



ICT-2007-216676

ECRYPT II

European Network of Excellence in Cryptology II

Network of Excellence

Information and Communication Technologies

D.SYM.9

New developments in symmetric key cryptanalysis

Due date of deliverable: 20. December 2011

Actual submission date: 20. December 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)	

New developments in symmetric key cryptanalysis

Editor

Svetla Nikova (K.U. Leuven)

Contributors

Andrey Bogdanov (K.U. Leuven)

Simon Knellwolf (FHNW),

Willi Meier (FHNW),

Nicky Mouha (K.U. Leuven),

Vesselin Velichkov (K.U. Leuven)

20. December 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. The information in this document 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.

Contents

1	Cube Attacks and Other Higher Order Differential Techniques	3
1.1	Cube Attacks	3
1.1.1	Terminology	4
1.1.2	Cube Attacks on Trivium	5
1.2	Cube Testers	5
1.2.1	Terminology and Basic Idea	5
1.2.2	Examples of Testable Properties	5
1.2.3	Cube Testers on Trivium	6
1.3	Improving Cube Testers	6
1.3.1	Higher Order Derivatives of Boolean Functions	6
1.3.2	Conditional Differential Cryptanalysis	7
1.3.3	Application to Trivium, Grain, and KATAN / KTANTAN	7
1.3.4	Dynamic Cube Attack on Grain-128	8
1.4	Conclusion	8
2	Cryptanalysis of ARX Structures	9
2.1	Introduction	9
2.2	Rotational Cryptanalysis	10
2.3	The Differential Analysis of S-functions	11
2.4	The Additive Differential Probability of ARX	12
3	Biclique Cryptanalysis of Block Ciphers	13

Executive Summary

Symmetric cryptography addresses the problem of protecting the secret information using a shared secret key. The message is transformed in such a way that it cannot be recovered without this key. Algorithms which are used for symmetric encryption are known as symmetric ciphers: block ciphers and stream ciphers. The security of symmetric encryption algorithms can in general not be proved (with the exception of the one-time pad). Instead, the trust in a symmetric cipher is based on the fact that no weaknesses have been found after a long and thorough evaluation phase. The field which focuses on methods to defeat the secrecy of the information, i.e. which aims at breaking the ciphers is called **cryptanalysis**.

In this deliverable we provide a state of the art survey of recent developments in symmetric cryptanalysis. It is an update of the deliverable DSYM6 and includes the best to our knowledge new results and techniques developed for cryptanalysis of block ciphers, stream ciphers and hash functions.

Chapter 1

Cube Attacks and Other Higher Order Differential Techniques

Cube attacks have been presented by Dinur and Shamir [DS09] in 2008 as a new cryptanalytic tool applicable to a broad class of secret key primitives (stream ciphers and block ciphers). If the primitive has inherently low degree an attacker can recover the secret key by solving a system of linear equations. The authors of [DS09] introduced a new terminology which was reused in [ADMS09] to describe cube *testers*. In contrast to the cube *attacks*, cube testers typically yield distinguishing rather than key recovery attacks. Both types of attack apply in a scenario where the attacker can make chosen queries to the cipher (chosen IV or chosen plaintext).

Cube attacks as well as cube testers have a natural description in terms of higher order derivatives of Boolean functions as defined by Lai [Lai94] in 1994. In this perspective, cube testers have been improved using the idea of conditional differential cryptanalysis in [KMNP10, KMNP11]. A similar idea underlies the dynamic cube attack [DS11]. These improvements resulted in the first attack on the stream cipher Grain-128 and on the best known attacks on reduced variants of the stream ciphers Grain v1, Trivium, and the KATAN / KTANTAN family of lightweight block ciphers.

We first describe cube attacks and cube testers, before we describe the more recent improvements. We also provide a short translation of the cube terminology to the terminology of higher order differential cryptanalysis.

1.1 Cube Attacks

Cube attacks exploit implicit low-degree equations in cryptographic algorithms. They only require black box access to the target primitive, and were successfully applied to reduced variants of the stream cipher Trivium [DS09]. A very similar technique has been proposed earlier in [Vie07]. The attacker recovers a secret key through specific queries to the cipher, followed by solving a linear system of equations in the secret key variables. A one time preprocessing phase is required to determine which queries should be made during the online phase of the attack.

1.1.1 Terminology

Let f be a function mapping $\{0, 1\}^n$ to $\{0, 1\}$, $n > 0$. The *algebraic normal form* (ANF) of f is its representation as a polynomial over $\text{GF}(2)$ in variables x_1, \dots, x_n . It has the form

$$\sum_{i=0}^{2^n-1} a_i \cdot x_1^{i_1} x_2^{i_2} \cdots x_{n-1}^{i_{n-1}} x_n^{i_n},$$

where a_0, \dots, a_{2^n-1} are binary coefficients, and i_j denotes the j -th digit of the binary encoding of i . A key observation regarding cube attacks is that for any function $f : \{0, 1\}^n \rightarrow \{0, 1\}$, one has

$$a_{2^n-1} = \sum_{x \in \{0,1\}^n} f(x).$$

That is, the sum of all entries in the truth table equals the coefficient of the highest degree monomial in the ANF of f . For example, let $n = 4$ and f be defined as

$$f(x_1, x_2, x_3, x_4) = x_1 + x_1 x_2 x_3 + x_1 x_2 x_4 + x_3.$$

Summing f over all 16 distinct inputs yields zero, the coefficient of the monomial $x_1 x_2 x_3 x_4$. Instead, cube attacks sum over a *subset* of the inputs. For example, summing over the four possible values of $(x_1, x_2) \in \{0, 1\}^2$ gives

$$f(0, 0, x_3, x_4) + f(0, 1, x_3, x_4) + f(1, 0, x_3, x_4) + f(1, 1, x_3, x_4) = x_3 + x_4,$$

which is the polynomial that multiplies $x_1 x_2$ in f :

$$f(x_1, x_2, x_3, x_4) = x_1 x_2 \cdot (x_3 + x_4) + x_1 + x_3.$$

Generalizing, given a function $f : \{0, 1\}^n \rightarrow \{0, 1\}$ and an index set $I \subset \{1, \dots, n\}$, summing f over all possible values $x_i \in \{0, 1\}$ for $i \in I$ results in the polynomial p such that

$$f(x_1, \dots, x_n) = t_I \cdot p(\cdots) + q(x_1, \dots, x_n)$$

where t_I is the monomial containing all the x_i with $i \in I$, p has no variable in common with t_I , and no monomial in the polynomial q contains t_I .

Following the terminology of [DS09], p is called the *superpoly* of I in f , t_I is called a *maxterm* if p has degree 1, and f is called the *master polynomial*.

