ECRYPT II

ICT-2007-216676

ECRYPT II

European Network of Excellence in Cryptology II

Network of Excellence

Information and Communication Technologies

D.SYM.7
Intermediate Status Report

Due date of deliverable: 31. July 2011
Actual submission date: 5. July 2011

Start date of project: 1 August 2008 Duration: 4 years

Lead contractor: Katholieke Universiteit Leuven (KUL)

Revision 1.0

Project co-funded by the European Commission within the 7th Framework Programme		
Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission services)	
RE	Restricted to a group specified by the consortium (including the Commission services)	
CO	Confidential, only for members of the consortium (including the Commission services)	

Intermediate Status Report

Contributors
Praveen Gauravaram (DTU),
Florian Mendel (TUG),
María Naya-Plasencia (FHNW),
Vincent Rijmen (KUL),
Deniz Toz (KUL)

5. July 2011
Revision 1.0

The work described in this report has in part been supported by the Commission of the European Communities through the ICT program under contract ICT-2007-216676.

Contents

Chapter 1

Introduction

1.1 Overview

This report was produced in partial fulfillment of contract ICT-2007-216676 (ECRYPT II), sponsored by the European Commission through the ICT Programme. The information in this paper is provided as is, and no warranty is given or implied that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

We present a short overview of the recent results on the five finalists for NIST's SHA-3 competition. The next five chapters treat each one of the finalists. The chapters have the following structure.

1. Public analysis: lists all publications that appeared since D.SYM.4 (June 2010) [36], describes briefly what is in each publication without commenting on the relevance of the findings in the publication.

2. Third round tweak: provides information on the tweak applied to the hash function upon entering the 3rd round, explains the (claimed) impact on the published results.

3. Summary & conclusion discusses the status of the hash function, possibly with some subjective elements.

Except where mentioned explicitly differently, the published results are all on the versions of the finalists *before* application of the third round tweaks.

For further updates on the status of the SHA-3 finalists, we refer to the SHA-3 Zoo maintained by Symlab partners [3].

Acknowledgments.

The authors would like to thank the designers of the five finalists for answering all our questions. Additionally, we would like to thank Dan Bernstein, Anne Canteaut and Pawel Morawiecki.

1.2 Zero-sum distinguishers

Zero-sum distinguishers are a rather novel technique, that can be used to analyze hash functions. The technique has recently been applied to several of the SHA-3 finalists. In the

remainder of this chapter, we survey some recent results and mention elements that are not specific to one of the 5 finalists.

1.2.1 Zero-sum partitions for iterated permutations

Recall from [7, 36] that for a function $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$, a zero-sum of size K is a subset $\{x_1, x_2, \ldots, x_K\} \subset \mathbb{F}_2^n$ of elements that sum to zero and for which the corresponding set of images $\{F(x_1), F(x_2), \ldots, F(x_K)\}$ also sum to zero.

Boura and Canteaut present a stronger form of zero-sum distinguishers called zero-sum partitions.

Definition 1 (Zero-sum partition [16]) *Let $P : \mathbb{F}_2^n \to \mathbb{F}_2^n$ be a permutation. A $K = 2^k$ size zero-sum partition for P is a collection of 2^{n-k} disjoint zero-sums $X_i = \{x_{i,1}, \ldots, x_{i,2^k}\} \subset \mathbb{F}_n^2$. That is,*

$$(\bigcup_{i=1}^{2^{n-k}} X_i = \mathbb{F}_2^n) \wedge (\sum_{j=1}^{2^k} x_{i,j} = \sum_{j=1}^{2^k} P(x_{i,j}) = 0), \forall 1 \leq i \leq 2^{n-k}.$$

When the function F is a permutation P defined over $\mathbb{F}_2^n$, a zero-sum partition for P of size $K = 2^k$ consists of 2^{n-k} zero-sums. In other words, finding a zero-sum partition for P is equivalent to finding $(2^{n-k} - 1)$ zero-sums.

Boura and Canteaut present a general idea to construct zero-sum partitions for an iterated permutation of form $P = R_r \circ \ldots \circ R_1$ where all R_i are round transforms over $\mathbb{F}_2^n$ and when the non-linear part of the round transform, denoted χ, consists of $n_r = n/n_0$ parallel applications of a smaller SBox χ_0 defined over $\mathbb{F}_2^{n_0}$. Their idea combines the following two techniques:

1. Exploit the non-linear part of the round transform to derive an improved bound on the degree of the iterated permutation.

2. Exploit the fact that a few iterations of the round transform are not enough for providing full diffusion which leads to some *multiset* properties of the linear part of the round transform for a small number of rounds.

1.2.2 Generalization of Aumasson and Meiers' results.

For the iterated permutation $P = R_r \circ \ldots \circ R_1$, let t be an integer such that $t \in [1, r]$. Let $F_{r-t} = R_r \circ \ldots \circ R_{t+1}$ and $G_t = R_1^{-1} \circ \ldots \circ R_t^{-1}$ be the decomposed transforms of P. Boura and Canteaut [16] generalize the zero-sum partition of P based on the algebraic degree of its iterations in the form of following proposition.

Proposition 1 *Let d_1 and d_2 be such that the degree of F_{r-t} denoted $\deg(F_{r-t}) \leq d_1$ and that of G_t denoted $\deg(G_t) \leq d_2$. Let V be any subspace of $\mathbb{F}_2^n$ of dimension $d + 1$ where $d = \max(d_1, d_2)$, and let W denote the complement of V, i.e, $V \oplus W = \mathbb{F}_2^n$. Then the sets $X_a = \{G_t(a + z), z \in V\}, a \in W$ form a zero sum partition of $\mathbb{F}_2^n$ of size 2^{d+1} for the r-round permutation of P.*

Improvement of trivial bounds by using spectral properties.

Boura and Canteaut [15, 16] improve this result by a few more rounds of the permutation exploring the spectral properties of the non-linear part χ_0 of the round transform in the permutation P. In particular, they use the following theorem to improve the trivial bound of [7] when the values occurring in the Walsh spectrum of F are divisible by a high power of 2.

Theorem 1 ([18]) *Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ be a function such that all values in its Walsh spectrum are divisible by 2^ℓ, for some integer ℓ. Then for any $G : \mathbb{F}_2^n \to \mathbb{F}_2^n$, we have* $\deg(G \circ F) \leq n - \ell + \deg(G)$.

Extensions using multiset properties.

In addition to exploiting the degrees of the round transformation, Boura and Canteaut [16] also exploit the fact often a few iterations of the round transform are insufficient to achieve complete diffusion, thus leading to some *multiset* properties for a small number of round transforms. Here, extensions using one round *multset* are briefed.

Recall that the zero-sum partition due to Theorem 1 can be obtained for any choice of the subspace V. It is possible to extend the number of zero-sum partitions obtained for t rounds to $t+1$ rounds by considering the subspaces that correspond to a collection of any $\lceil (d+1)/n_0 \rceil$ rows such that $V = \oplus_{i \in \mathcal{I}} B_i$, for some set $\mathcal{I} \subset \{0, \ldots, n_r\}$ of size $\lceil (d+1)/n_0 \rceil$ where B_i for $0 \leq i < n_r$ is the n_0-dimensional subspace corresponding to the rows. As variables of different rows are not mixed after the application of χ, $\chi(a+V) = b+V$ for some b. Using this property, Boura and Canteaut prove the following proposition which finds some zero-sum partitions of size 2^{d+1} for the r-round permutation P.

