Cryptanalytic Attacks on Symmetric Ciphers or How to Design a Secure Cryptosystem

Konstantinos Limniotis

Hellenic Data Protection Authority,
Kifissias 1-3,
11523 Athens, Greece
Email: klimniotis@dpa.gr

Dept. of Informatics & Telecommunications,
National and Kapodistrian University of Athens,
15784 Athens, Greece
Email: klimn@di.uoa.gr

5th Conference on e-Democracy -
Security, Privacy and Trust in a Digital World

December 5-6, 2013, Athens, Greece

Talk Outline

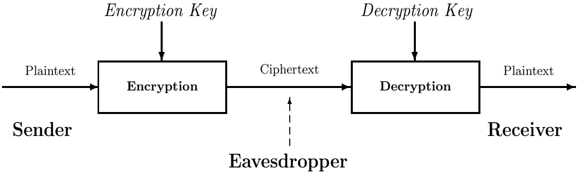
1. Introduction - Definitions
2. Mathematical analysis
    - Properties of cryptographic functions
3. Security of modern ciphers
    - Security of cryptographic protocols
4. Conclusions

Cryptography in practice

- e-democracy: Need to build citizens' trust
    - Without trust, citizens will not visit portals, will not exchange data,...
- Security challenges: confidentiality (privacy), integrity, authentication, transparency
- Cryptographic primitives have a crucial role
    - Confidentiality of the transmitted data is mainly ensured by symmetric cryptography
        - Characteristic example: SSL/TLS protocol (underlying in the `https` connections)
    - Symmetric cryptography is also used in several other cases (wireless networks, mobile networks, RFID applications etc.)
- Aim of this talk: Overview of recent developments and current research trends

Symmetric ciphers

A typical cryptosystem



Symmetric cryptography

- Encryption Key = Decryption Key
- The key is only shared between the two parties
    - The security rests with the secrecy of the key (Kerchoffs principle)

Two types of symmetric ciphers

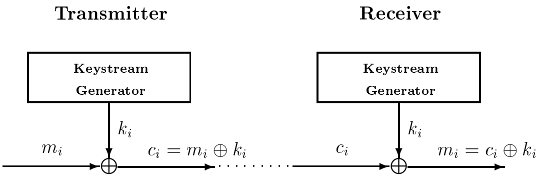
- Stream ciphers
- Block ciphers

Attacks models

- Ciphertert-only attack
  - The attacker knows only the ciphertext
- Known-plaintext attack
  - The attacker also knows part of the plaintext
- Chosen-plaintext attack
  - It is assumed that the attacker is able to choose plaintexts to encrypt and, then, to observe the corresponding ciphertexts
- Chosen-ciphertext attack
  - It is assumed that the attacker is able to choose ciphertexts to decrypt and, then, to observe the corresponding plaintexts
- The last two types of attacks are more theoretical than practical.
  - However, a cipher is being considered as (computationally) secure only if all types of attacks require prohibitive computational cost

Stream ciphers

Typical Case: Binary additive stream cipher



- Suitable in environments characterized by a limited computing power or memory, and the need to encrypt at high speed
- The seed of the keystream generators constitutes the secret key
- Security depends on
    - Pseudorandomness of the keystram k_i
    - Properties of the underlying functions that form the keystream generator

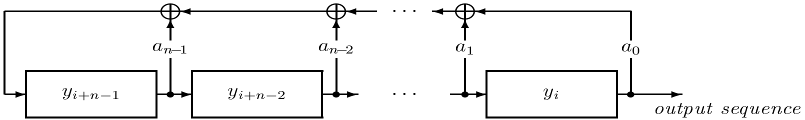
The optimal cipher: one-time pad

Description

- If $M = m_1 m_2 \ldots m_n$, then $k = k_1 k_2 \ldots k_n$ satisfying
  - k is trully random
  - k is aperiodic
  - For each different message, we use different key
- Encryption: $c_i = m_i \oplus k_i$, $i = 1, 2, \ldots, n$
- Decryption: $m_i = c_i \oplus k_i$, $i = 1, 2, \ldots, n$
- Such cipher is perfectly secure (Claude Shannon - 1949)
  - $p(M|C) = p(M)$ for any pair M, C
- However both randomness as well as aperiodicity can not be ensured in a realistic model
- Designing of stream ciphers strives to resemble the one-time pad

