Cryptanalysis of the MALICIOUS Framework*

Tim Beyne and Chaoyun Li

imec-COSIC, ESAT, KU Leuven, Belgium
name.lastname@esat.kuleuven.be

This note describes several attacks on the MALICIOUS framework for creating backdoored tweakable block ciphers [2]. It is shown that, although the embedded malicious tweak pair itself is hard to recover, it is feasible to find additional weak tweak pairs that can be used to mount key-recovery attacks. Full-round attacks on most instances of LowMC-M are given. Our attacks are far from optimized and significant future improvements are to be expected.

We focus on low-data attacks, since these are the most relevant for typical use-cases of LowMC. In addition, this implies that our attacks can not be prevented by limiting the amount of data that can be encrypted using the weak tweak pair.

Despite our findings, we believe that the MALICIOUS framework can be used to create backdoored variants of LowMC provided that the parameters are modified.

1 Malicious and Weak Tweak Pairs

Let n denote the block size in bits, k the key size, s the size of the nonlinear part, and r the number of rounds. Peyrin and Wang [2] argue that, since the malicious round tweak difference is unique with overwhelming probability, finding a malicious tweak pair costs roughly $2^{(n+(r-1)s)/2}$ evaluations of the XOF which is used as the tweak-schedule (assuming the tweak is long enough).

As noted by the authors, this reasoning does not take into account the existence of tweak pairs which might be a backdoor for a different input difference. In the following section, we compute the probability that this is the case for a random round-tweak difference. It will be argued that *some* weak tweak pair can be found at a cost of roughly $2^{(rs-n)/2}$ XOF evaluations. Although this is a much lower cost than for finding the backdoor itself, it does not allow an attacker to find a malicious tweak pair in time lower than the security level of 2^k for any of the LowMC-M instances because $rs - n \gg 2k$.

Nevertheless, it is feasible to find 'weak' tweak pairs such that a probability one related-tweak differential exists for some smaller number of rounds. It will be shown in Section 2 how such a pair can be used to set up a key-recovery attack on full-round instances of LowMC-M.

1.1 Counting Weak Round Tweak Differences

A round-tweak difference $(\Delta t_0, \Delta t_1, \ldots)$ will be called *weak* if there exists a differential characteristic with input difference $\Delta_1 \in \mathbb{F}_2^n$ such that the nonlinear

* The work of Tim Beyne was supported by a PhD Fellowship from the Research Foundation-Flanders (FWO).

part of the state is inactive in the first r rounds. Note that $\varDelta t_i \in \mathbb{F}_2^n$, but the bottom $n-s$ coordinates are zero when $i > 0$. The difference before the S-box layer in round $i+1 > 1$ is given by $\varDelta_{i+1} = L_i\,\varDelta_i + \varDelta t_i$ where $L_i \in \mathbb{F}_2^{n\times n}$ is the linear layer of round i. Consequently,

$$\varDelta_i = (L_i L_{i-1} \cdots L_1)\varDelta_1 + \sum_{j=0}^{i} (L_j L_{j-1} \cdots L_1)\varDelta t_j.$$

Let $\lfloor\cdot\rfloor_s$ denote the first s coordinates of some vector. The tweak difference $(\varDelta t_0, \varDelta t_1, \ldots)$ does not activate the nonlinear part in the first r rounds if

$$\sum_{j=0}^{l} \lfloor (L_l L_{l-1} \cdots L_{j+1})\,\varDelta t_j \rfloor_s = \lfloor (L_l L_{l-1} \cdots L_1)\,\varDelta_1 \rfloor_s,$$

for any $l \in \{1, \ldots, r\}$. For any fixed choice of the tweak difference variables, this results in a system of $s \times r$ linear equations in n unknowns. For random linear layers, and assuming $s \times r \gg n$, such a system will be inconsistent with high probability. More precisely, the probability that a random choice of the first r round tweaks results in a right hand side that makes the system consistent, will be 2^{n-rs}. Indeed, the column space of the coefficient matrix of the linear system is of dimension n in an ambient space of dimension $r \times s$ [2, p. 21-22].

1.2 Finding Weak Round Tweak Differences

A tweak pair such that the round-tweak differences $(\varDelta t_0, \ldots, \varDelta t_{r-1})$ result in a consistent linear system can be found by using collision search methods at the cost of roughly $2^{(rs-n)/2}$ XOF evaluations, and including the cost of a multiplication by a $(rs-n) \times rs$ matrix to account for the checking of the consistency of the system of equations above. The amount of memory required depends on the input size of the XOF $H : \mathbb{F}_2^{\ell} \to \mathbb{F}_2^{n+(r-1)s}$. It will be assumed that $\ell \geq (rs-n)/2$, which will be the case throughout this note.

Specifically, let $A \in \mathbb{F}_2^{rs\times(rs-n)}$ be a matrix with column space the orthogonal complement of the column space of the coefficient matrix of the system of equations. Let $B \in \mathbb{F}_2^{rs\times[n+(r-1)s]}$ be the matrix mapping the round tweak to the right-hand side of the equations. The goal is to find a collision for the function $f : \mathbb{F}_2^{\ell} \to \mathbb{F}_2^{rs-n}$ defined by $f(t) = A^{\top} \times B \times H(t)$.