Proposition 2 ([16]) *Let d_1 and d_2 be such that* $\deg(F_{r-t-1}) \leq d_1$ *and* $\deg(G_t) \leq d_2$. *Let us decompose the round transformation after t rounds into $R_{t+1} = A_2 \circ \chi \circ A_1$ where both A_1 and A_2 have degree 1. Let $\mathcal{I}$ be any subset of $\{0, \ldots, n_r - 1\}$ of size $\lceil (d+1)/n_0 \rceil$, $V = \oplus_{i \in \mathcal{I}} B_i$ and W be its complement. Then the sets*

$$X_a = \{(G_t \circ A_1^{-1})(a+z), z \in V\}, a \in W$$

form a zero-sum partition of $\mathbb{F}_2^n$ of size 2^k, with $k = n_0 \lceil d+1/n_0 \rceil$, for the r-round permutation P.

Similarly, Boura and Canteaut [16] explore *multiset* properties over two more rounds by exploiting the structure of the round transforms as well as their linear part to produce zero-sum partitions.

Finally, Boura, Canteaut and De Cannière [17] prove the following theorem:

Theorem 2 ([17]) *Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ be a function corresponding to the concatenation of m smaller SBoxes $S_1, \ldots, S_m$, defined over $F_2^{n_0}$. Let δ_k be the maximal degree of the product of any k coordinates of anyone of these smaller SBoxes. Then for any function $G : \mathbb{F}_2^n \to \mathbb{F}_2^\ell$, we have*

$$\deg(G \circ F) \leq n - \frac{n - \deg(G)}{\gamma}$$

where

$$\gamma = \max_{1 \leq i \leq n_0 - 1} \frac{n_0 - i}{n_0 - \delta_i}$$

Most notably, if all SBoxes are balanced, we have

$$\deg(G \circ F) \leq n - \frac{n - \deg(G)}{n_0 - 1}$$

Moreover, if $n_0 \geq 3$ *and all Sboxes are balanced functions of degree at most* $n_0 - 2$*, we have*

$$\deg(G \circ F) \leq n - \frac{n - \deg(G)}{n_0 - 2}$$

Chapter 2

BLAKE

2.1 Public analysis

The security of BLAKE has been the subject of several studies since June 2010. We describe them here chronologically. All of these analyses follow the techniques of differential cryptanalysis.

2.1.1 Near collisions for reduced-round compression functions (linearization)

In [42] Su et al. propose improved near-collision attacks on reduced-round variants of the compression function of BLAKE. The attacks use linearization by replacing modular additions by XORs, as in [4]. For improving previously known results as well as making them applicable to BLAKE-64, the authors loosen the restrictions on the differential pattern. They use differences of Hamming weight smaller than or equal to 2 in the intermediate states and use the path with the highest probability.

The results obtained are near-collisions for the compression function on 152 bits (for BLAKE-32), 396, and 306 bits (for BLAKE-64) on 4, 4 and 5 middle rounds respectively. The middle rounds are 7 to 10 in the case of 4 rounds and 6 to 10 in the case of 5. The time complexities of these attacks are 2^{21}, 2^{16} and 2^{216} respectively.

2.1.2 Collisions for weakened variants

The authors of BLAKE proposed several weakened variants as toy examples for cryptanalytic purposes. In [44] Vidali et al. cryptanalyse two of these variants: BLOKE and BRAKE. The variant BLOKE does not permute the message words and constants in each round of the compression function and the variant BRAKE additionally removes the feed-forward and the round constants.

The authors are able to build collisions on BLOKE for an arbitrary number of rounds exploiting the fact that all the round functions are identical. It is easy to find a message block that will produce the same output as the input for one round, and therefore in this case, for an arbitrary number of rounds r it can produce a fixed point of the rounds sequence. Using this kind of fixed-point block messages and the feed-forward, the new chaining value will not depend on the old one, but only on the salt and the counter. This can directly lead

to a collision of two messages of the same length to which we append one fixed-point message blocks with a negligible cost.

In the case of BRAKE, there is no feed-forward. Consequently, the output of the round functions is not canceled by the input, so the attack becomes a bit more complex. They also exploit the fixed points and are able to produce internal collisions, that will not lead to a collision because of the last padded block, as the used messages have different lengths.

Both analyses work for the 256- and the 512-bit variants.

2.1.3 Near collisions for reduced-round compression functions (hill-climbing)

Hill-climbing techniques are algorithms that start with an arbitrary possible solution to a problem and iteratively make small changes for improving it. In [41], Sönmez Turan and Uyan use this technique to find practical near-collisions on reduced versions of several SHA-3 candidates. One of these analyzed candidates is BLAKE-32. They use message blocks with 1 bit of difference as input and the counter and salt fixed to zero. The result is practical near-collision attacks on 209 and 184 bits for the compression function of BLAKE-32 reduced to 1.5 and 2 rounds respectively, with a time complexity of 2^{26}. For more rounds this method doesn't provide significant results.

2.1.4 Security analysis of BLAKE-32 based on differential properties

In [30], Ming et al. point out a property based on the order of using the message words in the internal function of BLAKE-32: if we have a difference in one of the threads of the internal state that is erased by a message word, it is easy to find a configuration where this non-difference is maintained over 1.5 rounds.

The authors also present some differential properties of the G function that are used in the internal function as well as of G^{-1}. They try to exploit the property of the message words distribution for building collisions or near-collisions on 6 rounds of the compression function with a meet-in-the-middle technique. The authors use the differential properties on G and G^{-1} to conclude that this is not possible.

2.1.5 Boomerang distinguishers

The boomerang attack is a method based on differential cryptanalysis introduced by Wagner in [45]. It was first introduced for analysing block ciphers. Its main principle is to consider the block cipher E as composed by two consecutive steps E_0 and E_1. The attacker can exploit then a good differential path that holds with probability p for the first step E_0, $\triangle \rightarrow \triangle^*$; and a good differential trail that holds with probability q for the second step E_1, $\bigtriangledown^* \rightarrow \bigtriangledown$, the following way:

1. Choose a pair of input messages (P_1, P_2) of difference $\triangle$, and compute $C_1 = E(P_1)$ and $C_2 = E(P_2)$.
2. With $C_3 = C_1 \oplus \bigtriangledown$ and $C_4 = C_2 \oplus \bigtriangledown$ compute $P_3 = E^{-1}(C_3)$ and $P_4 = E^{-1}(C_4)$.
3. If the differential trails are verified, $P_3 \oplus P_4 = \triangle$. This will occur with probability at least p^2q^2.

In [13], Biryukov et al. apply the boomerang attack on BLAKE-32 in a quite straightforward way. They use some high probability differential trails and the main results obtained are a distinguisher on the compression function reduced to seven rounds with complexity of 2^{232} and a distinguisher on eight rounds of the keyed permutation with a complexity of 2^{242}.

