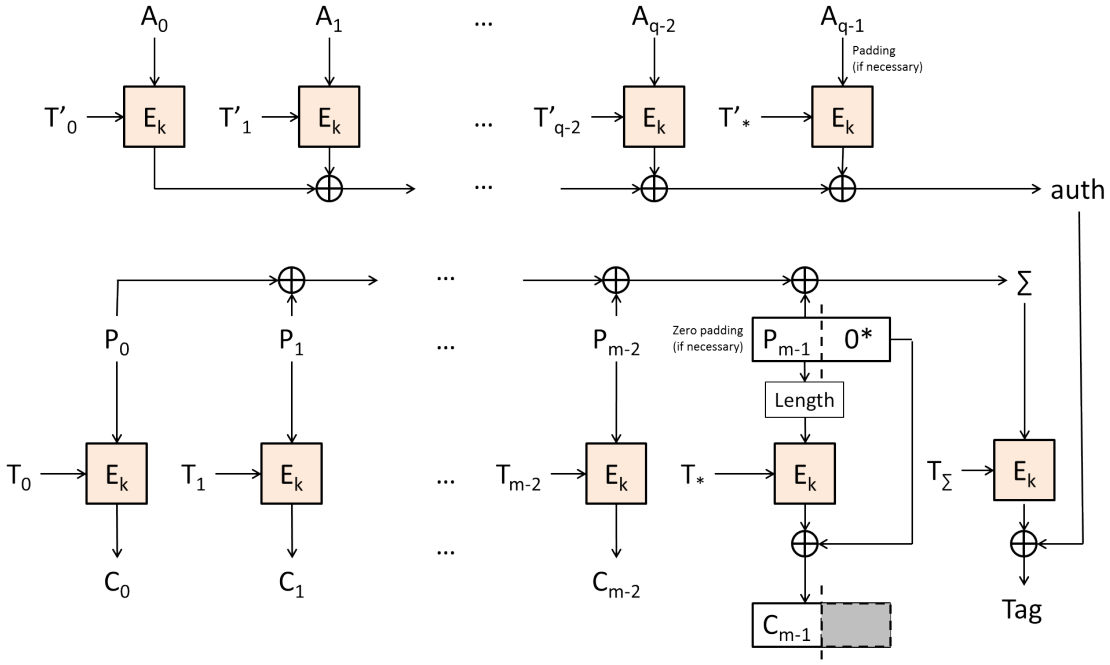

Fig. 1. The SCREAM authenticated encryption mode.

is padded with zeros if it is not a full block. The checksum is then encrypted and the output is XORed with *auth* to generate the tag.

For the security purpose, the tweaks used in each Scream are distinct. The following is the description of the tweaks for SCREAM, where 0^* is zero padding and c is the block counter starting from the first block to the second last block encryption of the associated data and continued on for the first block encryption of the plaintext until the second last block of the plaintext.

- Associated data processing.
  - For all but the last block:
    - $T'_c = (N\|10\|c)$
  - For the last full block:
    - $T'_* = (N\|11\|010\|0^*)$
  - For the last partial block:
    - $T'_* = (N\|11\|011\|0^*)$
- Plaintext encryption.
  - For all but the last block:
    - $T_c = (N\|0\|c)$
  - For the last full block:
    - $T_* = (N\|11\|000\|0^*)$
  - For the last partial block:
    - $T_* = (N\|11\|001\|0^*)$
- Tag generation.
  - For the last plaintext block that is a full block:
    - $T_\sum = (N\|11\|100\|0^*)$
  - For the last plaintext block that is a partial block:
    - $T_\sum = (N\|11\|101\|0^*)$

2 Attack on SCREAM

Our attack is based on the zero padding on the last plaintext partial block. First, we query a nonce N, an associated data A and a plaintext P to the encryption oracle. The choice of nonce and associated data can be arbitrary. For the plaintext P, we let it be an arbitrary message that ends with a partial block consisting of all 0 bits. Here we use $P = (P_0\|P_1\|...\|P_{m-2}\|00)$, where the last partial block consists of two zero bits, as an example. The output will be a tag TAG and a ciphertext with two bits in the last partial block, $C = (C_0\|C_1\|...\|C_{m-2}\|**)$. It is also shown in Figure 2.