Keystream Generators in stream ciphers

Basic building block: Linear Feedback Shift Register (LFSR)



Output sequence:

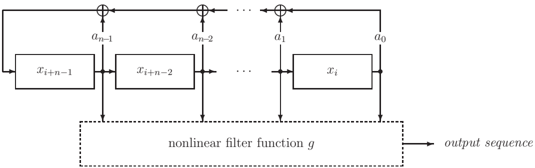
$$y_{i+n} = a_{n-1}y_{i+n-1} + \cdots + a_1 y_{i+1} + a_0 y_i,\ a_j \in \{0,1\}\ \forall\ j = 0,1,\ldots,n-1$$

- Easy implementation
- Nice mathematical properties
- But: The derived keystreams are easily predictable and, thus, cryptographically weak

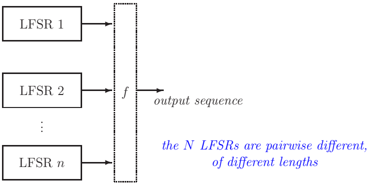
Predictability of keystreams: Linear complexity

- Linear complexity of a sequence: The length of the shortest LFSR that generates the sequence
- If the length of the keystream is N and its linear complexity is L, then the shortest LFSR is unique if and only if $L \leq \frac{N}{2}$
- Berlekamp-Massey algorithm: Efficient recursive computation of the shortest LFSR that generates a given sequence (Total complexity: $\mathcal{O}(N^2)$)
    - The same algorithm is also used for decoding famous error-control codes (BCH/Reed-Solomon codes)
- Knowledge of $2L$ consecutive bits of the keystream suffices to generate the remainder!!
    - $\Rightarrow$ High linear complexity is prerequisite in keystreams
    - Appropriate use of nonlinear functions

Classical Keystream Generators



(a) Nonlinear filter generator



(b) Nonlinear combiner generator

- High linear complexity is ensured by appropriately choosing the underlying Boolean functions
- If these functions though do not satisfy certain properties, the system may be vulnerable to attacks
- More recently, nonlinear FSRs are preferred (although their mathematics are not well-known)

Known stream ciphers

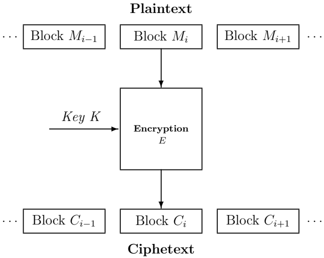
- RC4
  - Used in WEP, WPA, SSL/TLS
- A5/1
  - Used in mobile telephony (GSM)
- E0
  - Used in Bluetooth protocol

eStream project (2004–2008): Effort to promote the design of efficient and compact stream ciphers suitable for widespread adoption.

- Finalists:
  - Software implementation: HC-128, Rabbit, Salsa20/12, SOSEMANUK
  - Hardware implementation: Grain v1, MICKEY 2.0, Trivium

Block ciphers

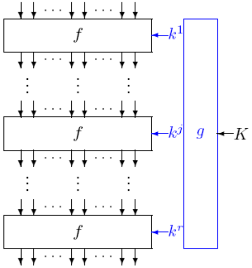
Typical Case: Electronic Codebook Mode (ECB)



- Encryption on a per-block basis (typical block size: 128 bits)
- The encryption function E performs key-dependent substitutions and permutations (Shannon's principles)
- Security depends on
  - Generation of the sub-keys used in E
  - Properties of the underlying functions of E