The authors point out that this results also apply to BLAKE-64, reaching about the same number of rounds.

2.1.6 Distinguishers based on iterative differentials

In [24] Khovratovich and Dunkelman propose an original approach for building distinguishers on BLAKE-32. These distinguishers use a good iterative differential path that uses two different characteristics of the G function. The probability that the path through one round holds, is 2^{-132}. The authors find solutions for three rounds with a complexity in time of 2^{62} using trail backtracking and message modification techniques. The remaining rounds are verified probabilistically.

The result is distinguishers for 4, 5 and 6 rounds of the internal permutation with complexity 2^{192}, 2^{324} and 2^{456} respectively and memory needs of 2^{354}. Also, a trade-off is proposed for reducing these memory needs, that results on an increment of 2^{32} in the time complexity.

2.1.7 Tuple cryptanalysis and permutation distinguishers

Tuple cryptanalysis has been introduced in [6] by Aumasson et al. It is a variant of structural cryptanalysis that considers ordered rather than unordered multisets: the core element used is a tuple, i.e. a list with possibly repeated elements. The authors apply this method to ARX constructions.

The best result obtained on BLAKE with the tuple cryptanalysis is a known-key distinguisher on the core permutation that works for 2.5 rounds. This distinguisher has a time complexity of 2^{32} for BLAKE-256 and of 2^{64} for BLAKE-512. The authors also suggest a possible 4-round distinguisher in the chosen-key model for rounds 3.5 to 7.5, while pointing out that this attack needs further study to be verified.

2.2 Third round tweak

The authors of BLAKE propose as a tweak for entering the third round of the SHA-3 competition to increase the number of rounds performed of the internal function for hashing each message block. In the case of BLAKE-32 and BLAKE-28 this number is increased from 10 to 14. For BLAKE-64 and BLAKE-48 the number of rounds becomes 16 (instead of 14 previously). This increases the security margin, as the result on the biggest number of rounds is 8 for both versions. Besides increasing the number of rounds, the hash function remains unchanged. Due to this, the authors can legitimately claim that previous cryptanalytic results on BLAKE remain valid. They also propose a renaming of the variants: BLAKE-h, where h is the hash size, for distinguishing the finalist version.

2.3 Summary & conclusion

BLAKE remains a secure hash funtion and no flaws on it have been showed up to now. The security margin of BLAKE seems solid.

The cryptanalysis that we have presented can be roughly classified in three categories: near-collisions, cryptanalysis of weakened variants and distinguishers. The near-collision attacks work on the compression function up to a very reduced number of rounds and it does not seem as if this could be extended much further, with the techniques used until now. The cryptanalysis of the weakened toy examples BLOKE and BRAKE are not applicable to BLAKE, mainly because of the permutation of the order of message words in each round, making it difficult to find the fixed-blocks that make the attack work. Distinguishers on the compression function are not a direct threat to the security of a hash function, but they help to understand it better. In some occasions, they are the starting point for some more dangerous attacks.

In the case of BLAKE, this kind of distinguishers does not exist on the full compression function, and the ones that have been mounted on the biggest number of rounds, are still far from the total, specially after the final round tweak. The distinguishers that reach the largest number of rounds, based on boomerangs, work up to 8 rounds.

We point out here that to our knowledge no analysis has been done on the security against (second) preimage attacks. It also seems that BLAKE-256 has been more studied than BLAKE-512. These two points may be interesting to investigate further.

Chapter 3

Grøstl

3.1 Public analysis

3.1.1 Improved Differential Attacks for ECHO and Grøstl

In [34] Peyrin introduces a new technique: *the internal differential attack*. An updated and corrected version can be found in [35]. The attack exploits the fact that the two parallel pipes, denoted by P and Q, in the compression function of Grøstl are very similar to one another. Simplifying, one could state that for certain differences (a, b) the equality

$$P(x) + Q(x + a) = b \tag{3.1}$$

holds for a comparably large number of inputs x. Recall that the Grøstl compression function operates as follows:

$$h_{t+1} = h_t + P(h_t + m_t) + Q(m_t) \tag{3.2}$$

It follows that if $h_t = a$, then $h_{t+1} = a+b$ for a comparable large number of message blocks m_t. In the real attack on Grøstl, Peyrin doesn't work with a single internal differential (a, b), but with a set of internal differentials (a truncated internal differential). The rebound technique is used to reduce the complexity of the search for a right pair.

This results in a distinguisher for the compression function of Grøstl-256, with time complexity 2^{192} and memory complexity 2^{64}, as well as a distinguisher for the compression function of Grøstl-512 reduced to 11 rounds, with time complexity 2^{640} and memory complexity 2^{64}. Peyrin describes also a collision attack for Grøstl-256 reduced to 5 rounds, with time complexity 2^{79} and memory complexity 2^{64}, as well as a collision attack for Grøstl-512 reduced to 6 rounds with time complexity 2^{177} and memory complexity 2^{64}.

3.1.2 Improved Collision Attacks on the Reduced-Round Grøstl Hash Function

In [22] Ideguchi et al. extend and improve upon the results of [34, 35] by changing the rebound part of the attack. The authors present a collision attack for Grøstl-256 reduced to 6 rounds, with time complexity 2^{112} and memory complexity 2^{32}, and a semi-free-start collision attack for Grøstl-256 reduced to 7 rounds. Also a semi-free-start collision attack on the compression function of Grøstl-256 reduced to 8 rounds is presented.

3.1.3 How to improve rebound attacks

In [33] Naya-Plasencia improves the complexities of the attacks that use rebound attack techniques given in [34, 35]. The author improves the merging of large lists by using some additional observations and by using a better list merging algorithm. The main idea is to do a sieving so that one does not have to try all the elements in one list with all the elements in the other(s) at the merging and filtering steps.

This results in a distinguisher for the compression function of Grøstl-256, with time complexity 2^{182} and memory complexity 2^{64}, as well as a distinguisher for the compression function of Grøstl-512 reduced to 11 rounds, with time complexity 2^{630} and memory complexity 2^{64}.

3.1.4 New non-ideal properties of AES-based permutations: applications to ECHO and Grøstl

In [38], Sasaki et al. introduce *non-full-active Super-Sbox analysis* which can detect non-ideal properties of a class of AES-based permutations with a complexity lower than previously known. By considering differential paths with a lower number of active S-boxes in some of the Super-Sboxes (no state with all bytes active), the memory complexity of a differential attack can be reduced.

This results in a distinguisher for the permutation used in the compression function of Grøstl-256 reduced to 8 rounds, with time complexity 2^{48} and memory complexity 2^{8}, as well as a semi-free-start collision attack on the compression function of Grøstl-512 reduced to 7 rounds, with time complexity 2^{152} and memory complexity 2^{56}.