If $\ell \gg (rs-n)/2$, then a parallel collision search using Van Oorschot-Wiener collision search costs roughly $2^{(rs-n)/2}$ extended (by a single matrix-vector multiplication) XOF evaluations with little memory [4]. If $\ell \approx (rs-n)/2$, then a golden collision search with a cost of $\mathcal{O}(2^{3\ell/2}/\sqrt{M})$ evaluations and M memory can be used.

To conclude, a weak tweak pair can be found with a computational cost of $2^{(rs-n)/2}$ extended XOF evaluations. We assume the attacker is capable of $2^c \leq 2^k$ such evaluations. Note that a small constant factor in the collision search cost is neglected here, which can be justified by arguing that an extended

XOF evaluation takes significantly less time than a single LowMC-M evaluation. The memory cost depends on ℓ, which we assume to be at least as large as the security level k.

2 Key-Recovery Attacks

This section describes two key-recovery approaches that exploit weak round-tweak differences: a simple differential-linear attack and a difference-enumeration attack. The latter attack is simply an adaptation of the attack by Rechberger *et al.* [3]. Both attacks can be used in the low-data setting.

2.1 Simple Differential-Linear Attack

By the results in the previous section, an adversary capable of 2^c extended XOF evaluations can find a weak tweak pair such that there exists a differential characteristic with probability one over the first $r_1 = \lfloor (2c + n)/s \rfloor$ rounds of LowMC-M. Denote this first part of the cipher by F_1.

By choosing an appropriate input mask u and output mask v, one can always find a linear trail over $r_2 = \lfloor n/s \rfloor$ rounds of LowMC which does not activate any S-boxes. This approximation over the second part of the cipher, F_2, can be combined with the deterministic differential $\Delta_1 \to \Delta_2$ over F_1. Indeed,

$$v^\top (F_2 \circ F_1)(x + \Delta_1) = v^\top (F_2 \circ F_1)(x) + u^\top \Delta_2.$$

Consequently, one obtains a differential-linear distinguisher for $r_1 + r_2 \approx 2(c + n)/s$ rounds of LowMC-M. The data requirements of the distinguisher are minimal.

Finally, we can set up a key-recovery attack based on this distinguisher. The last r_3 rounds of LowMC can be decrypted by guessing $r_3 \times s$ bits. For each such guess, the attacker can proceed as follows:

1. Partially decrypt the ciphertext pairs through the last r_3 rounds. The decryption operation requires one unstructured and one highly structured matrix multiplication per round. When the number of tested pairs is small and r_3 is much smaller than the total number of rounds, the total time-complexity of this computation does not exceed that of a single LowMC-M evaluation.
2. On average, two pairs will suffice to discard a wrong guess.

Once the last r_3 rounds have been decrypted, the attacker may proceed to decrypt the remaining rounds in a round-by-round manner. To ensure that the candidate key is likely to be unique in each step, $r_3 \times s$ plaintext pairs suffice. In this case, the memory usage is low and the time requirements are dominated by the first step, which has a total computational complexity of less than $2^{r_3 s}$ LowMC-M evaluations. As shown in Table 1, a full-round attack is possible for several instances of LowMC-M.

Table 1. Cost of the basic differential-linear attack assuming $n = k$, and for several values of c. Only instances for which a full-round attack is possible are shown. The data requirements could be reduced (but not below 4) at a modest increase in time complexity.

							Key-recovery	
		s	r	r_1	r_2	r_3	$\log_2$(Time)	Data
$n = 128$	$c = 128$	3	208	128	42	38	114	228
		6	104	64	21	19	114	228
		9	70	42	14	14	126	252
$n = 256$	$c = 256$	3	384	256	85	43	129	258
		9	129	85	28	16	144	288
	$c = 196$	3	384	216	85	83	249	498

The number of rounds covered by this attack is at most

$$r_1 + r_2 + r_3 \leq \left\lfloor \frac{2c + k}{s} \right\rfloor + \left\lfloor \frac{n}{s} \right\rfloor + \left\lfloor \frac{k}{s} \right\rfloor.$$

If $c = k$, then the above simplifies to roughly $(n + 4k)/s$ rounds.

2.2 Modified Difference Enumeration Attack

This section gives a better attack by slightly modifying the difference-enumeration attacks from Rechberger *et al.* [3]. For simplicity, we consider only the case $d = 1$.