The encryption function E in a block cipher

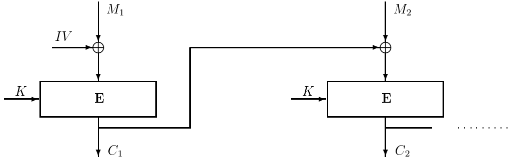
- Iterative structure
  - Several rounds occur
- A sub-key is being used in each round
- The round function f performs substitution and permutations, via multi-output Boolean functions (S-boxes, P-boxes)
  - S-boxes and P-boxes provide the cryptographic properties of diffusion and confusion respectively (Claude Shannon - 1949)



Modes of operations for block ciphers

- In ECB mode, two identical message blocks are encrypted into identical ciphetext clocks
- Other modes of operation alleviate this issue: CBC, CFB, OFB, CTR modes
  - CBC may also used for constructing hash functions
  - CFB, OFB and CTR transform a block cipher into a (powerful) stream cipher

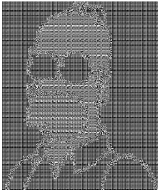
CipherBlock Chaining Mode of operation (CBC)

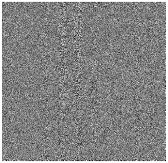


ECB vs CBC

An example plaintext

Encrypted with AES in ECB and CBC mode





- Repeated patterns in a plainxtext is a realistic assumption, as shown in the example (*obtained from a Bart Preneel's presentation, available online (MITACS, Toronto, 2010)*)
- Hence, ECB mode is of limited use

Known block ciphers

- Advanced Encryption Standard (AES)
  - NIST's standard since 2001 (initial submission: Rijndael cipher)
  - Supported key lengths: 128, 192, 256 bits
  - Widespread adoption (SSL/TLS, IPSec, commercial products,...)
- Data Encryption Standard (DES)
  - The predecessor of AES (1976-1996)
  - Official withdrawing: 2004 (although it is still being met today)
  - Key size: 56 bits (actually, the only flaw of the algorithm)
- 3DES
  - Modification of DES, to use key of 112 or 168 bit length
  - Still in use today - although not very efficient
- Other block ciphers: IDEA, MARS, RC6, Serpent, Twofish

A common approach for block and stream ciphers

- Despite their differences, a common study is needed for their building blocks (multi-output and single-output Boolean functions respectively)
- The attacks in block ciphers are, in general, different from the attacks in stream ciphers and vice versa. However:
  - For both cases, almost the same cryptographic criteria of functions should be in place
- Challenges:
  - There are tradeoffs between several cryptographic criteria
  - The relationships between several criteria are still unknown
  - Constructing functions satisfying all the main criteria is still an open problem

Boolean Functions

A Boolean function f on n variables is a mapping from $\mathbb{F}_2^n$ onto $\mathbb{F}_2$

- The vector $f = \big(f(0,0,\ldots,0), f(1,0,\ldots,0),\ldots,f(1,1,\ldots,1)\big)$ of length 2^n is the truth table of f
- The Hamming weight of f is denoted by $\mathrm{wt}(f)$
  - f is balanced if and only if $\mathrm{wt}(f) = 2^{n-1}$
- The support $\mathrm{supp}(f)$ of f is the set $\{\boldsymbol{b} \in \mathbb{F}_2^n : f(\boldsymbol{b}) = 1\}$

Example: Truth table of balanced f with $n = 3$

x_1	0	1	0	1	0	1	0	1
x_2	0	0	1	1	0	0	1	1
x_3	0	0	0	0	1	1	1	1
$f(x_1,x_2,x_3)$	0	1	0	0	0	1	1	1

A vectorial Boolean function f on n variables is a mapping from $\mathbb{F}_2^n$ onto $\mathbb{F}_2^m$, $m > 1$

Algebraic Normal Form and degree of functions

- Algebraic Normal Form (ANF) of f:

$$f(x) = \sum_{\boldsymbol{v} \in \mathbb{F}_2^n} a_{\boldsymbol{v}} x^{\boldsymbol{v}}, \quad \text{where } x^{\boldsymbol{v}} = \prod_{i=1}^{n} x_i^{v_i}$$

.

