More on Shabal's permutation

Jean-Philippe Aumasson[1], Atefeh Mashatan[2], and Willi Meier[1]

[1] FHNW, Windisch, Switzerland
[2] EPFL, Lausanne, Switzerland

1 Introduction

1.1 Shabal

Shabal [2] is a hash function candidate to the NIST Hash Competition, whose iterated hash scheme is based on a keyed permutation called $\mathcal{P}$ (which may be seen as a block cipher).

A chaining value of Shabal is a string of $(384 + 512 + 512)$ bits, and is denoted (A, B, C). Given a message block M of 512 bits, the compression function of Shabal computes a new chaining value (A', B', C') as follows:

$$(A', B') \leftarrow \mathcal{P}_{M,C}(A \oplus W, B + M)$$
$$(A', B', C') \leftarrow (A', C - M, B') \ .$$

Here, M and C act as the key of $\mathcal{P}$, W is a counter incremented at each new compression, and arithmetic operations are computed wordwise with respect to 32-bit words. The mode of operation of Shabal repeats this compression function for each message block, and finally makes three additional compressions with the last message block, without incrementing W, and it returns the final value of B.

The permutation $\mathcal{P}$ is parametrized by two values:

- p, the number of loops performed in $\mathcal{P}$; larger values of p provide better security guarantees.
- $r \geq 2$, the size of A (in 32-bit words).

Furthermore, p and r should satisfy $16p \equiv 0 \bmod r$. The designers of Shabal recommend $(p, r) = (3, 12)$, which are the parameters of the Shabal version submitted to the NIST Hash Competition.

1.2 Previous observations

One the requirements set by NIST [5] is that candidate algorithms, when used with HMAC to construct a PRF, "must resist any distinguishing attack that requires much fewer than $2^{n/2}$ queries and significantly less computation than a preimage attack." It is thus important for SHA-3 candidates to act as random transforms, yet NIST does not require optimal security against distinguishers (which is close to 2^n queries).

One can note that the compression function of Shabal was not designed to act as a PRF (see e.g. [2, Theorem 6]). Moreover, due to Shabal's original structure, the compression function (denoted $\mathcal{R}$ in [2]) transforms a chaining value (A, B, C) to a new chaining value (A', B', C'), such that $B' = C - M$. Given a random and unknown chaining value, one can thus easily distinguish that transform from a random function: query the function with distinct blocks M_1 and M_2, then check that the values obtained satisfy $B'_1 - B'_2 = M_2 - M_1$.

Therefore, in this note we will rather concentrate on $\mathcal{P}$, as it is used in Shabal, than on the compression function (that is, we disregard the output $B' = C - M$).

Although $\mathcal{P}$ does not formally need to be a pseudorandom permutation (PRP) to guarantee Shabal's security, previous works considered algorithms to distinguish $\mathcal{P}$ from a random permutation: The documentation of Shabal [2, §11.6] presents a simple distinguisher for the inverse permutation $\mathcal{P}^{-1}$. Then, in [1] we presented a distinguisher for $\mathcal{P}$ with $(p, r) = (3, 12)$ rounds. But as noted in [3], this is not applicable when Shabal's mode of operation is followed, that is, when the operations $B + M$ and $C - M$ outside $\mathcal{P}$ are considered. Also, those results do not generalize to any parameters (p, r).

Recently, Knudsen, Matusiewicz, and Thomsen [4] presented more properties for $\mathcal{P}$, with fixed points and key-multicollisions (that is, several values of (M, C) such that $\mathcal{P}_{M,C}(A, B) = (A', B')$, for some fixed A, B, A', B'). Their results also generalize to any (p, r), and are independent and different from ours.

1.3 Contribution

In this note, we show an algorithm that distinguishes $\mathcal{P}$ from a random permutation, while respecting Shabal's mode of operation, for any valid choice of (p, r). As an aside, we present simple pseudo-collision and pseudo-second-preimage attacks against a slight variant of Shabal; unlike trivial "pseudo" attacks that work by inverting $\mathcal{P}$, the two IV's differ on only 32 bits in our attacks. These results may be seen as near collisions with respect to the IV. Finally, we propose simple tweaks to avoid the properties highlighted.

2 The keyed permutation $\mathcal{P}$

Shabal's keyed permutation $\mathcal{P}$ works on words of 32 bits. Given inputs A, B, C, and M, it thus operates on $A_0, \ldots, A_{11}$, $B_0, \ldots, B_{15}$, $C_0, \ldots, C_{15}$, and $M_0, \ldots, M_{15}$. In the algorithm below, indices of A are implicitly reduced modulo r, and indices of B, C, and M are reduced modulo 16.