Preprocessing Phase

This phase must be done only once and does not require online access to the cipher. The goal is to find sufficiently many maxterms of the master polynomial. Each maxterm gives rise to a linear equation in the bits of the key. Hence, recovering the full key requires k maxterms, where k is the length of the key. Testing whether t_I actually is a maxterm and reconstructing its linear representation in key bits is achieved by probabilistic linearity tests [BLR90].

Online Phase

The attacker now evaluates the superpolys of all the k maxterms by summing f over the corresponding public variables. Each result corresponds to a linear combination of the key bits (determined in the preprocessing phase). Assuming that the degree of the master polynomial is d , each evaluation requires at most 2^{d-1} chosen queries. Once enough linear superpolys are found, the key can be recovered by solving the system of linear equations.

1.1.2 Cube Attacks on Trivium

The stream cipher Trivium was designed by De Cannière and Preneel [De 06] and was chosen for the final eSTREAM portfolio of hardware oriented ciphers¹. Trivium takes as input a 80-bit key and a 80-bit IV. Its initialization has 1152 rounds. Each round corresponds to clocking three non-linear feedback shift registers.

The cube attack could be successfully applied to a reduced variant with 767 initialization rounds [DS09].

1.2 Cube Testers

Unlike cube attacks, cube testers typically yield distinguishing rather than key recovery attacks. Cube testers have been introduced in [ADMS09] based on the terminology of [DS09], but similar ideas have been used earlier [EJT07, Fil02, FKM08, KM08, Saa06].

1.2.1 Terminology and Basic Idea

Informally, a *distinguisher* for a master polynomial f , representing a cipher with a fixed secret key, is a procedure that identifies a specific property of f that is unlikely to be observed for a randomly chosen function. In the case of cube testers, the procedure is allowed to make queries with chosen values for some public variables. The set of public variables is divided into two complementary subsets: cube variables (CV) and superpoly variables (SV). Cube testers evaluate the superpoly of the CV for different configurations of the SV in order to exhibit a distinguishing property. We illustrate these notions with our previous example,

$$f(x_1, x_2, x_3, x_4) = x_1 + x_1x_2x_3 + x_1x_2x_4 + x_3.$$

For the index set $I = \{1, 2\}$ the superpoly is $p(x_3, x_4) = x_3 + x_4$. Here, x_1 and x_2 are the CV, and x_3 and x_4 are the SV. The superpoly is linear in the SV which can be detected by the attacker. An even stronger property can be detected for the index set $I = \{3, 4\}$. Then, the superpoly evaluates to 0 for every configuration of the SV x_1 and x_2 . In comparison, for a function chosen uniformly at random from all functions $\{0, 1\}^4 \rightarrow \{0, 1\}$, the superpoly of x_3x_4 is zero with probability only 1/16. This yields a distinguisher for f that always identifies f (no false negatives) and with probability 1/16 incorrectly identifies a random function as f (false positives). The distinguisher makes 2^4 queries to f . Note that in this case, the distinguisher requires the entire truth table of f . At the cost of a higher rate of false positives, the number of queries can be reduced.

1.2.2 Examples of Testable Properties

Let us give some examples of properties which can be used to build cube testers. We let C be the size of CV, and S be the size of SV.

- **Imbalance.** A random function is expected to contain roughly the same number of zeroes and ones in its truth table. A strongly imbalanced truth table can be used as a distinguishing property. Typically, not the entire truth table is queried, but only a random sample. If the sample has size 2^N , $N < S$, the distinguisher requires 2^{C+N} queries to the cipher.

¹See <http://www.ecrypt.eu.org/stream/>

- **Constantness.** This is a special case of a maximally imbalanced truth table (all zero or all one).
- **Low Degree.** A superpoly of a random function has degree at least $S - 1$ with high probability. Probabilistic tests for the degree of Boolean functions are given in [AKK⁺03, Sam07], for example. The test in [AKK⁺03] for degree d queries the superpoly at about $d4^d$ points and always accepts if the ANF of the function has degree at most k , otherwise it rejects with some bounded error probability.
- **Linear Variables.** This is a special case of low degree. Probabilistic linearity tests are also required for the cube attacks, but in a slightly different context.
- **Neutral Variables.** Dually to the linearity, one can test whether a SV is neutral in the superpoly. Alternatively, this can be seen as a special case of imbalance on a restricted part of the truth table.

In practice, imbalance and neutral variables turned out to be most effective. Note that, in contrast to the cube attacks, these properties do not require the superpoly to have a particularly low degree.

1.2.3 Cube Testers on Trivium

For a reduced variant of Trivium with 790 rounds, a distinguisher based on neutral variables is given in [ADMS09]. The distinguisher requires 2^{31} chosen IV queries.

1.3 Improving Cube Testers

Recently, cube testers have been improved in two directions. Both directions have in common that some SV are assigned with specific values in order to amplify the non-random behavior of the superpoly. We first explain the basic idea from [KMNP10, KMNP11] that is based on a higher order differential view on cube testers. Then we summarize [DS11] which exploits a particular weakness of Grain-128.

1.3.1 Higher Order Derivatives of Boolean Functions

Let us briefly review the terminology of Lai [Lai94] which gives a more general view on cube attacks and cube testers in terms of higher order derivatives.

Let $f : \{0, 1\}^n \rightarrow \{0, 1\}$ be a Boolean function and $a \in \{0, 1\}^n$. The derivative of f with respect to a is defined as

$$\Delta_a f(x) = f(x) \oplus f(x \oplus a).$$

Note that computing the derivative of f with respect to a gives the output difference of f for the input difference a . Analogously, the following definition corresponds to the generalization of higher order differential cryptanalysis [Knu94]. Let $a_1, \dots, a_d \in \{0, 1\}^n$. The d -th derivative of f with respect to a_1, \dots, a_d is defined as

$$\Delta_{a_1, \dots, a_d}^{(d)} f(x) = \sum_{c \in L(a_1, \dots, a_d)} f(x \oplus c),$$

where $L(a_1, \dots, a_d)$ is the set of all 2^d linear combinations of a_1, \dots, a_d .

Relation to Cube Attacks.

The superpoly of $I \subset \{1, \dots, n\}$ in f is precisely the derivative of f with respect to $\{e_i, i \in I\}$, where $e_i \in \{0, 1\}^n$ has a one at position i and zeros otherwise. Hence, superpolys are a special case of higher order derivative with respect to points of Hamming weight one.

1.3.2 Conditional Differential Cryptanalysis

In [KMNP10, KMNP11] the differential view on cube testers is combined with the idea of conditional differential cryptanalysis from [BAB93]. The technique has been extended to higher order cryptanalysis and applied to constructions based on non-linear feedback shift registers (NLFSR).

Suppose a prototypical NLFSR-based cipher with an internal state of length ℓ which is initialized with a key k and an initial value x . Let s_0, s_1, \dots be the consecutive state bits generated by the cipher, such that $(s_i, \dots, s_{i+\ell})$ is the state after i rounds, and let h be the output function of the cipher such that $h(s_i, \dots, s_{i+\ell})$ is the output after i rounds. Every state bit is a function of (k, x) and the same is true for the output of h . For some fixed i , let $f = h(s_i, \dots, s_{i+\ell})$.