  - The sum is performed over $\mathbb{F}_2$ (XOR addition)
- The degree $\deg(f)$ of f is the highest number of variables that appear in a product term in its ANF.
- If $\deg(f) = 1$, then f is called affine function
  - If, in addition, the constant term is zero, then the function is called linear
- In the previous example: $f(x_1, x_2, x_3) = x_1x_2 + x_2x_3 + x_1$.
- $\deg(f) = 2$

Univariate representation of Boolean functions

- $\mathbb{F}_2^n$ is isomorphic to the finite field $\mathbb{F}_{2^n}$,
- $\Rightarrow$ Any function $f \in \mathbb{B}_n$ can also be represented by a univariate polynomial, mapping $\mathbb{F}_{2^n}$ onto $\mathbb{F}_2$, as follows

$$f(x) = \sum_{i=0}^{2^n-1} \beta_i x^i$$

where $\beta_0, \beta_{2^n-1} \in \mathbb{F}_2$ and $\beta_{2i} = \beta_i^2 \in \mathbb{F}_{2^n}$ for $1 \leq i \leq 2^n - 2$

- The coefficients of the polynomial determine the Discrete Fourier Transform of f
- The univariate representation is more convenient in several cases

Correlation immunity

- If the output of a Boolean function f is correlated to at least one of its inputs, then it is vulnerable to correlation attacks.
- The $f \in \mathbb{B}_n$ is t-th correlation immune if it is not correlated with any t-subset of $\{x_1, \ldots, x_n\}$; namely if

$$Pr(f(\boldsymbol{x}) = 0 | x_{i_1} = b_{i_1}, \ldots, x_{i_t} = b_{it}) = Pr(f(\boldsymbol{x}) = 0)$$

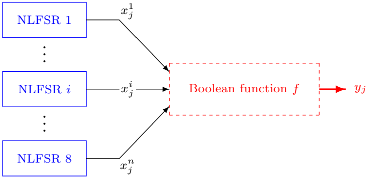
  for any t positions $x_{i_1}, \ldots, x_{i_t}$ and any $b_{i_1}, \ldots, b_{i_t} \in \mathbb{F}_2$
- If a t-th order correlation immune function is also balanced, then it is called t-th order resilient.
- A known trade-off: If f is k-th order resilient for $1 \leq k \leq n-2$, then $\deg(f) \leq n - k - 1$.

Linear approximation attacks

- Cryptographic functions need to be balanced, as well as of high degree
  - The maximum possible degree of a balanced Boolean function with n variables is $n-1$
- High degree though is not adequate to prevent linear cryptanalysis (in block ciphers - Matsui, 1992) or best affine approximation attacks (in stream ciphers - Ding et. al., 1991)
  - A function should not be well approximated by a linear/affine function
  - Any function of degree 1 that best approximates f is a best affine/linear approximation of f

Example of approximation attacks

The Achterbahn cipher [Gammel-Göttfert-Kniffler,2005] (candidate in eSTREAM project)



- Lengths of nonlinear FSRs: 22-31
- $f(x_1, \ldots, x_8) = \sum_{i=1}^{4} x_i + x_5x_7 + x_6x_7 + x_6x_8 + x_5x_6x_7 + x_6x_7x_8$
- Johansson-Meier-Muller, 2006: cryptanalysis via the linear approximation $g(x_1, \ldots, x_8) = x_1 + x_2 + x_3 + x_4 + x_6$, satisfying $\mathrm{wt}(f+g) = 64$ $(p(f=g) = 3/4)$

The notion of nonlinearity

- The minimum distance between f and all affine functions is the nonlinearity of f:

$$\mathsf{nl}(f) = \min_{l \in \mathbb{B}_n : \deg(l)=1} \mathrm{wt}(f+l)$$

- Nonlinearity is computed via the Fast Walsh Transform
- High nonlinearity is prerequisite for thwarting attacks based on affine (linear) approximations
- Constructions of correlation-immune functions with high nonlinearity exist (Maiorana-McFarland class (Camion-Carlet-Charpin-Sendrier, 1992),...)