3.1.5 Updated Differential Analysis of Grøstl

In [39, 40], Schläffer updates the security analysis available on Grøstl, taking into account the tweaks described in [21]. Most importantly, he investigates the impact of the tweak on the rebound attacks. The paper includes a semi-free-start collision attack on the compression function of Grøstl-256 reduced to 6 rounds, with a time complexity of 2^{112} and a memory complexity of 2^{64}, as well as a semi-free-start collision attack on the compression function of Grøstl-512 reduced to 6 rounds, with a time complexity of 2^{180} and a memory complexity of 2^{64}. Collision attacks on the hash function are given for versions reduced to 3 rounds.

3.2 Third round tweak

The tweaks are described in [21]. The results published on Grøstl convinced the designers to make the internal permutations P and Q more different from one another. The tweak achieves this by two modifications:

1. The shift values for the Q transformations are changed.
2. The round constants in both the P and the Q transformations are changed.

3.2.1 New shift values for Q

For Grøstl-256:

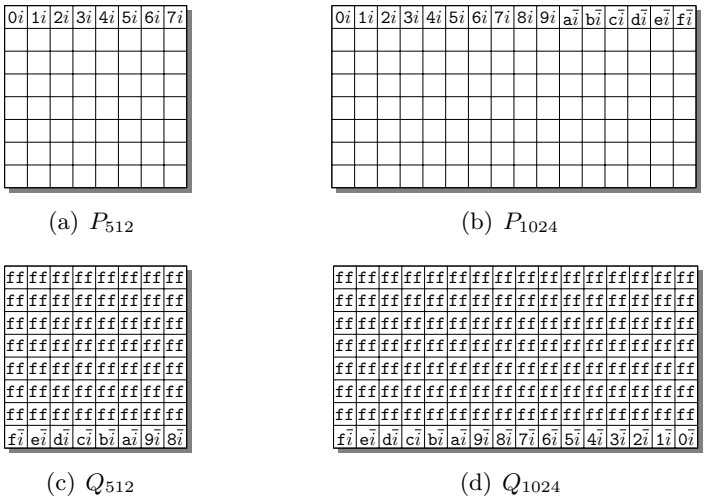
row	0	1	2	3	4	5	6	7
offset	1	3	5	7	0	2	4	6

For Grøstl-512:

row	0	1	2	3	4	5	6	7
offset	1	3	5	11	0	2	4	6

3.2.2 New round constants

The new round constants are shown in Figure 3.1.



(a) P_{512} (b) P_{1024}

(c) Q_{512} (d) Q_{1024}

Figure 3.1: The new round constants in Grøstl [40].

3.3 Summary & conclusion

Among the five finalists, Grøstl is probably the design that has been analysed the most often for its resistance against rebound attacks. Although its security margin against this attack appears to be the smallest among the five finalists, we can be relatively confident that the security margin against rebound attacks, or any other attack based on truncated differentials, will not change soon.

Grøstl is also the design that follows the most the design strategy of AES. This may give an additional good feeling about its security margin, although the majority of the results of security analysis on AES can of course not be translated directly to the hash function setting.

Chapter 4

JH

4.1 Public analysis

This section includes recent analysis results on JH that has been done after June 2010.

4.1.1 Rebound attacks

In [33] Naya-Plasencia improves the complexities of the attacks that use rebound attack techniques given in [37]. The author improves the merging of large lists by using some additional observations and by using a better list merging algorithm. The main idea is to do a sieving so that one does not have to try all the elements in one list with all the elements in the other(s) at the merging and filtering steps.

The result is 2^{97} in time and memory for the 16 round semi-free-start collision on hash function JH and 2^{96} in time and memory for 1008-bit semi-free start near-collision for 19 rounds and 896-bit semi-free-start near-collision for the 22 rounds of the compression function of JH.

4.1.2 Practical Near-Collisions

In [28] Turan et al. present near-collisions for up to 10 rounds of the compression function of JH, F_d, by using a simple hill climbing method. In the attack they use two 512-bit random messages (M_1, M_2) with 1-byte difference, without considering the padding block. They try to minimize the function

$$f_{M1,M2}(CV) = h_w(F_d(M_1, CV) \oplus F_d(M_2, CV))$$

where $CV \in \{0,1\}^n$ is the chaining value that is to be optimized and h_w is the Hamming weight. The best result presented in the paper is for 10 rounds: they obtain a 820-bit near-collision with a complexity of $2^{23.24}$.

4.2 Third round tweak

JH has been tweaked for the final round of the SHA-3 competition. The number of rounds of the compression function is increased from 35.5 to 42. The reasoning for the tweak is given as follows: The last half round is removed for better hardware efficiency and the number of

rounds is changed to 42 to increase the security margin. This new version may be referred to as JH42 to distinguish it from the original design.

4.3 Summary & conclusion

The cryptanalytic results found so far on reduced-round versions of JH don't represent a threat on the security hash function: The results given in [28] are practical, but it is possible to attack only a relatively small number of rounds. Rebound attacks also seem quite powerful in the compression function setting due to the simplicity of the round function. Therefore, it is possible and easier to analyse more number of rounds but the complexities is much higher. Moreover, current attacks on the compression function of JH cannot be extended to a collision attack on the hash function, since JH process an additional block during the padding algorithm.

JH didn't get much attention so far, therefore there is not many analysis results published. Hence whether the existing attacks can be improved or other cryptanalysis methods can be applied to JH is an open question for the moment.

Chapter 5

Keccak

Recall that Keccak-$[r, c]$ refers to Keccak with r-bit bitrate and c-bit capacity, Keccak-f refers to the underlying permutations of Keccak and Keccak-$f[b]$ refers to the permutation with a width of size b bits (e.g., $b = 1600$).

5.1 Public analysis

5.1.1 Preimage attacks on the weaker versions of Keccak by using SAT solvers

Morawiecki and Srebrny [31] analyse the security of versions of Keccak with a very small number of rounds by using satisfiability problem (SAT) solver algorithms. SAT solvers are typically used to solve decision problems described in Conjunctive Normal Form (CNF). Cryptographic algorithms that need to be analysed using SAT solvers are first represented in CNF form and then this CNF is passed as an input to an efficient SAT solver for a solution.

Let H be the reduced-round hash function derived from Keccak. Morawiecki and Srebrny consider the preimage attack on H to find short input length message p where $|p| \in \{24, 32, 40\}$ bits for a target hash value h. This means that H has only one application of the permutation Keccak-f as the input p can be properly padded with the padding bits, denoted by pad, such that the message $m = p \| pad$ fits in a single block of H. Morawiecki and Srebrny use the tools CryptLogVer [32] and PrecoSAT [2] to find a preimage for H as follows:

1. Generate CNF equivalent of the algorithm.
   (a) Develop a Hardware Description Language (HDL) code of H.
   (b) Use the first application of CryptLogVer called Quartus II, a software tool produced by Altera, to analyse and synthesize HDL.
   (c) Use a built-in functionality of Quartus II to obtain Boolean expressions of H compiled in HDL.
   (d) Use the second application of CryptLogVer to convert Boolean expressions to CNF.
2. Set pad bits and the target hash value h of H.
3. Pass the CNF form to PrecoSAT solver for a solution which is p such that $H(p \| pad) = h$.