The attack covers the first r_1 rounds of the cipher using a deterministic difference. In LowMC without a tweak, the largest possible choice of r_1 is[1]

$$r_1^{\mathsf{LowMC}} = \left\lfloor \frac{n}{s} \right\rfloor.$$

In LowMC-M, however, this number of rounds can be significantly increased by choosing a good weak tweak pair. Finding a weak tweak pair can be done in time $2^{(rs-n)/2}$. For an attacker with the capability of 2^c extended XOF evaluations, the number of rounds r_1 thus increases to

$$r_1 = \left\lfloor \frac{2c + n}{s} \right\rfloor.$$

Let δ denote the average number of possible output differences over the S-box layer for a uniform random input difference. Recall that we have $\delta = (29/8)^{s/3}$ for LowMC [3, Sect. 3.1.3]. In the next r_2 rounds, all δ^{r_2} possible differences in the forward direction are enumerated. In the final r_3 rounds, the differences

[1] A few extra rounds may be possible if s does not divide n, but this will be ignored for simplicity.

are enumerated in the backward direction. The differences are matched in the middle, which means that $\delta^{r_2+r_3} < 2^n$ should hold in order to avoid random collisions. That is, $r_2 + r_3 < n/\log_2 \delta$ must hold. The complexity of this distinguisher is dominated by the list creation, which amounts to $\max\{\delta^{r_2}, \delta^{r_3}\}$ memory accesses.

For key-recovery, one also has to compute the characteristic (which is likely to be unique) followed by the inputs. This can be done in roughly $\delta^{r_2} + \delta^{r_3}$ time for each input pair using a meet-in-the-middle approach. Due to the fact that the LowMC S-box is differentially 2-uniform, the key-recovery step requires only two plaintext pairs. The time-complexity of the entire attack is thus dominated by $2(\delta^{r_2} + \delta^{r_3})$ storage operations. The storage requirements are $n(\delta^{r_2} + \delta^{r_3})$ bits. To optimize the time-complexity, we set $r_2 \approx r_3$. Specifically,

$$r_2 = \left\lfloor \frac{r - r_1}{2} \right\rfloor \quad \text{and} \quad r_3 = \left\lceil \frac{r - r_1}{2} \right\rceil .$$

The complexities for full-round LowMC-M are given in Table 2. For all instances except those with the largest value of s (for $n = 128$, $s = 90$ and for $n = 256$, $s = 120$), one can find a weak tweak pair in less than 2^k time such that the attack improves over brute-force.

2.3 Other Strategies

The attacks described above both apply to the low-data setting. In principle, the LowMC-M specification allows for up to 2^{64} chosen plaintexts. To exploit this data, it would be natural to consider a standard differential attack. Based on the calculations in the original LowMC paper [1] related to differential characteristics, one would conclude that full-round attacks are possible. Nevertheless, the estimates in [1] – coming from the design point of view – are very conservative. A more detailed investigation seems to be necessary to obtain good cost estimates.

Similarly, the differential-linear attack could be significantly improved by adding a statistical part.

References

1. Albrecht, M.R., Rechberger, C., Schneider, T., Tiessen, T., Zohner, M.: Ciphers for MPC and FHE. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015, Part I. LNCS, vol. 9056, pp. 430–454. Springer, Heidelberg (Apr 2015)
2. Peyrin, T., Wang, H.: The MALICIOUS framework: Embedding backdoors into tweakable block ciphers. In: Micciancio, D., Ristenpart, T. (eds.) Advances in Cryptology - CRYPTO 2020 - 40th Annual International Cryptology Conference, CRYPTO 2020, Santa Barbara, CA, USA, August 17-21, 2020, Proceedings, Part III. Lecture Notes in Computer Science, vol. 12172, pp. 249–278. Springer (2020), https://doi.org/10.1007/978-3-030-56877-1_9
3. Rechberger, C., Soleimany, H., Tiessen, T.: Cryptanalysis of low-data instances of full LowMCv2. IACR Trans. Symm. Cryptol. 2018(3), 163–181 (2018)
4. van Oorschot, P.C., Wiener, M.J.: Parallel collision search with cryptanalytic applications. Journal of Cryptology 12(1), 1–28 (Jan 1999)

Table 2. Cost of the difference-enumeration attack assuming $n = k$, and for several values of c. Only instances for which a full-round attack is more efficient than brute force are listed. Memory requirements are listed in bits.

							Key-recovery		
		s	r	r_1	r_2	r_3	$\log_2$(Time)	$\log_2$(Memory)	Data†
$n = 128$	$c = 128$	3	208	128	40	40	76.32	82.32	4
		6	104	64	20	20	76.32	82.32	4
		9	70	42	14	14	80.04	86.04	4
		30	23	12	5	6	112.48	118.48	4
	$c = 96$	3	208	106	51	51	96.76	102.76	4
		6	104	53	25	26	97.72	103.72	4
		9	70	35	17	18	101.36	107.36	4
	$c = 64$	3	208	85	61	62	116.55	122.55	4
		6	104	42	31	31	117.19	123.19	4
		9	70	28	21	21	119.05	125.05	4
$n = 256$	$c = 256$	3	384	256	64	64	120.91	127.91	4
		9	129	85	22	22	124.63	131.63	4
		60	21	12	4	5	186.80	193.80	4
	$c = 196$	3	384	213	85	86	161.14	168.14	4
		9	129	71	29	29	163.64	170.64	4
		60	21	10	5	6	223.96	230.96	4
	$c = 128$	3	384	170	107	107	200.80	207.80	4
		9	129	56	36	37	207.27	214.27	4

† As noted by Rechberger *et al.* [3, §4.2.1], it might be necessary to use slightly more than two pairs to ensure distinct differences over the S-boxes are available.