Known results on nonlinearity of Boolean functions

- For even n, the maximum possible nonlinearity is $2^{n-1} - 2^{n/2-1}$, achieved by the so-called bent functions
  - Several constructions of bent functions are known
  - But bent functions are never balanced!
- For odd n, the maximum possible nonlinearity is still unknown
  - By concatenating bent functions, we can get nonlinearity $2^{n-1} - 2^{\frac{n-1}{2}}$. Can we impove this?
    - For $n \leq 7$, the answer is no
    - For $n \geq 15$, the answer is yes (Patterson-Wiedemann, 1983 - Dobbertin, 1995 - Maitra-Sarkar, 2002)
    - For $n = 9, 11, 13$, such functions have been found more recently (Kavut, 2006)
- Several constructions of balanced functions with high nonlinearity exist. However:
  - Finding the highest possible nonlinearity of balanced Boolean functions is still an open problem

Higher-order nonlinearity

- Approximating a function by a low-order function (not necessarily linear) may also lead to cryptanalysis (Non–linear cryptanalysis - Knudsen-1996, low-order approximation attacks - Kurosawa et. al. - 2002)
- The rth order nonlinearity of a Boolean function $f \in \mathbb{B}_n$ is given by

$$\mathsf{nl}_r(f) = \min_{g \in \mathbb{B}_n : \deg(g) \leq r} \mathrm{wt}(f+g)$$

- The rth order nonlinearity remains unknown for $r > 1$
  - Recursive lower bounds on $\mathsf{nl}_r(f)$ (Carlet, 2008)
  - Specific lower and upper bounds for $\mathsf{nl}_2(f)$ (Cohen, 1992 - Carlet, 2007)
  - More recent lower bounds for 2-nd order nonlinearity: Gangopadhyay et. al. - 2010, Garg et. al. - 2011, Singh - 2011, Singh et. al. - 2013

Computing best low order approximations

- Computing even the best 2-nd order approximations is a difficult task
  - Efficient solution for specific class of 3-rd degree functions (Kolokotonis-Limniotis-Kalouptsidis, 2009)
    - The problem is appropriately reduced in computing best affine approximation attacks of the underlying 2-nd degree sub-functions
  - For the Achterbahn's combiner function: $q(x) = x_5x_7 + x_6x_8 + x_1 + x_2 + x_3 + x_4$ is a best 2-nd approximation
  - $\mathrm{wt}(f+q) = 32$ $(p(f=q) = 7/8 > 3/4)$
- No much is known regarding constructions of functions with high r-th nonlinearity, for $r \geq 2$
  - A class of highly nonlinear 3-rd degree functions satisfying $\mathsf{nl}_2(f) = \mathsf{nl}(f)$ (Kolokotronis-Limniotis, 2012)

More recent attacks: Algebraic attacks

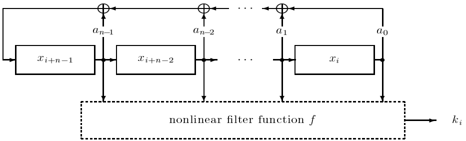
Milestones

- Algebraic attacks (Courtois-Meier, 2003)
- Fast algebraic attacks (Courtois, 2003)
- The basic idea is to reduce the degree of the mathematical equations employing the secret key
- Known cryptographic Boolean functions failed to thwart these attacks
- Some applications of algebraic attacks
  - Six rounds of DES, with only one known plaintext (Courtois-Bard, 2006)
  - Keeloq block cipher (Courtois-Bard-Wagner, 2008)
  - Hitag2 stream cipher (Courtois et. al., 2009)