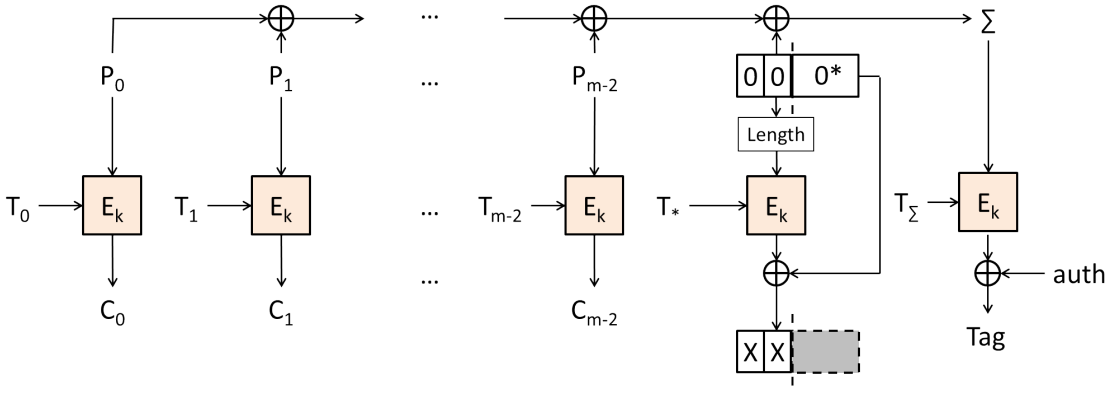

Fig. 2. The single query on SCREAM authenticated encryption mode.

Next, let us analyze the encryption procedure with the inputs of the same nonce N, the same associated data A and a plaintext P' modified from P by changing the last partial block to only a single 0 bit, that is $P' = (P_0\|P_1\|...\|P_{m-2}\|0)$. The encryption procedure can be seen from Figure 3. We get that the tag will surely be equal to TAG and the ciphertext are all known except the single bit as the last partial ciphertext block.

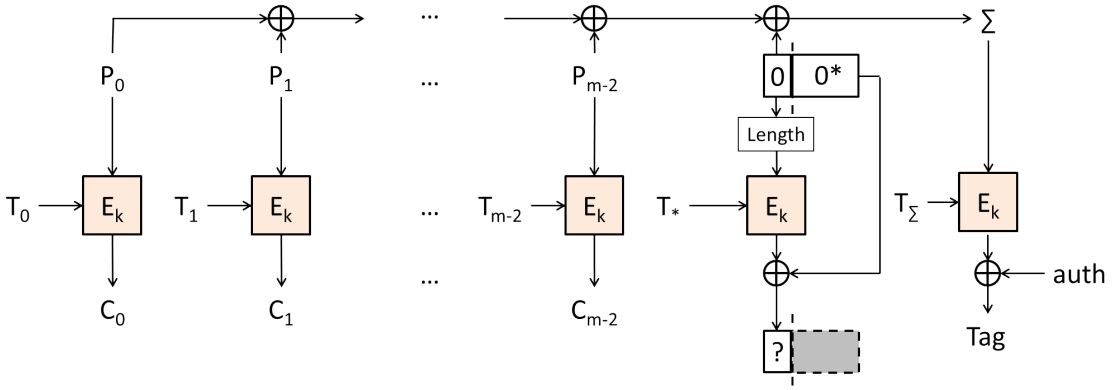

Fig. 3. The forgery on SCREAM authenticated encryption mode.

Therefore, we can query the decryption oracle with $(N, A, C' = (C_0\|C_1\|...\|C_{m-2}\|0), TAG)$ If it is valid, we have found a forgery successfully. Otherwise, we can conclude that $(N, A, C'' = (C_0\|C_1\|...\|C_{m-2}\|1), TAG)$ is a valid forgery.

3 Conclusion

By making only two queries - one encryption query and one decryption query, we have obtained a successful forgery on `SCREAM` and `iSCREAM`.

References

1. M. Liskov, R.L. Rivest, D. Wagner: Tweakable block ciphers. In: J. Cryptology, 24 (2011), pp. 588-613.
2. V. Grosso, G. Leurent, F.-X. Standaert, K. Varici: LS-designs: Bitslice encryption for efficient masked software implementations. In: FSE 2014.
3. V. Grosso, G. Leurent, F.-X. Standaert, K. Varici, F. Durvaux, L. Gaspar, S. Kerckhof: SCREAM and iSCREAM: Side-Channel Resistant Authenticated Encryption with Masking. In: CEASAR competition.