1. **for** $i = 0, \ldots, 15$
2. $\quad B_i \leftarrow B_i \lll 17$
3. **for** $j = 0, \ldots, p-1$
4. $\quad$ **for** $i = 0, \ldots, 15$
5. $\quad\quad A_{i+16j} \leftarrow 3 \times (A_{i+16j} \oplus 5 \times (A_{i-1+16j} \lll 15) \oplus C_{8-i})$
6. $\quad\quad A_{i+16j} \leftarrow A_{i+16j} \oplus B_{i+13} \oplus (B_{i+9} \wedge \neg B_{i+6})$
7. $\quad\quad A_{i+16j} \leftarrow A_{i+16j} \oplus M_i$
8. $\quad\quad B_i \leftarrow (B_i \lll 1) \oplus \neg A_{i+16j}$
9. **for** $i = 0, \ldots, 35$
10. $\quad A_i \leftarrow A_i + C_{i+3}$

Table 1 shows the indices used in the main loop of $\mathcal{P}$, for the recommended parameters $(p, r) = (3, 12)$, while Table 2 shows the indices of the last loop.

Some preliminary observations about the main loop of $\mathcal{P}$ are:

- Indices of B, C, and M do not depend on j
- Exactly one word from C and one word from M are input (through a XOR) within a loop, and the same pairs (C_{8-i}, M_i) repeat for different j's
- A difference in A_{i+16j} propagates to B_i with probability one
- The algebraic degree of the new value of B_i is at most that of the new value of A_{i+16j}

3 Distinguisher for $\mathcal{P}$ as used in Shabal

We present a related-key distinguisher for the compression function of Shabal, for any parameters (p, r). For ease of exposition, we present this observation of the recommended value $(p, r) = (3, 12)$, but the result trivially generalizes to any (p, r).

3.1 Key observation

Let $\Delta = \texttt{80000000}$. In Shabal, the multiplication by 3 preserves a difference Δ with probability one; this can be seen by rewriting

$$\begin{aligned} 3 \times (x + \Delta) &= ((x \oplus \Delta) \ll 1) + (x \oplus \Delta) \\ &= (x \lll 1) + (x \oplus \Delta) \\ &= ((x \lll 1) + x) \oplus \Delta = (3 \times x) \oplus \Delta \ . \end{aligned}$$

To exploit this property, we observe that in $\mathcal{P}$ the words $C_0, \ldots, C_{15}$ only appear in

$$A_{i+16j} \leftarrow 3 \times (A_{i+16j} \oplus 5 \times (A_{i-1+16j} \lll 15) \oplus C_{8-i}) \ .$$

Table 1. Word indices used in the main loop of $\mathcal{P}$, for $(p, r) = (3, 12)$.

j	i	A_{i+16}	A_{i+16-1}	B_{i+13}	B_{i+9}	B_{i+6}	C_{8-i}	M_i
0	0	0	11	13	9	6	8	0
0	1	1	0	14	10	7	7	1
0	2	2	1	15	11	8	6	2
0	3	3	2	0	12	9	5	3
0	4	4	3	1	13	10	4	4
0	5	5	4	2	14	11	3	5
0	6	6	5	3	15	12	2	6
0	7	7	6	4	0	13	1	7
0	8	8	7	5	1	14	0	8
0	9	9	8	6	2	15	15	9
0	10	10	9	7	3	0	14	10
0	11	11	10	8	4	1	13	11
0	12	0	11	9	5	2	12	12
0	13	1	0	10	6	3	11	13
0	14	2	1	11	7	4	10	14
0	15	3	2	12	8	5	9	15
1	0	4	3	13	9	6	8	0
1	1	5	4	14	10	7	7	1
1	2	6	5	15	11	8	6	2
1	3	7	6	0	12	9	5	3
1	4	8	7	1	13	10	4	4
1	5	9	8	2	14	11	3	5
1	6	10	9	3	15	12	2	6
1	7	11	10	4	0	13	1	7
1	8	0	11	5	1	14	0	8
1	9	1	0	6	2	15	15	9
1	10	2	1	7	3	0	14	10
0	11	3	2	8	4	1	13	11
1	12	4	3	9	5	2	12	12
1	13	5	4	10	6	3	11	13
1	14	6	5	11	7	4	10	14
1	15	7	6	12	8	5	9	15
2	0	8	7	13	9	6	8	0
2	1	9	8	14	10	7	7	1
2	2	10	9	15	11	8	6	2
2	3	11	10	0	12	9	5	3
2	4	0	11	1	13	10	4	4
2	5	1	0	2	14	11	3	5
2	6	2	1	3	15	12	2	6
2	7	3	2	4	0	13	1	7
2	8	4	3	5	1	14	0	8
2	9	5	4	6	2	15	15	9
2	10	6	5	7	3	0	14	10
2	11	7	6	8	4	1	13	11
2	12	8	7	9	5	2	12	12
2	13	9	8	10	6	3	11	13
2	14	10	9	11	7	4	10	14
2	15	11	10	12	8	5	9	15