The idea of conditional differential cryptanalysis is to derive conditions on x that control the propagation of the difference up to some round r . This results in a system of equations

$$\begin{cases} \Delta_a s_1(k, x) = \gamma_1, \\ \Delta_a s_2(k, x) = \gamma_2, \\ \dots \\ \Delta_a s_r(k, x) = \gamma_r, \end{cases} \quad (1.1)$$

where the $\gamma_i \in \{0, 1\}$ describe the differential characteristic. The goal is to find a large sample of inputs that follow the same characteristic, such that their difference is imbalanced at the output. The conditions may also involve variables of the key. This allows for key recovery or classification of weak keys.

Analyzing the conditions is a crucial part of conditional differential cryptanalysis. If the system (1.1) is represented as an ideal in a suitable ring of Boolean polynomials, automatic tools such as Gröbner basis algorithms can be used for the analysis.

1.3.3 Application to Trivium, Grain, and KATAN / KTANTAN

Grain v1 is a stream cipher proposed by Hell, Johansson, and Meier [HJM07] and, as Trivium, has been selected for the final eSTREAM portfolio. It accepts an 80-bit key and a 64-bit initial value. Initialization takes 160 rounds. Grain-128 was designed by Hell, Johansson, Maximov, and Meier [HJMM06] as a bigger version of Grain v1. It accepts a 128-bit key, a 96-bit initial value, and initialization takes 256 rounds. Table 1.1 shows the results obtained with conditional differential cryptanalysis for reduced variants of Grain v1, Grain-128, and Trivium [KMNP10, KMNP11].

Conditional differential cryptanalysis also has been applied to the KATAN / KTANTAN family of lightweight block ciphers, designed by De Cannière, Dunkelman, and Knežević [DDK09]. The family consists of six ciphers in two flavors and three block sizes. All members of the family showed a comfortable security margin with respect to conditional differential cryptanalysis.

Table 1.1: Conditional differential cryptanalysis of NLFSR-based stream ciphers. Indicated is the *query* complexity (i.e., the number of chosen IVs). The time and memory complexities are negligible for all attacks.

Cipher	Rounds	Complexity	Weak keys	Type of attack
Grain v1	97	2^{31}	all	key recovery
	104	2^{39}	all	distinguisher
Grain-128	213	2^{25}	all	key recovery
	215	2^{25}	all	distinguisher
Trivium	798	2^{25}	all	distinguisher
	868	2^{25}	2^{31}	distinguisher
	961	2^{25}	2^{26}	distinguisher

1.3.4 Dynamic Cube Attack on Grain-128

The dynamic cube attack [DS11] is another way to improve the efficiency of cube testers and to turn them into key recovery attacks. As for conditional differential cryptanalysis specific values are assigned to the SV. But contrary to previous attacks, these values are dynamically adapted during the evaluation of the superpoly. This potentially increases the non-random behavior and eventually leaks information on the key. Finding the dependencies of the dynamic SV from the CV and eventually from the key is the intricate part of the attack.

In [DS11] a approach specific to Grain-128 has been used. The output function of Grain-128 has a single monomial of degree 3 (all other monomials have lower degree). This monomial contributes most to the high degree of the master polynomial f . Hence, one can try to nullify this monomial by nullifying one of its variables. In order to nullify this variable, one analyzes its algebraic representation in state bits, which gives rise to other variables that have to be nullified, and so on.

The dynamic cube attack can recover the full key of Grain-128 about 2^{38} times faster than exhaustive search for about 7.5% of the keys [DGP⁺11]. This attack is an improvement of the first attack on full Grain-128 which only worked for a much smaller class of weak keys [DS11].

1.4 Conclusion

In the last few years, higher order differential cryptanalysis underwent a revival in stream cipher cryptanalysis. Cube testers and their improvements can serve as a benchmark for evaluating the algebraic strength of constructions based on low degree components, and as a reference for choosing the number of rounds. The success of higher order differential techniques on stream ciphers influenced cryptanalysis of other primitives as well. Notable examples are non-random properties of the SHA-2 compression function [LM11], and zero-sum distinguishers [AM, BCC11].

Chapter 2

Cryptanalysis of ARX Structures

Increasingly, cryptographic primitives use operations such as addition modulo 2^n , rotation and exclusive ORs (ARX), as well as bitwise Boolean functions. In NIST's SHA-3 hash function competition, this applies to 6 out of the 14 second-round candidates and 2 out of 5 finalists. In this report, we give an overview of recent developments in the field of ARX-based cryptography. We provide a summary of results in rotational cryptanalysis, a technique that was very recently used to attack reduced-round Threefish, the block cipher that is at the core of the Skein hash function. We also describe the framework of S-functions, which is accompanied by a software toolkit. This framework can be used to calculate the probability that given input differences lead to given output differences for ARX-based constructions. It can also be used to count the number of possible output differences. The calculations can be efficiently performed using matrix multiplications. The S-function framework is further extended to analyze adp^{ARX} , the probability with which additive differences propagate through the following sequence of operations: modular addition, bit rotation and XOR (ARX).

2.1 Introduction

Differential cryptanalysis [BS91a] and linear cryptanalysis [MY92] are two of the most common methods in the cryptanalysis of block ciphers, hash functions and MACs.

For block ciphers, differential cryptanalysis [BS91a] analyzes how plaintext input differences lead to output differences in the ciphertext. The dual of differential cryptanalysis is linear cryptanalysis [MY92]. In linear cryptanalysis, a linear approximation is made between the bits of the plaintext, key and ciphertext. Both differential cryptanalysis and linear cryptanalysis can be used to construct distinguishers or even key-recovery attacks for the block cipher.

To analyze the non-linear components of a cryptographic primitive, differential cryptanalysis typically involves the construction of a difference distribution table. In this table, the number of occurrences for every combination of input and output differences is shown. For linear cryptanalysis, all linear approximations are enumerated in a linear approximation table. This is the standard approach for designs based on S-boxes.

Not all cryptographic primitives are based on S-boxes, however. It is also possible to achieve non-linearity by combining operations such as addition modulo 2^n , exclusive OR (XOR), Boolean functions, bit rotations and bit shifts. For Boolean functions, it is assumed that the same Boolean function is used for each bit position i of the n -bit input words. All

of these operations are very well suited for implementation in software, but constructing difference distribution tables and linear approximation tables for them becomes impractical for $n \geq 32$.

Cryptographic algorithms that involve addition, XOR and rotation, were referred to recently as AXR [Wei09], a name that was later changed to ARX. In this report, we describe the rotational cryptanalysis of ARX constructions, and introduce a toolkit for the efficient differential cryptanalysis of ARX.