Algebraic attacks

Example

- Stream cipher based on a nonlinear filter generator



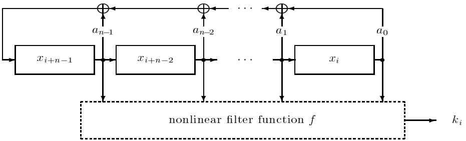
- $k_i = f(L^i(x_0, x_1, \ldots, x_{N-1}))$ - the filter function f has high degree
- Assume that there exists $g \in \mathbb{B}_n$ of low degree such that $f * g = h$, where h is also of low degree. Then,

$$k_i g(L^i(x_0, x_1, \ldots, x_{N-1})) = h(L^i(x_0, x_1, \ldots, x_{N-1}))$$

- Several other proper choices of g, h may also reduce the degree of the system

An example: The Toyocrypt cipher

- A submission to a Japanese government call



- The nonlinear filter function is

$$f(x_1,\ldots,x_{128}) = q(x_1,\ldots,x_{128}) + x_{10}x_{23}x_{32}x_{42} + \prod_{i=1}^{62} x_i +$$

$$+\, x_1x_2x_9x_{12}x_{18}x_{20}x_{23}x_{25}x_{26}x_{28}x_{33}x_{38}x_{41}x_{42}x_{51}x_{53}x_{59}$$

where $\deg(q) = 2$.

- By multiplying f with the affine functions $1 + x_{23}$ or $1 + x_{42}$, we get two functions of degree only 3

How to proceed with algebraic attacks

- Once the degree of the equations have been reduced, several algebraic techniques have been proposed for solving the (still nonlinear) system:
  - Linearization of the system
  - Use of Gröbner bases
  - More specific techniques: XL, XSL
- Hence, the core of the algebraic attacks is the transformation of the initial system to a new one having low degree

Annihilators and algebraic immunity

Definition

Given $f \in \mathbb{B}_n$, we say that $g \in \mathbb{B}_n$ is an annihilator of f if and only if g lies in the set

$$\mathcal{AN}(f) = \{g \in \mathbb{B}_n : f * g = 0\}$$

Definition

The algebraic immunity $\mathsf{AI}_n(f)$ of $f \in \mathbb{B}_n$ is defined by

$$\mathsf{AI}_n(f) = \min_{g \neq 0} \{\deg(g) : \ g \in \mathcal{AN}(f) \cup \mathcal{AN}(f+1)\}$$

- A high algebraic immunity is prerequisite for preventing algebraic attacks (Meier-Pasalic-Carlet, 2004)
- Well-known upper bound: $\mathsf{AI}_n(f) \leq \lceil \frac{n}{2} \rceil$

Some properties of algebraic immunity

Low nonlinearity implies low algebraic immunity: (Carlet *et. al.*, 2006)

$$\mathsf{nl}_r(f) \geq \sum_{i=0}^{\mathsf{AI}_n(f)-r-1} \binom{n}{i}$$

Especially for $r = 1$: (Lobanov, 2005)

$$\mathsf{nl}(f) \geq 2 \sum_{i=0}^{\mathsf{AI}_n(f)-2} \binom{n-1}{i}$$

Rizomiliotis, 2010: Improvements on the above bounds

- The notion of partial algebraic immunity is defined

Fast algebraic attacks

- Consider again the filter generator: $k_i = f(L^i(x_0, x_1, \ldots, x_{N-1}))$
- Assume that there exists a low degree $g \in \mathbb{B}_n$ such that $h = f * g$ is of reasonable degree. Then again,

$$k_i g(L^i(x_0, x_1, \ldots, x_{N-1})) = h(L^i(x_0, x_1, \ldots, x_{N-1}))$$

- There exists a linear combination of the first $\sum_{i=0}^{\deg(h)} \binom{N}{i}$ equations that sum the right-hand part to zero $\Rightarrow$ We get one equation of degree at most $\deg(g)$

Comparison with conventional algebraic attacks

- $g + h \in \mathcal{AN}(f) \Rightarrow$ the degree of $g + h$ may be greater than $\mathrm{AI}_n(f)$,
  - Maximum AI does not imply resistance to fast algebraic attacks
- But: Knowledge of consecutive keystream bits is required

Fast Algebraic Immunity

Known result: For any pair of integers (e, d) such that $e + d \geq n$, there exists a nonzero function g of degree at most e such that $f * g$ has degree at most d.