The authors carry out the above attack on the reduced versions of Keccak with varying number of rounds, state sizes and message lengths. They performed experiments on a 4-core Intel Xeon 2.5 GHz which was a part of Grid'5000 system [1] where only one core was used. (The PrecoSAT tool desn't lend itself to parallel processing.) For the version Keccak-$[1024, 576]$ which outputs a 1024-bit hash value, preimages were found for upto 3 rounds in time (evaluated in seconds) better than the exhaustive search time which checks all the combinations of unknown message bits. It was observed that the times of the attack on the version Keccak-$[80, 120]$ tested for 3 rounds and the version Keccak-$[24, 26]$ tested for 3,4 and 5 rounds are not any better than the exhaustive search time. In addition, the attack fails for the versions Keccak-$[1024, 576]$ and Keccak-$[80, 120]$ for beyond 3 rounds because the PrecoSAT solver was not able to find a solution.

5.1.2 Second preimage attacks for up to 8 rounds of Keccak

Bernstein [8] describes a second preimage attack for Keccak instantiated with 6, 7 and 8 rounds of the permutation Keccak-f by exploiting the low algebraic degree 2 of the round function of Keccak-f. However, Bernstein himself claims that *"..., for people who have even the slightest understanding of the physical reality of attack cost, this attack makes no sense: it's inherently memory-intensive and communication-intensive, and for the same machine size it's clearly much slower than parallel exhaustive search, even though it has somewhat fewer bit operations."* Therefore, we omit here a description of this attack.

5.1.3 Zero-sum distinguishers for Keccak-$f[1600]$ permutation

The text in this section and the following relies heavily on the material present in Section 1.2 In the previous report [36], zero-sum distinguishers for 16 and 18 rounds of the Keccak-$f[1600]$ permutation were briefed based on the analyses of Aumasson and Meier [7] and Boura and Canteaut [15] respectively. It should also be recalled that the result of [15] prompted the designers of Keccak to increase the number of rounds of the Keccak-$f[1600]$ permutations from 18 to 24 in order to preserve the hermetic sponge property of Keccak [9].

Zero-sum partitions for reduced round Keccak-$f[1600]$.

Boura and Canteaut present zero-sum distinguishers on 20 out of 24 rounds of the Keccak-$f[1600]$ permutation [16]. In fact, they show zero-sum partitions on 20 rounds of Keccak-$f[1600]$. The previous techniques that produced zero-sum distinguishers for 16 and 18 rounds of Keccak-$f[1600]$ also produce zero-sum partitions [16].

Generalization of Aumasson and Meiers' results. The zero-sum partitions for 16 rounds of Keccak-$f[1600]$ derived by Aumasson and Meier follow from Proposition 1 by choosing for V a subspace which is generated by $(d + 1)$ elements of the canonical basis, where $d = \max(\deg(F_{r-t}), \deg(G_t))$. The degree of Keccak-$f[1600]$ after 10 rounds is at most $2^{10} = 1024$ and its inverse after 6 rounds is at most $3^6 = 729$. Therefore, by choosing $t = 6$ many zero sum partitions of size 2^{1025} can be found.

Improvement of trivial bounds by using spectral properties. Boura and Canteaut [15, 16] improve this result by a few more rounds of the permutation exploring the spectral properties of the non-linear part χ_0 of the round transform in the permutation P. They note

that all elements in the Walsh spectrum of the non-linear permutation χ_0 are divisible by 2^3. Since the Walsh spectrum of χ_0 and its inverse χ_0^{-1} are the same, the Walsh spectrum of χ_0^{-1} is also divisible by 2^3. As there are $n_r = 320$ applications of χ_0, Walsh spectra of R and R^{-1} applied on the 1600-bit state of Keccak-f[1600] are divisible by $2^{3\times 320} = 960$. Since 6 rounds of inverse R has a degree of at most 729, application of Theorem 1 on 7 inverse rounds of R a maximum degree of $1600 - 960 + 729 = 1369$. This bound allows to find zero sum partitions of size 2^{1370} for 17 rounds of Keccak-f[1600] by choosing $t = 7$.

Extensions using multiset properties. Proposition 2 can be applied to Keccak-f[1600] by choosing $V = \oplus_{i\in\mathcal{I}} B_i$ where $\mathcal{I}$ is any collection of 274 rows to produce zero-sum partitions of size 2^{1370} for 18 rounds of Keccak-f[1600]. Using multiset properties over two more rounds of Keccak-f[1600] leads to 64 zero-sum partitions of size 2^{1461} for 19 rounds and 64 zero-sum partitions of size 2^{1586} for 20 rounds.

5.1.4 Zero-sum partitions for full Keccak-f[1600]

In the round transform of Keccak-f[1600], $R = \iota \circ \chi \circ \pi \circ \rho \circ \theta$. Let $A_1 = \pi \circ \rho \circ \theta$ which is linear and let $A_2 = \iota$ which is an addition of a constant value. Therefore, the composed linear layer $A_1 \circ A_2$ can be defined as $L = \pi \circ \rho \circ \theta$. The non-linear function χ is equivalent to 320 parallel applications of the SBox χ_0 and χ^{-1} has a degree 3. Application of Theorem 2 to the round transform R of Keccak-f[1600] leads to the following bounds for any function F:

$$\deg(F \circ R) = \mathrm{F} \circ \chi \leq n - \frac{n - \deg(F)}{3}$$

$$\deg(F \circ R^{-1}) = \deg((F \circ L^{-1}) \circ \chi^{-1}) \leq n - \frac{n - \deg(F)}{3}$$

For instance, the bound on the degree for 11 forward rounds of R is $1600 - \frac{1600-1024}{3} = 1408$ and for 12 forwards rounds of R is $1600 - \frac{1600-1408}{3} = 1536$. Similarly, the bound on the degree for 7 inverse rounds of R is $1600 - \frac{1600-729}{3} = 1309$ and for 8 inverse rounds of R is $1600 - \frac{1600-1309}{3} = 1503$. Using these bounds, by choosing any subspace in $\mathbb{F}_2^{1600}$ corresponding to a collection of 318 rows after the layer L in the 11th round of Keccak-f[1600] produces the sets that form a zero-sum partition of size 2^{1590}.

Improved Zero-sum partitions for full Keccak-f[1600]. Duan and Lai [19] show an improved zero-sum partition for the full Keccak-f[1600] permutation by observing that the product of any two components of χ^{-1} has degree at most 3 instead of 4 as noted by Boura and Canteaut [16] (and used by Boura et al. [17]). This observation implies that in Theorem 2, $\delta_2 = 3$ and hence, $\gamma = 2$. Duan and Lai use this property of χ^{-1} to present an improved upper bound for the degree of $F \circ R^{-1}$ for any F as given below:

$$\deg(F \circ R^{-1}) = \deg((F \circ L^{-1}) \circ \chi^{-1}) \leq n - \frac{n - \deg(F)}{2}.$$

Using this result, improved bounds for the degrees of the inverse of 7 to 15 rounds can be derived for Keccak-f[1600]. For instance, the inverse of 7, respectively 8 rounds has a degree of 1164, respectively 1382 instead of 1309, respectively 1503, due to Boura et al. [17]. Similarly, the inverse of 11 rounds has a degree of at most 1572 instead of 1596. Hence, by choosing the

intermediate states after the L layer on the 12th round of Keccak-f[1600] in any subspace V corresponding to a collection of 315 rows, zero-sum partition of size 2^{1575} can be constructed for full 24-round Keccak-f[1600].