Examples of ARX-based designs are the XTEA block cipher [NW97], the Salsa20 stream cipher family [Ber08], as well as the hash functions MD5, SHA-1. ARX and bitwise Boolean functions are also used in 6 out of the 14 second-round candidates of NIST's SHA-3 hash function competition [Nat07]: BLAKE [AHMP08], Blue Midnight Wish [GKK⁺09], Cube-Hash [Ber09], Shabal [BCCM⁺08], SIMD [LBF09] and Skein [FLS⁺09]. Out of the five finalists, NIST selected two purely ARX-based hash functions: BLAKE and Skein.

2.2 Rotational Cryptanalysis

In [KN10], the concept of rotational cryptanalysis is explained for ARX constructions. Let us consider the pair $(x, x \lll r)$, consisting of both x and x rotated to the left by r positions. We refer to $(x, x \lll r)$ as a rotational pair. Rotational cryptanalysis is based on the observation that if the inputs to the XOR, rotation or bitwise Boolean function operations are rotational pairs, the outputs are rotational pairs as well. With some probability, the same observation also holds for the addition operation. If rotational pairs can be obtained more easily for a given cryptographic primitive than for a random permutation, this observation can be used to build a distinguisher or even a key recovery attack.

The concept of rotational cryptanalysis is not new, but has recently gained a lot of interest because of the increasing number of ARX-based designs. In pioneering work by Biham [Bih94], rotational pairs of keys were considered for the block ciphers LOKI89, LOKI91 and Lucifer. This approach was extended in [KSW97] to related-key attacks on several block ciphers. This is not pure rotational cryptanalysis, however, because the attacker searches for plaintexts of the form $(p, F(p))$, where F is the round transformation.

For Salsa20 [Ber08], Bernstein explicitly prevented attacks based on rotational pairs, by using constants without rotational symmetry at the input of the permutation. However, he did not give complexity estimates of such an attack.

In [MT09], a related-key attack was constructed using rotational pairs for a variant of the block cipher ESSENCE. This attack uses the observation that rotational pairs pass through bitwise Boolean functions with probability one. For the linear function L , the following observation was made:

Let us use the polynomial representation of $\mathbb{F}_{2^{32}}$. A multiplication of any $a \in \mathbb{F}_{2^{32}}$ by x then corresponds to a binary left shift by one position, and an XOR with the feedback polynomial if and only if the most significant bit of a is one. The following relation thus holds:

$$\text{MSB}(a) = 0 \Leftrightarrow a \cdot x = a \ll 1 . \quad (2.1)$$

The linear function L of ESSENCE is implemented as an LFSR. Using the polynomial representation of $\mathbb{F}_{2^{32}}$, we can write $L(v) = v \cdot x^{32}$.

We have that $L(v \cdot x) = (v \cdot x) \cdot x^{32} = (v \cdot x^{32}) \cdot x = L(v) \cdot x$. For a random $v \in \mathbb{F}_{2^{32}}$, with probability 2^{-2} we have that $\text{MSB}(v) = 0$ and $\text{MSB}(L(v)) = 0$. In that case:

$$L(v \ll 1) = L(v) \ll 1 . \quad (2.2)$$

If $\text{MSB}(a) = 0$, then $a \ll 1$ and $a \lll 1$ are equivalent. Because of the particular choice of the Boolean function in ESSENCE, this attack does not work on ESSENCE itself, but on a variant of the ESSENCE block cipher.

The idea of rotational inputs was used as well to find fixed points and key collisions for the permutation \mathcal{P} used in the hash function Shabal [KMT09]. Rotational pairs were traced through bitwise logical operations and rotations, however rotations were chosen in such a way that there was no loss in probability for the additions operations in $\mathcal{U}(x) = x + x \lll 1$ and $\mathcal{V}(x) = x + x \lll 2$.

In [KN10], Khovratovich et al. formally explain the concept of rotational cryptanalysis, and applied it to reduced rounds of the Threefish block cipher, the core of the Skein hash function. As in [MT09], a related-key chosen-plaintext attack is constructed, where both keys and plaintexts are rotated. Cryptanalysis results are presented on 39, 42 and 43 full rounds of Threefish-256, -512 and -1024 respectively, where the attack complexity is estimated to be slightly less than generic.

In [KNR10], Khovratovich et al. combine the rotational cryptanalysis attack with the rebound attack [MRST09]. Before this result, the rebound attack approach was only applied to AES-like constructions. A cryptanalytic result is obtained on an estimated 53/57 out of the 72 rounds of the Skein-256/512 compression function and the Threefish cipher.

A mention of the completeness of ARX is given in [Ber08]. A formal proof can be found in [KN10]. Because ARX is shown to be functionally complete, it is possible to use addition, rotation and XOR to implement any circuit (including all block ciphers, hash functions and MACs), but not necessarily in the most efficient way. Although this result indicates the soundness of ARX constructions, a new toolkit is required for the cryptanalysis and design of these primitives. Such a toolkit is presented in the next section.

2.3 The Differential Analysis of S-functions

In [MVCP10], Mouha et al. presented the first fully general framework to analyze these constructions efficiently. It was inspired by the cryptanalysis techniques for SHA-1 by De Cannière and Rechberger [DR06] (clarified in [MDIP09]), and by methods introduced by Lipmaa, Wallén and Dumas [LWD04]. The framework was used to calculate the probability that given input differences lead to given output differences, as well as to count the number of output differences with non-zero probability. The methods are based on graph theory, and the calculations can be efficiently performed using matrix multiplications.

The framework proposed by Mouha et al. is accompanied by a software toolkit, which has been made publicly available on-line¹. It can be used to calculate XOR-differential probability of addition (xdp^+), the additive differential probability of XOR (adp^\oplus), as well as $\text{xdp}^+(\alpha, \beta, \dots \rightarrow \gamma)$, which is the calculation of xdp^+ for more than two inputs, and the differential probability $\text{xdp}^{\times C}$ of multiplication by a constant C where differences are expressed by xor.

¹The software toolkit is part of the ECRYPT II Tools for Cryptography website, and is available at: <http://www.ecrypt.eu.org/tools/s-function-toolkit>

The tool can also efficiently count the number of output differences for each of these operations. For example, this problem occurs in the cryptanalysis of Threefish-512 [AccM⁺09], where an exponential-in- n time algorithm is proposed. Using the toolkit, however, this can be solved in linear time in n .

Additionally, the toolkit provides a general algorithm to efficiently list the output differences with the highest probability, assuming input differences and the operation are given.

2.4 The Additive Differential Probability of ARX

In [VMCP11], Velichkov et al. calculate the probability with which additive differences propagate through the following sequence of operations: modular addition, bit rotation and XOR (ARX). This probability is denoted by adp^{ARX} .