Table 2. Word indices used in the final loop of $\mathcal{P}$, for $(p, r) = (3, 12)$.

i	0	1	2	3	4	5	6	7	8	9	10	11
A_i	0	1	2	3	4	5	6	7	8	9	10	11
C_{i+3}	3	4	5	6	7	8	9	10	11	12	13	14
i	12	13	14	15	16	17	18	19	20	21	22	23
A_i	0	1	2	3	4	5	6	7	8	9	10	11
C_{i+3}	15	0	1	2	3	4	5	6	7	8	9	10
i	24	25	26	27	28	29	30	31	32	33	34	35
A_i	0	1	2	3	4	5	6	7	8	9	10	11
C_{i+3}	11	12	13	14	15	0	1	2	3	4	5	6

Therefore, a difference Δ in C_{8-i} leads to a difference Δ in A_{i+16j} with probability one. Then, $\mathcal{P}$ sets

$$A_{i+16j} \leftarrow A_{i+16j} \oplus M_i \ ,$$

which also transfers a difference Δ to A_{i+16j} with probability one. We can thus cancel the difference introduced by C_{8-i} with an identical difference in M_i. To summarize, given a difference Δ in C_{8-i} and in M_i, for any i('s), the values of B returned by $\mathcal{P}$ do not change. However, some differences will be introduced in A by the final loop of $\mathcal{P}$. For example, with a difference in C_8, there will be differences Δ in A_5 and in A_9 (cf. Table 2).

Given any key (M, C), one can thus find exactly $2^{16} - 1 = 65535$ "semi-equivalent keys" for $\mathcal{P}$. These keys (M', C') are such that $M'_i = M_i \oplus \Delta$ and $C'_{8-i} = C_{8-1} \oplus \Delta$, for i in any proper subset $\mathcal{S}$ of $\{0, \ldots, 15\}$, and $M'_j = M_j$, $C'_{8-j} = C_{8-j}$ for $j \in \{0, \ldots, 15\} \setminus \mathcal{S}$.

3.2 The distinguisher

The observation above directly gives a related-key distinguisher for the keyed permutation $\mathcal{P}$. Recall that in this scenario, one is given a black-box access to a permutation π that is either $\mathcal{P}$ or a random permutation with a same interface, and a randomly chosen key. One can thus identify π with one query $\pi_{M,C}(A, B)$ for some arbitrary A and B, and one query $\pi_{M',C'}(A, B)$ with the same A and B, and where (M', C') is a related key of the above form (it suffices to flip one bit of M_0 and one bit of C_8, for example).

This distinguisher can be adapted to $\mathcal{P}$ as used by Shabal's mode of operation, that is, when $\mathcal{P}$ is applied to $(A \oplus W, B + M)$. It suffices to query with the input (A, B, C, M), and then with (A, B', C', M') where for some i we have $M'_i = M_i \oplus \Delta$, $B'_i = B_i \oplus \Delta$, and $C'_{8-i} = C_{8-i} \oplus \Delta$. The difference Δ in B' ensures that there will be no difference in the input of $\mathcal{P}$ (since M is added wordwise to B before the call to $\mathcal{P}$). Given two inputs with those differences, $\mathcal{P}$ will produce values with same B''s.

This distinguisher generalizes to any (p, r) other than $(3, 12)$: in each single loop, the differences Δ cancel out for any i, hence repeating the main loop 3 or r times does not change anything; the parameter r, defining the size of A, also has no influence on our distinguisher (note that, as stated in [2, §2.3], "the final loop of $\mathcal{P}$ (...) is not fully generic towards the parameter r").

Finally, note that our observation does not give a distinguisher with respect to the standard definition of pseudorandom permutation (PRP), but rather in the related-key setting. Moreover, we cannot mount a distinguisher on HMAC-Shabal, nor a distinguisher on Shabal with a secret IV. This is because the difference introduced gives a difference in the new chaining value (namely, in A', and in the new $B' = C - M$), and because of the finalization procedure of Shabal.