Although the zero-sum partition distinguishers on the Keccak-f[1600] permutation by Boura et al. [17] and Duan and Lai [19] do not produce any distinguishers for Keccak itself, these results contradict the hermetic sponge design strategy of Keccak. Hence, Keccak-f is not free from structural distinguishers as initially put forward by the Keccak designers in the hermetic sponge strategy

5.2 Third round tweak

The following tweaks were applied to Keccak after it was selected for the final round of the SHA-3 hash function competition. The designers of Keccak revised their documents for the third round of the competition [11, 12] and also submitted a new document on cryptographic sponge functions [10].

1. The padding technique for Keccak has been shortened and simplified. The new padding rule appends a 1 bit, a sufficient number of 0 bits and finally a 1 bit such that the length of the message is a multiple of the block length. This padded technique is called multi-rate padding [11] as it is suitable for a family of sponge functions sharing the same permutation with different rate-capacity pairs. Designers note that the new padding rule is more efficient than the previous one as it appends down to 2 bits instead of the at least 25 bits in the previous padding rule. For long messages, the efficiency gain is negligible, but short messages can be 3 bytes longer for the same number of calls to Keccak-f[1600] This rule appends at most the number of bits in a block plus one.

2. The diversification parameter d present in the previous version of Keccak is removed for the final round. Recall that this parameter was originally proposed because a protocol based on a hash function might require different instances of the hash function for different output lengths instead of the same hash function and this parameter was intended to diversify between different instances of a hash function.

   Instead of having a separate parameter for this purpose, designers note that diversification between different hash function instances from the same underlying permutation (independent of whether they produce the same or different output length) can be established by using a *domain separation* technique. Considering that the underlying construction is secure, the derived functions can be treated as independent functions. For example, *domain separation* can be implemented by appending or prepending different constants to the input of each instance of hash function [12]. Different instances of hash functions, denoted H_i, based on the Keccak-f[1600] permutation and that process a message M can be proposed as H_i =Keccak($M\|C_i$) or H_i =Keccak($C_i\|M$) where C_i are constants for each instance of the hash function.

3. The restriction that bitrate r can only take values that are multiple of 8 bits, is removed for the final round version of Keccak. The new version supports values between 1 and the size of the permutation b.

There are no changes to the permutation function of Keccak. The published results on the analysis of Keccak still apply to the tweaked version of Keccak as these results do not depend on the design features of Keccak that are changed for the final round.

5.3 Summary & conclusion

So far, Bernstein's (second) preimage attacks on reduced Keccak [8] and zero-sum partitions for the full Keccak-f[1600] permutation first by Boura *et al.* [16] and later by Duan and Lai [19] can be considered as the best known analytical results on Keccak. While these results are interesting, they are far from posing any reasonable attack on Keccak. Firstly, Bernstein's (second) preimage attack on 8 rounds of Keccak has a complexity much closer to the brute force attack complexity and moreover, it is far from attacking even half the rounds of the algorithm. Secondly, zero-sum partitions on Keccak-f[1600] do not contradict the flat sponge claim of Keccak as they require a work factor which is significantly higher than 2^{800} queries to the permutation or its inverse [11]. In addition, in the zero-sum partition distinguishers the attacker has complete control over the choice of the input bits to the permutation which is not possible when it has to attack Keccak. Hence, it can be concluded that there are no serious attacks on Keccak and current analytical techniques do not seem to really help in achieving some good attacks even on the reduced versions of Keccak or its underlying permutation Keccak-f .

A possible avenue for further research on Keccak may be the construction of distinguishers for the Keccak-f permutations such that flat sponge claim can be contradicted. Such analysis on the reduced round versions of Keccak-f permutations is an encouraging sign towards the analysis of Keccak.

Chapter 6

Skein

6.1 Public analysis

This section lists public analysis appeared for the hash function since D.SYM.4. We briefly describe what is in each publication and summarize the main results. For a detailed description we refer to the original publications.

6.1.1 Pseudo-Linear Approximations for Threefish

McKay and Vora present an attack on round-reduced Threefish in [27], using pseudo-linear functions to analyze the block cipher. The attack is inspired by linear cryptanalysis. The main idea is to consider larger groupings of contiguous bits (referred to as windows) instead of single bits. The authors consider two operations on the windows (of size w): bitwise exclusive-or and addition modulo 2^w. While no approximation is needed for bitwise exclusive-or, addition modulo 2^n on the window is approximated by addition modulo 2^w. This gives a perfect approximation if the carry into the window is estimated correctly. Using these simple non-linear approximations (that are pseudo-linear as they are composed of exclusive-or and addition modulo 2^w) the authors show attacks on round-reduced Threefish. In detail, they show a key recovery attack on 11 rounds of Threefish-256 (without whitening) using an 8 round approximation and a key recovery attack on 15 rounds of Threefish-512 (without whitening) using a 12-round approximation.

6.1.2 Tuple Cryptanalysis of Threefish

Aumasson et al. introduced tuple cryptanalysis in [6]. It is a variant of integral cryptanalysis with applications to ARX primitives - composed of only three operations: additions, rotations, and exclusive-ors. The work has been inspired by the attack of Biryukov and Shamir in [14] on the SASAS structure. First the impact of modular addition, rotation, and exclusive-or on the properties of the tuple is analyzed. Then the order of the elements in the tuple is considered to improve the results. Applying tuple cryptanalysis to Threefish using an inside-out approach results in distinguishing attacks on Threefish-512 for 9 rounds and Threefish-1024 for 12 rounds in the known-key setting. Furthermore, by using the chosen-key setting the attack can be extended to 20 rounds of Threefish-512.

6.1.3 Statistical Analysis of Skein (Cube Tester)

Kaminsky analyzes the statistical properties of the Skein-512 hash function in [23]. Cube tests, invented by Aumasson et al. in [5], are used to evaluate/probe the internal polynomial structure of Skein-512 for a large number of choices of the input variables. The cube test data were calculated on a cluster with 40 cores. In total 3,603,992,046,760 Skein-512 computations were performed. The data are then subjected to three statistical tests (balance, independence, and off-by-one test) to disprove the null hypothesis that the hash function is a random polynomial. While the balance and off-by-one tests don't find nonrandom behavior, the independence test do find nonrandom behavior in Skein-512 which disproves the null hypothesis that the hash function is a random polynomial.

6.1.4 Near-Collisions for the Compression Function

Su et al. present free-start near-collisions for 24 rounds of the Skein compression function in [43] and [42], respectively. In order to find good differential trails, the authors use a linear approximation of the compression function of Skein. Therefore, all the modular addition are replaced by bitwise exclusive-or. Using the linear approximation the authors search for differential trails with low Hamming weight, since these trails are expected to result in a lower attack complexity. The results are free-start near-collisions for 24 rounds of Skein-256, Skein-512 and Skein-1024. It is important to note that all these trails use differences in the tweak input to the compression function to get 8 rounds in the middle of the differential trail with no differences. Using message modification techniques in the first rounds results in attacks for the Skein-256, Skein-512, and Skein-1024 compression functions with complexities of 2^{60}, 2^{230}, and 2^{395}.