The S-function framework of [MVCP10] is extended to compute the differential probability adp^{ARX} . A method is described to compute adp^{ARX} based on the matrix multiplication technique proposed in [LWD04], and generalized in [MVCP10]. The time complexity of the proposed algorithm is linear in the word size. A formal proof for the correctness of the algorithm is provided.

It is observed that the adp^{ARX} can differ significantly from the probability obtained by multiplying the differential probabilities of addition, rotation and XOR. This confirms the need for an efficient calculation of the differential probability for the ARX operation. The result of [VMCP11] is the first in literature to calculate adp^{ARX} efficiently. Accurate and efficient calculations of differential probabilities are required for the efficient search for characteristics used in differential cryptanalysis.

Chapter 3

Biclique Cryptanalysis of Block Ciphers

This chapter briefly summarizes recent progress in key recovery for block ciphers due to the new technique of biclique cryptanalysis to enhance meet-in-the-middle attacks.

Following the lines of the work [BKR11a, BKR11b], we motivate and summarize the recent progress in meet-in-the-middle attacks on block ciphers, including the first key recovery against all three variants of AES in the standard single-key model faster than brute force.

Attacks on hash functions have received a lot of attention lately. First of all, this is due to the relatively recent success in collision attacks of MD5 [WY05], SHA-0 [BCJ⁺05, CJ98] and SHA-1 [XWY05] (including the practical cryptanalysis MD5 [SLdW07, SSA⁺09] in the context) and to preimage attacks on MD5 [SA09].

Differential cryptanalysis [BS91b] was discovered at the example of block ciphers. However, it was successfully applied to hash function analysis as well and even found its own non-negligible development there. Moreover, the cryptanalysis of hash functions has got to the point that there has been evidence that techniques of hash function cryptanalysis can result in new insight into block ciphers, e.g. the related-key attacks on AES with local collisions [SA10, BDK⁺10, BKN09, BN10]. So far it has not been clear if local collisions can result in a secret single-key attack.

Though there has been a great deal of meet-in-the-middle attacks on block ciphers recently [BR10, CBF11, CE85, DK10, ODP07, Iso11, WRG⁺11], they receive less focus from cryptanalysts than the standard attack vectors. A basic meet-in-the-middle attack requires only the information-theoretical minimum of plaintext-ciphertext pairs. The limited use of these attacks can be attributed to the requirement for large parts of the cipher to be independent of particular key bits. As this requirement is not met in AES and most AES candidates, the number of rounds broken with this technique is rather small [DK10, CBF11], which seems to prevent it from producing results on yet unbroken number of rounds in AES. We also mention that the collision attacks [DS08, HDB09] use some elements of the meet-in-the-middle framework.

A new concept called *bicliques* was first introduced for hash cryptanalysis by Savelieva et al. [DKS11]. It originates from the splice-and-cut framework [AS08, AS09, JGW] in hash function cryptanalysis, and more specifically its element called initial structure. The biclique approach led to the best preimage attacks on the SHA family of hash functions so far, including the attack on 50 rounds of SHA-512, and the first attack on a round-reduced Skein hash

function [DKS11]. The concept of bicliques for block ciphers and biclique cryptanalysis for block ciphers was introduced in [BKR11a,BKR11b]. Here bicliques allow to get significant cryptanalytic results, including the first key recovery for all versions of the full AES faster than brute force.

Tables 3.1 and 3.2 summarize the results on key recovery for AES and preimage finding for AES in hash modes.

Table 3.1: Biclique key recovery for AES [BKR11a,BKR11b]

rounds	data	computations/succ.rate	memory	biclique length in rounds
AES-128 secret key recovery				
8	$2^{126.33}$	$2^{124.97}$	2^{102}	5
8	2^{127}	$2^{125.64}$	2^{32}	5
8	2^{88}	$2^{125.34}$	2^8	3
10	2^{88}	$2^{126.18}$	2^8	3
AES-192 secret key recovery				
9	2^{80}	$2^{188.8}$	2^8	4
12	2^{80}	$2^{189.74}$	2^8	4
AES-256 secret key recovery				
9	2^{120}	$2^{253.1}$	2^8	6
9	2^{120}	$2^{251.92}$	2^8	4
14	2^{40}	$2^{254.42}$	2^8	4

Table 3.2: Biclique preimage search of AES in hash modes (compression function) [BKR11a, BKR11b]

rounds	computations	succ.rate	memory	biclique length in rounds
AES-128 compression function preimage, Miyaguchi-Preneel				
10	$2^{125.83}$	0.632	2^8	3
AES-192 compression function preimage, Davies-Meyer				
12	$2^{125.71}$	0.632	2^8	4
AES-256 compression function preimage, Davies-Meyer				
14	$2^{126.35}$	0.632	2^8	4

The concept of a biclique. A biclique (a complete bipartite graph) connects 2^d pairs of intermediate states with 2^{2d} keys. This is the main source of computational advantage in the key recovery — by constructing a biclique on 2^d vertices only, one covers quadratically as many keys 2^{2d} . d is called the dimension of the biclique.

A biclique is characterized by its length (number of rounds covered) and dimension d . The dimension is related to the cardinality of the biclique elements and is one of the factors that determines the advantage over brute force.

Bicliques from independent related-key differentials. Often the easiest way to construct a biclique in a cipher is to consider two related-key differentials holding with probability one — one with forward key modification and one with backward key modification. If those differentials are truncated, this can result in a higher dimensional biclique. The biclique key recovery for the full AES uses bicliques of dimension $d = 8$ constructed from truncated related-key probability-one differentials in [BKR11a,BKR11b].

Bicliques from interleaving related-key differentials. This is frequently a more involving approach. It is also based on related-key differentials. However, they can interleave (that is, intersect in active nonlinear components such as S-boxes). The propagation in those differentials can also be of probabilistic nature. This removes the constraint on the biclique length natural for bicliques from independent related-key differentials. This imposes a limitation of the highest dimension of a biclique though. It is typical to have $d = 1$ in key recoveries on round-reduced AES in [BKR11a,BKR11b]. The construction of such bicliques can follow the rebound strategy borrowed from the domain of hash functions.

Summarizing, the novel biclique meet-in-the-middle cryptanalysis on block ciphers introduced in [BKR11a,BKR11b] is a promising cryptanalytic technique that is essential to the security evaluation of modern block ciphers. It is advisable to include an assessment of the biclique cryptanalysis applicability into any new block cipher design.