4 Pseudo collisions and pseudo second preimages for a variant of Shabal

We now consider a variant of Shabal that makes $24N$ iterations, $N \geq 2$, instead of 36, in its final loop.

Consider two inputs to $\mathcal{P}$ with difference Δ in C_i and M_i for all $i = 0, \ldots, 15$. As shown before, there will be no difference in the B' returned by $\mathcal{P}$, but there will also be no difference in A'. Indeed, the final loop will then introduce the difference Δ in each A_i an even number of times, hence these differences will cancel out.

However, the operation $C - M$ can also make differences vanish in C, for a well-chosen C (or with probability 2^{-16} for a random C). Therefore, the new chaining value (A', B', C') will contain no difference. We thus obtain a pseudo collision, by appending (at least) one same block to our colliding pair.

Finally, a collision can be found instantaneously for the variant of Shabal considered, with two IV's that differ on exactly 32 bits (all the MSB's of B_i, C_i, $i = 0, \ldots, 15$). An example, found using Shabal's reference implementation, with $N = 2$, is given in Appendix A.

Note that the above technique allows to find second preimages of messages of at least two blocks, and for which the first block satisfies $(C_i \oplus \Delta) - (M_i \oplus \Delta) = C_i - M_i$, $i = 0, \ldots, 15$. This equality is satisfied with probability 2^{-16} for a random message, since each of the 16 equalities is satisfied with probability $1/2$.

Why this does not work on the original version

To extend the pseudo-collision attack to the original Shabal that makes 36 rounds in its final loop, it is sufficient to find a set of indices $\mathcal{S} \in \{0, \ldots, 15\}$ for which we will set a difference Δ in $C_i, i \in \mathcal{S}$, that such that

- $i \in \mathcal{S} \Rightarrow (i + 8) \bmod 16 \in \mathcal{S}$; because we need a difference Δ in M_{i+8} for each difference in C_i, and we need that $C - M$ makes the differences vanish;
- in Table 2, each A_i is combined an *even* number of times with a C_{i+3} where $(i + 3) \bmod 16 \in \mathcal{S}$, so that differences in C cancel out in the final loop.

Below we show that such a (non-empty) $\mathcal{S}$ does not exist, and thus that we cannot apply the attack to the original Shabal.

Consider lines 9 and 10 of the algorithm of $\mathcal{P}$ on §2. The index i varies from 0 to 35, i.e., there are 36 possibilities. On the other hand, the indices for C are taken modulo 16. Note that $36 = 2 \times 16 + 4$. Hence, the first four indices appear three times as indices for C, whereas the rest (i.e., the last 12) appear twice. Let K denote the set of the first four indices, that is, $K = \{3, 4, 5, 6\}$.

We begin by showing that each C_k, $k \in K$, is combined with a different set of A_j's, $0 \leq j \leq 11$. That is, it is not possible to find distinct $k_1, k_2 \in K$ such that both C_{k_1} and C_{k_2} are combined with A_j, for some $0 \leq j \leq 11$. Consider a $k \in K$ and denote it as $k = \ell + 3$, for some $\ell \in \{0, 1, 2, 3\}$. Then, k appears as an index for C exactly when i equals ℓ, $\ell + 16$, or $\ell + 32$. Since the indices of A are taken modulo 12, these three values project to ℓ, $\ell + 4$, and $\ell + 8$ as indices of A. Now, the set $\{\ell, \ell + 4, \ell + 8 | 0 \leq \ell \leq 3\}$ contains exactly 12 (distinct) elements which span the 12 indices of A. Hence, we have shown that the elements of K split the indices of A into four disjoint sets of size three.

Recall the first condition on $\mathcal{S}$, that if $i \in \mathcal{S}$ then $(i + 8) \bmod 16 \in \mathcal{S}$, and consider an index of A of the form ℓ, $\ell \in \{0, 1, 2, 3\}$. We have ℓ as an index for A exactly when i equals ℓ, $\ell + 12$, and $\ell + 24$. These correspond to $C_{\ell+3}$, $C_{\ell+15}$, and $C_{\ell+11}$, respectively. Note that $\ell + 11 \equiv \ell + 3 \bmod 8$. As a result, $\ell + 3 \in \mathcal{S}$ if and only if $\ell + 11 \in \mathcal{S}$. Moreover, $\ell + 15 \notin \mathcal{S}$, otherwise A_ℓ would be combined with an odd number of indices from $\mathcal{S}$, which would contradict our second assumption.