6.1.5 Near-Collision for the Compression Functions

In [47, 46] Yu et al. present a free-start near-collision for Skein-256 reduced to 32 rounds. The main idea of the attack is to combine two short differential trails into a long differential trail exploiting the non-linear properties of the modular addition and to place the expansive/dense part of the differential trail in the middle (rounds 16-24). The result is a free-start near-collision for 32 rounds of Skein-256. Using message modification techniques in rounds 16-24 results in an attack complexity of 2^{105}. Similar as in the attack on 24 rounds by Su et al. differences in the tweak are used to get a differential trail that has 8 consecutive rounds with no differences.

6.1.6 Rotational Rebound Attacks on Reduced Skein

Rotational rebound attacks are proposed by Khovratovich et al. in [26]. They combine the rebound attack[29] with rotational cryptanalysis of Threefish [25]. In the previous work on Threefish in [25] Khovratovich and Nikolic used the fact that Threefish is an ARX primitive - it is composed of only three operations: additions, rotations, and exclusive-ors - and each of these operations preserve the rotational property with a high probability. Furthermore, the constant C_5 used in Threefish is rotational, i.e. $C_5 = C_5 \lll 2$, while the round counters have low Hamming weight. All these facts allow the authors to launch a rotational attack on Threefish. To cancel the effect of the additions of the round counters they introduce corrections in the last 4 bits of each word of the key. The results are rotational distinguishers

for 39 and 42 rounds of Threefish-256 and Threefish-512. By combining this idea with the rebound attack the authors show how to construct a rotational collision for 53 and 57 rounds of the Skein-256 and Skein-512 compression functions leading to a distinguisher (rotational q-collision) with complexity of 2^{251} and 2^{503}, respectively.

6.2 Third Round Tweak

Like all the other finalists also Skein has been tweaked for the third round. The only change is in the key schedule constant of Threefish. The old constant

$$C_5 = \texttt{0x5555555555555555}$$

is replaced by

$$C_{240} = \texttt{0x1BD11BDAA9FC1A22}\ .$$

This change does significantly decrease the efficiency of the rotational distinguisher attack. By replacing C_5 by C_{240}, which is not rotational, the attacker has to provide corrections for 64-bit values instead of for only 4-bit values resulting from the round counters. This reduces the number of attacked rounds significantly and increases the security margin of Skein.

6.3 Summary & Conclusion

Most cryptanalysis of Skein mainly focus on the block cipher Threefish, only a few results are known for the compression or hash function. The best known analysis of the Round 1 and Round 2 version of Skein and Threefish uses rotational cryptanalysis, but due to the change of the constant in the key schedule of Threefish in round 3 the efficiency of this attack is significantly reduced. However, most of the other analysis of round 2 of Skein and Threefish also applies for round 3. This includes the free-start near-collision attack of Su et al., the pseudo-linear approximations for Threefish by McKay and Vora, and probably also the statistical analysis of Skein (using cube tester) by Kaminsky. There is only one known analysis targeting round 3 of Skein done by Yu et al. They show a free-start near-collision for 32 rounds of the Skein-256 compression function which improves the previous attack by Su et al. by 8 rounds. So far no attacks have been published for the Skein hash function.

Considering that Skein has 72 rounds and that the best attack is for only 32 rounds of the compression function, one can conclude that Skein offers a large security margin against known attacks. However, it still remains an open and interesting research problem to prove bounds for Skein against differential and linear attacks.

Bibliography

[1] Grid'5000, available at `www.grid5000.fr`

[2] PrecoSAT, available at `http://fmv.jku.at/precosat/#background`

[3] The sha-3 zoo, `http://ehash.iaik.tugraz.at/wiki/The_SHA-3_Zoo`

[4] Aumasson, J.P., Guo, J., Knellwolf, S., Matusiewicz, K., Meier, W.: Differential and invertibility properties of BLAKE. In: Fast Software Encryption. Lecture Notes in Computer Science, vol. 6147, pp. 318–332. Springer (2010)

[5] Aumasson, J.P., Dinur, I., Meier, W., Shamir, A.: Cube Testers and Key Recovery Attacks on Reduced-Round MD6 and Trivium. In: Dunkelman [20], pp. 1–22

[6] Aumasson, J.P., Leurent, G., Meier, W., Mendel, F., Mouha, N., Phan, R.C.W., Sasaki, Y., Susil, P.: Tuple cryptanalysis of ARX with application to BLAKE and Skein, eCRYPT II Hash Workshop 2011, 19-20 May 2011

[7] Aumasson, J.P., Meier, W.: Zero-sum distinguishers for reduced keccak-f and for the core functions of luffa and hamsi (2009), presented at the rump session of CHES 2009

[8] Bernstein, D.: Second preimages for 6 (7? (8??)) rounds of Keccak? Initial analysis was posted to the NIST hash function forum on November 28, 2010 and improved analysis was posted to the same forum on December 2 (2010), available at `http://ehash.iaik.tugraz.at/uploads/6/65/NIST-mailing-list_Bernstein-Daemen.txt` .

[9] Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: Note on zero-sum distinguishers for keccak-f (2010), available at `http://keccak.noekeon.org/NoteZeroSum.pdf`

[10] Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: Cryptographic sponge functions. Submission to NIST Round 3 (2011), available at `http://sponge.noekeon.org/CSF-0.1.pdf`

[11] Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: The KECCAK Reference. Submission to NIST (Round 3) (2011), available at `http://keccak.noekeon.org/Keccak-reference-3.0.pdf`

[12] Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: The KECCAK SHA-3 Submission. Submission to NIST (Round 3) (2011), available at `http://keccak.noekeon.org/Keccak-submission-3.pdf`

[13] Biryukov, A., Nikolić, I., Uyan, A.R.: Boomerang attacks on BLAKE-32. In: Fast Software Encryption. Lecture Notes in Computer Science, Springer (2011), to appear.