Bibliography

- [AccM⁺09] Jean-Philippe Aumasson, Çağdas Çalik, Willi Meier, Onur Özen, Raphael C.-W. Phan, and Kerem Varıcı. Improved Cryptanalysis of Skein. In Mitsuru Matsui, editor, *ASIACRYPT*, volume 5912 of *LNCS*, pages 542–559. Springer, 2009.
- [ADMS09] Jean-Philippe Aumasson, Itai Dinur, Willi Meier, and Adi Shamir. Cube Testers and Key Recovery Attacks on Reduced-Round MD6 and Trivium. In Dunkelman [Dun09], pages 1–22.
- [AHMP08] Jean-Philippe Aumasson, Luca Henzen, Willi Meier, and Raphael C.-W. Phan. SHA-3 proposal BLAKE. Submission to the NIST SHA-3 Competition (Round 2), 2008. <http://131002.net/blake/blake.pdf>.
- [AKK⁺03] Noga Alon, Tali Kaufman, Michael Krivelevich, Simon Litsyn, and Dana Ron. Testing Low-Degree Polynomials over $GF(2)$. In Sanjeev Arora, Klaus Jansen, José D. P. Rolim, and Amit Sahai, editors, *RANDOM-APPROX*, volume 2764 of *Lecture Notes in Computer Science*, pages 188–199. Springer, 2003.
- [AM] Jean-Philippe Aumasson and Willi Meier. Zero-sum distinguishers for reduced Keccak- f and for the core functions of Luffa and Hamsi. Online at <http://www.131002.net/papers.html>.
- [AS08] Kazumaro Aoki and Yu Sasaki. Preimage attacks on one-block MD4, 63-step MD5 and more. In *SAC 2008*, volume 5381 of *LNCS*, pages 103–119. Springer-Verlag, 2008.
- [AS09] Kazumaro Aoki and Yu Sasaki. Meet-in-the-middle preimage attacks against reduced SHA-0 and SHA-1. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 70–89. Springer-Verlag, 2009.
- [BAB93] Ishai Ben-Aroya and Eli Biham. Differential Cryptanalysis of Lucifer. In Douglas R. Stinson, editor, *CRYPTO*, volume 773 of *Lecture Notes in Computer Science*, pages 187–199. Springer, 1993.
- [BCC11] Christina Boura, Anne Canteaut, and Christophe De Cannière. Higher-Order Differential Properties of Keccak and *luffa*. In Joux [Jou11], pages 252–269.
- [BCCM⁺08] 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 the NIST SHA-3 Competition (Round 2), 2008. <http://ehash.iaik.tugraz.at/uploads/6/6c/Shabal.pdf>.
- [BCJ⁺05] Eli Biham, Rafi Chen, Antoine Joux, Patrick Carribault, Christophe Lemuet, and William Jalby. Collisions of SHA-0 and reduced SHA-1. In Ronald Cramer, editor, *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 36–57. Springer-Verlag, 2005.
- [BDK⁺10] Alex Biryukov, Orr Dunkelman, Nathan Keller, Dmitry Khovratovich, and Adi Shamir. Related-key cryptanalysis of the full aes-192 and aes-256. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 299–319. Springer-Verlag, 2010.
- [Ber08] Daniel J. Bernstein. The Salsa20 Family of Stream Ciphers. In Matthew J. B. Robshaw and Olivier Billet, editors, *The eSTREAM Finalists*, volume 4986 of *LNCS*, pages 84–97. Springer, 2008.
- [Ber09] Daniel J. Bernstein. CubeHash specification (2.B.1). Submission to the NIST SHA-3 Competition (Round 2), 2009. <http://cubehash.cr.yp.to/submission2/spec.pdf>.
- [Bih94] Eli Biham. New Types of Cryptanalytic Attacks Using Related Keys. *J. Cryptology*, 7(4):229–246, 1994.
- [BKN09] Alex Biryukov, Dmitry Khovratovich, and Ivica Nikolić. Distinguisher and related-key attack on the full AES-256. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 231–249. Springer-Verlag, 2009.
- [BKR11a] Andrey Bogdanov, Dmitry Khovratovich, and Christian Rechberger. Biclique cryptanalysis of the full aes. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 344–371. Springer-Verlag, 2011.
- [BKR11b] Andrey Bogdanov, Dmitry Khovratovich, and Christian Rechberger. Biclique cryptanalysis of the full aes. Cryptology ePrint Archive: Report 2011/449, 2011.
- [BLR90] Manuel Blum, Michael Luby, and Ronitt Rubinfeld. Self-Testing/Correcting with Applications to Numerical Problems. In *STOC*, pages 73–83. ACM, 1990.
- [BN10] Alex Biryukov and Ivica Nikolić. Automatic search for related-key differential characteristics in byte-oriented block ciphers: Application to aes, camellia, khazad and others. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 5677 of *LNCS*, pages 322–344. Springer-Verlag, 2010.
- [BR10] Andrey Bogdanov and Christian Rechberger. A 3-subset meet-in-the-middle attack: Cryptanalysis of the lightweight block cipher ktantan. In A. Biryukov, G. Gong, and D. Stinson, editors, *SAC 2010*, volume 6544 of *LNCS*, pages 229–240. Springer-Verlag, 2010.
- [BS91a] Eli Biham and Adi Shamir. Differential Cryptanalysis of DES-like Cryptosystems. *J. Cryptology*, 4(1):3–72, 1991.

- [BS91b] Eli Biham and Adi Shamir. Differential cryptanalysis of des-like cryptosystems. *J. Cryptology*, 4(1):3–72, 1991.
- [CBF11] Patrick Derbez Charles Bouillaguet and Pierre-Alain Fouque. Automatic search of attacks on round-reduced aes and applications. In Henri Gilbert, editor, *CRYPTO 2011*, volume 2442 of *LNCS*, pages 169–187. Springer-Verlag, 2011.
- [CE85] David Chaum and Jan-Hendrik Evertse. Cryptanalysis of des with a reduced number of rounds: Sequences of linear factors in block ciphers. In H. C. Williams, editor, *CRYPTO 1985*, volume 218 of *LNCS*, pages 192–211. Springer-Verlag, 1985.
- [CJ98] Florent Chabaud and Antoine Joux. Differential collisions in SHA-0. In H. Krawczyk, editor, *CRYPTO 1998*, volume 1462 of *LNCS*, pages 56–71. Springer-Verlag, 1998.
- [DDK09] Christophe De Cannière, Orr Dunkelman, and Miroslav Knezevic. KATAN and KTANTAN - A Family of Small and Efficient Hardware-Oriented Block Ciphers. In Christophe Clavier and Kris Gaj, editors, *CHES*, volume 5747 of *LNCS*, pages 272–288. Springer, 2009.
- [De 06] Christophe De Cannière. Trivium: A Stream Cipher Construction Inspired by Block Cipher Design Principles. In Sokratis K. Katsikas, Javier Lopez, Michael Backes, Stefanos Gritzalis, and Bart Preneel, editors, *ISC*, volume 4176 of *LNCS*, pages 171–186. Springer, 2006.
- [DGP⁺11] Itai Dinur, Tim Güneysu, Christof Paar, Adi Shamir, and Ralf Zimmermann. An Experimentally Verified Attack on Full Grain-128 Using Dedicated Reconfigurable Hardware. *IACR Cryptology ePrint Archive*, 2011:282, 2011.
- [DK10] Orr Dunkelman and Nathan Keller. The effects of the omission of last round’s mixcolumns on aes. *Inf. Process. Lett.*, 110(8-9):304–308, 2010.
- [DKS11] Christian Rechberger Dmitry Khovratovich and Alexandra Savelieva. Bicliques for preimages: attacks on Skein-512 and the SHA-2 family. *Cryptology ePrint Archive: Report 2011/286*, 2011.
- [DR06] Christophe De Cannière and Christian Rechberger. Finding SHA-1 Characteristics: General Results and Applications. In Xuejia Lai and Kefei Chen, editors, *ASIACRYPT*, volume 4284 of *LNCS*, pages 1–20. Springer, 2006.
- [DS08] Hüseyin Demirci and Ali Aydin Selçuk. A meet-in-the-middle attack on 8-round AES. In Kaisa Nyberg, editor, *FSE 2008*, volume 5086 of *LNCS*, pages 116–126. Springer-Verlag, 2008.
- [DS09] Itai Dinur and Adi Shamir. Cube Attacks on Tweakable Black Box Polynomials. In Antoine Joux, editor, *EUROCRYPT*, volume 5479 of *Lecture Notes in Computer Science*, pages 278–299. Springer, 2009.
- [DS11] Itai Dinur and Adi Shamir. Breaking Grain-128 with Dynamic Cube Attacks. In Antoine Joux, editor, *FSE*, volume 6733 of *Lecture Notes in Computer Science*, pages 167–187. Springer, 2011.