Now consider an index of A of the form $\ell + 8$, $\ell \in \{0, 1, 2, 3\}$. Analogous to above, we find that $A_{\ell+8}$ is combined with $C_{\ell+11}$, $C_{\ell+7}$, and $C_{\ell+3}$, implying that $\ell + 7 \notin \mathcal{S}$.

Finally, consider an index of A of the form $\ell + 4$, $\ell \in \{0, 1, 2, 3\}$. We have $\ell + 4$ as an index for A exactly when i equals $\ell + 4$, $\ell + 16$, and $\ell + 28$. These correspond to $C_{\ell+7}$, $C_{\ell+3}$, and $C_{\ell+15}$, respectively. We have already shown that $\ell + 7 \notin \mathcal{S}$ and $\ell + 15 \notin \mathcal{S}$. Hence, we muste also have that $\ell + 3 \notin \mathcal{S}$, not to violate the even occurence condition. This itself implies that $\ell + 11 \notin \mathcal{S}$. Hence, $\mathcal{S}$ must be empty.

5 Countermeasures

The properties described above can be avoided by modifying the main loop of Shabal. We propose several strategies:

1. Make indices of C (or of M) dependent on j (although this would just partially solve the problem).
2. Invert the order of the last two instructions, to force introduction of C's differences in B.
3. Replace the multiplication by 3 by an operation that does not preserve the MSB difference.

Acknowledgments

We would like to thank the designers of Shabal for valuable comments on a preliminary draft.

References

1. Jean-Philippe Aumasson. On the pseudorandomness of Shabal's keyed permutation. Available at `http://131002.net/data/papers/Aum09.pdf`, 2009. Public comment on the NIST Hash Competition.
2. Emmanuel Bresson, Anne Canteaut, Benoît Chevallier-Mames, Christophe Clavier, Thomas Fuhr, Aline Gouget, Thomas Icart, Jean-François Misarsky, María Naya-Plasencia, Pascal Paillier, Thomas Pornin, Jean-René Reinhard, Céline Thuillet, and Marion Videau. Shabal, a submission to NIST's cryptographic hash algorithm competition. Submission to NIST, 2008.
3. Anne Canteaut. Shabal. Presentation at the First SHA-3 Conference, 2009.
4. Lars Knudsen, Krystian Matusiewicz, and Søren S. Thomsen. Observations on the Shabal keyed permutation. Public comment on the NIST Hash Competition, 2009.
5. NIST. Announcing request for candidate algorithm nominations for a new cryptographic hash algorithm (SHA-3) family. *Federal Register*, 72(112), November 2007.

A Low-weight pseudo collision for a variant of Shabal

We present a pseudo-collision for a variant of Shabal, which makes 24 iterations instead of 36 in the final loop of $\mathcal{P}$. Based on the reference code of Shabal submitted to NIST, we thus modified the implementation of $\mathcal{P}$, and determined the IV by following Shabal's preprocessing with the modified $\mathcal{P}$. To simplify the presentation, we set the most significant bytes of C to zero.

The IV of the first message, for respectively A, B, and C, is:

```
64AE5994  D3EDC8F7  71EE96E3  E7A46592
D00D9D46  4DA04965  D405A8ED  138C3D53
F181D5ED  A061EB35  582A90EE  B4F2D655

C1099CB7  07B385F3  E7442C26  CC8AD640
EB6F56C7  1EA81AA9  73B9D314  1DE85D08
48910A5A  893B22DB  C5A0DF44  BBC4324E
72D2F240  75941D99  6D8BDE82  A1A7502B

09BF68D1  08BAD750  06028CB2  0134F359
05D469D8  041A8CC2  018B2A6E  04052780
0F07D787  0194358F  0C60D665  0E97D79A
050C3434  0ED9A06D  0537DC8D  0CDB5969
```

The IV of the second message is the same, except that all the MSB's of B and C words are changed.

The first message is composed of 32 zero words, and the second message of 16 words `80000000` followed by 16 zero words. The common 512-bit digest is

```
BF3B530D1FB7A4C3736EDB7D364E967E40F0A6CB221BAB2F3232E6928167F17A
1075C320AEB407C4F7D70D4EF9DB92FA98F94B5D5379983DA019D7450F713740
```

Note that such pseudo collisions can also be found when, in the final loop of $\mathcal{P}$, 35 is replaced by any integer $12N - 1$, for an even $N \leq 2$.