[14] Biryukov, A., Shamir, A.: Structural Cryptanalysis of SASAS. J. Cryptology 23(4), 505–518 (2010)

[15] Boura, C., Canteaut, A.: A zero-sum property for the Keccak-f permutation for 18 rounds. Submitted to the NIST's hash function email list (2010), available at `http://eprint.iacr.org/2010/589`

[16] Boura, C., Canteaut, A.: Zero-Sum Distinguishers for Iterated Permutations and Application to Keccak-f and Hamsi-256. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) Selected Areas in Cryptography. Lecture Notes in Computer Science, vol. 6544, pp. 1–17. Springer (2010)

[17] Boura, C., Canteaut, A., De Cannière, C.: Higher-order differential properties of Keccak and Luffa. Presented at FSE 2011. Cryptology ePrint Archive, Report 2010/589 (2010), available at `http://eprint.iacr.org/2010/589`

[18] Canteaut, A., Videau, M.: Degree of composition of highly nonlinear functions and applications to higher order differential cryptanalysis. In: Knudsen, L.R. (ed.) Advances in Cryptology - EUROCRYPT 2002. Lecture Notes in Computer Science, vol. 2332, pp. 518–533. Springer (2002)

[19] Duan, M., Lai, X.: Improved zero-sum distinguisher for full round Keccak-f permutation. Cryptology ePrint Archive, Report 2011/023 (2011), available at `http://eprint.iacr.org/2011/023`

[20] Dunkelman, O. (ed.): Fast Software Encryption, 16th International Workshop, FSE 2009, Leuven, Belgium, February 22-25, 2009, Revised Selected Papers, Lecture Notes in Computer Science, vol. 5665. Springer (2009)

[21] Gauravaram, P., Knudsen, L.R., Matusiewicz, K., Mendel, F., Rechberger, C., Schläffer, M., Thomsen, S.S.: Tweaks on Grøstl, `http://www.groestl.info/Round3Mods.pdf`

[22] Ideguchi, K., Tischhauser, E., Preneel, B.: Improved collision attacks on the reduced-round Grøstl hash function. In: Burmester, M., Tsudik, G., Magliveras, S.S., Ilic, I. (eds.) ISC. Lecture Notes in Computer Science, vol. 6531, pp. 1–16. Springer (2010)

[23] Kaminsky, A.: Cube Test Analysis of the Statistical Behavior of CubeHash and Skein. Cryptology ePrint Archive, Report 2010/262 (2010), `http://eprint.iacr.org/`

[24] Khovratovich, D., Dunkelman, O.: Iterative differentials, symmetries and message modification in BLAKE-256, eCRYPT II Hash Workshop 2011, 19-20 May 2011

[25] Khovratovich, D., Nikolić, I.: Rotational Cryptanalysis of ARX. In: Hong, S., Iwata, T. (eds.) FSE. Lecture Notes in Computer Science, vol. 6147, pp. 333–346. Springer (2010)

[26] Khovratovich, D., Nikolić, I., Rechberger, C.: Rotational Rebound Attacks on Reduced Skein. In: Abe, M. (ed.) ASIACRYPT. Lecture Notes in Computer Science, vol. 6477, pp. 1–19. Springer (2010)

[27] McKay, K.A., Vora, P.L.: Pseudo-Linear Approximations for ARX Ciphers: With Application to Threefish. Cryptology ePrint Archive, Report 2010/282 (2010), `http://eprint.iacr.org/`

[28] Meltem Sönmez Turan, E.U.: Practical Near-Collisions for Reduced Round Blake, Fugue, Hamsi and JH. Second SHA-3 Candidate Conference (2010), http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/Aug2010/documents/papers/TURAN_Paper_Erdener.pdf

[29] Mendel, F., Rechberger, C., Schläffer, M., Thomsen, S.S.: The Rebound Attack: Cryptanalysis of Reduced Whirlpool and Grøstl. In: Dunkelman [20], pp. 260–276

[30] Ming, M., Qiang, H., Zeng, S.: Security Analysis of BLAKE-32 Based on Differential Properties. In: International Conference on Computational and Information Sciences. pp. 783–786 (2010)

[31] Morawiecki, P., Srebrny, M.: A SAT-based preimage analysis of reduced KECCAK hash functions. Cryptology ePrint Archive, Report 2010/285 (2010), available at http://eprint.iacr.org/2010/285.pdf

[32] Morawiecki, P., Srebrny, M., Srebrny, M.: CryptLogVer (2010), available at http://www.pawelmorawiecki.pl/index.php?option=com_content&view=article&id=48&Itemid=53

[33] Naya-Plasencia, M.: How to improve rebound attacks. Cryptology ePrint Archive, Report 2010/607 (2010), http://eprint.iacr.org/

[34] Peyrin, T.: Improved differential attacks for ECHO and Grøstl. In: Rabin, T. (ed.) CRYPTO. Lecture Notes in Computer Science, vol. 6223, pp. 370–392. Springer (2010)

[35] Peyrin, T.: Improved differential attacks for ECHO and Grøstl. Cryptology ePrint Archive, Report 2010/223 (2010), http://eprint.iacr.org/

[36] Rechberger, C., Bjørstad, T.E., Daemen, J., De Cannière, C., Gauravaram, P., Khovratovich, D., Meier, W., Nad, T., Nikolić, I., Robshaw, M., Schläffer, M., Thomsen, S.S., Tischhauser, E., Toz, D., Assche, G.V., Varıcı, K.: D.SYM.4 SHA-3 Design and Cryptanalysis Report. ECRYPT II (2010), available at http://www.ecrypt.eu.org/documents/D.SYM.4.pdf

[37] Rijmen, V., Toz, D., Varıcı, K.: Rebound attack on reduced-round versions of JH. In: FSE. LNCS, vol. 6147, pp. 286–303. Springer (2010), http://www.cosic.esat.kuleuven.be/publications/article-1431.pdf

[38] Sasaki, Y., Li, Y., Wang, L., Sakiyama, K., Ohta, K.: New non-ideal properties of AES-based permutations: Applications to ECHO and Grøstl. Second SHA-3 Candidate Conference (2010), http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/Aug2010/documents/papers/SASAKI_ECHOanalysisFinal.pdf

[39] Schläffer, M.: Updated differential analysis of Grøstl. Available online (2010), http://groestl.info/groestl-analysis.pdf

[40] Schläffer, M.: Cryptanalysis of AES-based hash functions. PhD thesis (2011)

[41] Sönmez Turan, M., Uyan, E.: Near-collisions for the reduced round versions of some second round SHA-3 compression functions using hill climbing. In: INDOCRYPT. Lecture Notes in Computer Science, vol. 6498, pp. 131–143. Springer (2010)

[42] Su, B., Wu, W., Wu, S., Dong, L.: Near-Collisions on the Reduced-Round Compression Functions of Skein and BLAKE. Cryptology ePrint Archive, Report 2010/355 (2010), `http://eprint.iacr.org/`

[43] Su, B., Wu, W., Wu, S., Dong, L.: Near-Collisions on the Reduced-Round Compression Functions of Skein and BLAKE. In: Heng, S.H., Wright, R.N., Goi, B.M. (eds.) CANS. Lecture Notes in Computer Science, vol. 6467, pp. 124–139. Springer (2010)

[44] Vidali, J., Nose, P., Pasalic, E.: Collisions for variants of the BLAKE hash function. In: Information Processing Letters. vol. 110 (2010)

[45] Wagner, D.: The Boomerang Attack. In: Fast Software Encryption. Lecture Notes in Computer Science, vol. 1636, pp. 156–170. Springer (1999)

[46] Yu, H., Chen, J., Jia, K., Wang, X.: Near-Collision Attack on the Step-Reduced Compression Function of Skein-256, eCRYPT II Hash Workshop 2011, 19-20 May 2011

[47] Yu, H., Chen, J., Jia, K., Wang, X.: Near-Collision Attack on the Step-Reduced Compression Function of Skein-256. Cryptology ePrint Archive, Report 2011/148 (2011), `http://eprint.iacr.org/`