- [Dun09] Orr Dunkelman, editor. *Fast Software Encryption, 16th International Workshop, FSE 2009, Leuven, Belgium, February 22-25, 2009, Revised Selected Papers*, volume 5665 of *Lecture Notes in Computer Science*. Springer, 2009.
- [EJT07] Håkan Englund, Thomas Johansson, and Meltem Sönmez Turan. A Framework for Chosen IV Statistical Analysis of Stream Ciphers. In K. Srinathan, C. Pandu Rangan, and Moti Yung, editors, *INDOCRYPT*, volume 4859 of *LNCS*, pages 268–281. Springer, 2007.
- [Fil02] Eric Filiol. A New Statistical Testing for Symmetric Ciphers and Hash Functions. In Robert H. Deng, Sihang Qing, Feng Bao, and Jianying Zhou, editors, *ICICS*, volume 2513 of *LNCS*, pages 342–353. Springer, 2002.
- [FKM08] Simon Fischer, Shahram Khazaei, and Willi Meier. Chosen IV Statistical Analysis for Key Recovery Attacks on Stream Ciphers. In Serge Vaudenay, editor, *AFRICACRYPT*, volume 5023 of *LNCS*, pages 236–245. Springer, 2008.
- [FLS⁺09] Niels Ferguson, Stefan Lucks, Bruce Schneier, Doug Whiting, Mihir Bellare, Tadayoshi Kohno, Jon Callas, and Jesse Walker. The Skein Hash Function Family. Submission to the NIST SHA-3 Competition (Round 2), 2009. <http://www.skein-hash.info/sites/default/files/skein1.2.pdf>.
- [GKK⁺09] Danilo Gligoroski, Vlastimil Klima, Svein Johan Knapskog, Mohamed El-Hadedy, Jørn Amundsen, and Stig Frode Mjølsnes. Cryptographic Hash Function BLUE MIDNIGHT WISH. Submission to the NIST SHA-3 Competition (Round 2), 2009. http://people.item.ntnu.no/~danilog/Hash/BMW-SecondRound/Supporting_Documentation/BlueMidnightWishDocumentation.pdf.
- [HDB09] Mustafa Çoban Hüseyin Demirci, Ihsan Taskin and Adnan Baysal. Improved meet-in-the-middle attacks on aes. In Bimal K. Roy and Nicolas Sendrier, editors, *INDOCRYPT 2009*, volume 5922 of *LNCS*, pages 144–156. Springer-Verlag, 2009.
- [HJM07] Martin Hell, Thomas Johansson, and Willi Meier. Grain: A Stream Cipher for Constrained Environments. *IJWMC*, 2(1):86–93, 2007.
- [HJMM06] Martin Hell, Thomas Johansson, Alexander Maximov, and Willi Meier. A Stream Cipher Proposal: Grain-128. In *ISIT*, pages 1614–1618, 2006.
- [Iso11] Takanori Isobe. A single-key attack on the full gost block cipher. In Antoine Joux, editor, *FSE 2011*, volume 6733 of *LNCS*, pages 290–305. Springer-Verlag, 2011.
- [JGW] Christian Rechberger Jian Guo, San Ling and Huaxiong Wang. Advanced meet-in-the-middle preimage attacks: First results on full tiger, and improved results on md4 and sha-2.
- [Jou11] Antoine Joux, editor. *Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers*, volume 6733 of *Lecture Notes in Computer Science*. Springer, 2011.

- [KM08] Shahram Khazaei and Willi Meier. New Directions in Cryptanalysis of Self-Synchronizing Stream Ciphers. In Dipanwita Roy Chowdhury, Vincent Rijmen, and Abhijit Das, editors, *INDOCRYPT*, volume 5365 of *LNCS*, pages 15–26. Springer, 2008.
- [KMNP10] Simon Knellwolf, Willi Meier, and María Naya-Plasencia. Conditional Differential Cryptanalysis of NLFSR-Based Cryptosystems. In Masayuki Abe, editor, *ASIACRYPT*, volume 6477 of *Lecture Notes in Computer Science*, pages 130–145. Springer, 2010.
- [KMNP11] Simon Knellwolf, Willi Meier, and María Naya-Plasencia. Conditional Differential Cryptanalysis of Trivium and KATAN. In Ali Miri and Serge Vaudenay, editors, *Selected Areas in Cryptography*, 2011.
- [KMT09] Lars R. Knudsen, Krystian Matusiewicz, and Søren S. Thomsen. Observations on the Shabal keyed permutation. Official Comment to the NIST SHA-3 Competition, 2009. <http://www.mat.dtu.dk/people/S.Thomsen/shabal/shabal.pdf>.
- [KN10] Dmitry Khovratovich and Ivica Nikolic. Rotational Cryptanalysis of ARX. *Fast Software Encryption, 11th International Workshop, FSE 2010, Seoul, Korea*, 2010.
- [KNR10] Dmitry Khovratovich, Ivica Nikolic, and Christian Rechberger. Rotational Rebound Attacks on Reduced Skein. In Masayuki Abe, editor, *ASIACRYPT*, volume 6477 of *Lecture Notes in Computer Science*, pages 1–19. Springer, 2010.
- [Knu94] Lars R. Knudsen. Truncated and Higher Order Differentials. In Bart Preneel, editor, *FSE*, volume 1008 of *LNCS*, pages 196–211. Springer, 1994.
- [KSW97] John Kelsey, Bruce Schneier, and David Wagner. Related-key cryptanalysis of 3-WAY, Biham-DES, CAST, DES-X, NewDES, RC2, and TEA. In Yongfei Han, Tatsuaki Okamoto, and Sihan Qing, editors, *ICICS*, volume 1334 of *LNCS*, pages 233–246. Springer, 1997.
- [Lai94] Xuejia Lai. Higher order derivatives and differential cryptanalysis. In Richard E. Blahut, Daniel J. Costello, Ueli Maurer, and Thomas Mittelholzer, editors, *Communicationis and Cryptography: Two Sides of one Tapestry*, pages 227–233. Kluwer Academic Publishers, 1994.
- [LBF09] Gaëtan Leurent, Charles Bouillaguet, and Pierre-Alain Fouque. SIMD Is a Message Digest. Submission to the NIST SHA-3 Competition (Round 2), 2009. <http://www.di.ens.fr/~leurent/files/SIMD.pdf>.
- [LM11] Mario Lamberger and Florian Mendel. Higher-Order Differential Attack on Reduced SHA-256. *IACR Cryptology ePrint Archive*, 2011:37, 2011.
- [LWD04] Helger Lipmaa, Johan Wallén, and Philippe Dumas. On the Additive Differential Probability of Exclusive-Or. In Bimal K. Roy and Willi Meier, editors, *FSE*, volume 3017 of *LNCS*, pages 317–331. Springer, 2004.

- [MDIP09] Nicky Mouha, Christophe De Cannière, Sebastiaan Indesteege, and Bart Preneel. Finding Collisions for a 45-Step Simplified HAS-V. In Heung Youl Youm and Moti Yung, editors, *WISA*, volume 5932 of *LNCS*, pages 206–225. Springer, 2009.
- [MRST09] Florian Mendel, Christian Rechberger, Martin Schl affer, and S oren S. Thomsen. The Rebound Attack: Cryptanalysis of Reduced Whirlpool and Gr ostl. In Dunkelman [Dun09], pages 260–276.
- [MT09] Nicky Mouha and Meltem S onmez Turan. A related-key attack on the ESSENCE block cipher. *ECRYPT II First Hash Function Retreat, 2010, Graz, Austria (Rump session talk)*, 2009.
- [MVCP10] Nicky Mouha, Vesselin Velichkov, Christophe De Cannière, and Bart Preneel. The Differential Analysis of S-Functions. In Alex Biryukov, Guang Gong, and Douglas R. Stinson, editors, *Selected Areas in Cryptography*, volume 6544 of *Lecture Notes in Computer Science*, pages 36–56. Springer, 2010.
- [MY92] Mitsuru Matsui and Atsuhiko Yamagishi. A New Method for Known Plaintext Attack of FEAL Cipher. In *EUROCRYPT*, pages 81–91, 1992.
- [Nat07] National Institute of Standards and Technology. Announcing Request for Candidate Algorithm Nominations for a New Cryptographic Hash Algorithm (SHA-3) Family. *Federal Register*, 27(212):62212–62220, November 2007. Available: http://csrc.nist.gov/groups/ST/hash/documents/FR_Notice_Nov07.pdf (2008/10/17).
- [NW97] Roger M. Needham and David J. Wheeler. Tea extensions. Computer Laboratory, Cambridge University, England, 1997. <http://www.movable-type.co.uk/scripts/xtea.pdf>.
- [ODP07] Gautham Sekar Orr Dunkelman and Bart Preneel. Improved meet-in-the-middle attacks on reduced-round DES. In K. Srinathan, C. Pandu Rangan, and Moti Yung, editors, *INDOCRYPT 2007*, volume 4859 of *LNCS*, pages 86–100. Springer-Verlag, 2007.
- [SA09] Yu Sasaki and Kazumaro Aoki. Finding preimages in full md5 faster than exhaustive search. In Antoine Joux, editor, *EUROCRYPT 2009*, volume 5479 of *LNCS*, pages 134–152. Springer-Verlag, 2009.
- [SA10] Yu Sasaki and Kazumaro Aoki. Alex biryukov and orr dunkelman and nathan keller and dmitry khovratovich and adi shamir. In Antoine Joux, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 299–319. Springer-Verlag, 2010.
- [Saa06] Markku-Juhani Olavi Saarinen. Chosen-IV Statistical Attacks on eStream Ciphers. In Manu Malek, Eduardo Fern andez-Medina, and Javier Hernando, editors, *SECRYPT*, pages 260–266. INSTICC Press, 2006.
- [Sam07] Alex Samorodnitsky. Low-degree tests at large distances. In David S. Johnson and Uriel Feige, editors, *STOC*, pages 506–515. ACM, 2007.

- [SLdW07] Marc Stevens, Arjen K. Lenstra, and Benne de Weger. Chosen-prefix collisions for MD5 and colliding X.509 certificates for different identities. In Moni Naor, editor, *EUROCRYPT 2007*, volume 4515 of *LNCS*, pages 1–22. Springer-Verlag, 2007.
- [SSA⁺09] Marc Stevens, Alexander Sotirov, Jacob Appelbaum, Arjen K. Lenstra, David Molnar, Dag Arne Osvik, and Benne de Weger. Short chosen-prefix collisions for MD5 and the creation of a rogue ca certificate. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 55–69. Springer-Verlag, 2009.
- [Vie07] Michael Vielhaber. Breaking ONE.FIVIUM by AIDA an Algebraic IV Differential Attack. Cryptology ePrint Archive, Report 2007/413, 2007.
- [VMCP11] Vesselin Velichkov, Nicky Mouha, Christophe De Cannière, and Bart Preneel. The Additive Differential Probability of ARX. In Joux [Jou11], pages 342–358.
- [Wei09] Ralf-Philipp Weinmann. AXR - Crypto Made from Modular Additions, XORs and Word Rotations. *Dagstuhl Seminar 09031*, January 2009. Available: <http://www.dagstuhl.de/Materials/AbstractListing/index.en.phtml?09031>.
- [WRG⁺11] Lei Wei, Christian Rechberger, Jian Guo, Hongjun Wu, Huaxiong Wang, and San Ling. Improved meet-in-the-middle cryptanalysis of KTANTAN. Cryptology ePrint Archive: Report 2011/201, 2011.
- [WY05] Xiaoyun Wang and Hongbo Yu. How to break MD5 and other hash functions. In Ronald Cramer, editor, *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 19–35. Springer-Verlag, 2005.
- [XWY05] Yiqun Lisa Yin Xiaoyun Wang and Hongbo Yu. Finding collisions in the full SHA-1. In Victor Shoup, editor, *CRYPTO 2005*, volume 3621 of *LNCS*, pages 17–36. Springer-Verlag, 2005.