Definition

The fast algebraic immunity $\mathsf{FAI}_n(f)$ of $f \in \mathbb{B}_n$ is defined by

$$\mathsf{FAI}_n(f) = \min_{1 \leq \deg(g) \leq \mathsf{AI}_n(f)} \{2\,\mathsf{AI}_n(f), \deg(g) + \deg(f * g)\}$$

- Upper bound: $\mathsf{FAI}_n(f) \leq n$

Constructions of functions with maximum AI

- Dalai-Maitra-Sarkar, 2006: Majority function
  - For even n, a slight modification of the majority function also preserves maximum AI
- Carlet-Dalai-Gupta-Maitra-Sarkar, 2006: Iterative construction
- Li-Qi, 2006: Modification of the majority function
- Sarkar-Maitra, 2007: Rotation Symmetric functions of odd n
- Carlet, 2008: Based on properties of affine subspaces
  - Further investigation in Carlet-Zeng-Li-Hu, 2009
  - Generalization (for odd n) in Limniotis-Kolokotronis-Kalouptsidis, 2011
- Balanceness and/or high nonlinearity are not always attainable, whereas their fast algebraic immunity remains unknown

Constructions of functions with maximum AI (*Cont.*)

- Carlet-Feng, 2008: $\mathrm{supp}(f) = \{0, 1, \alpha, \alpha^2, \ldots, \alpha^{2^{n-1}-2}\}$, where α a primitive element of the finite field $\mathbb{F}_{2^n}$.
  - Balanceness and high (first-order) nonlinearity are ensured
  - Optimal against fast algebraic attacks, as subsequently shown (Liu-Zhang-Lin, 2012)
- Generalizations: Rizomiliotis (2010), Zeng-Carlet-Shan-Hu (2011)
  - Proper modifications of the Carlet-Feng construction (via the univariate representation of the function)
- Further generalizations in Limniotis-Kolokotronis-Kalouptsidis (2013)
  - Finding swaps between $\mathrm{supp}(f)$ and $\mathrm{supp}(f+1)$ that preserve maximum AI
- Still room for research regarding fast algebraic immunity (and r-th order nonlinearity for $r \geq 2$)

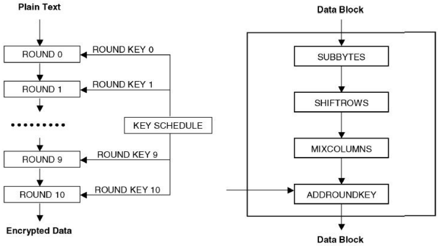
Current status

Open problems in the area

- Relationships between (fast) algebraic immunity and correlation immunity
  - The trade-off between correlation immunity and degree directly implies a trade-off between correlation immunity and fast algebraic immunity
- Evaluation of known families of cryptographic functions in terms of resistance against (fast) algebraic attacks
- Construction of functions with maximum (fast) algebraic immunity
  - Much progress on constructing functions with maximum AI, but the case of maximum FAI is much more difficult
- Nonlinear FSRs (or other nonlinear structures) have not been studied to the same extent

Design principles of AES

AES operation (key size=128 bits) - (Daemen-Rijmen, 1997)



- The S-box (SubByte) is a highly nonlinear function
- Designed to be resistant against all known cryptanalytic attacks
- The inherent algebraic structures (Murphy-Robshaw, 2002) do not allow mounting algebraic attacks

Recent attacks on AES

- The most important: A related-key attack for key lengths 192 and 256 bits (Biryukov et. al. - 2009)
  - Practical attacks for reduced number of rounds
    - Although such attacks are generally more theoretical than practical
  - It raises concern about the security margin of the AES
  - B. Schneier (2009): "*(...) for new applications I suggest that people don't use AES-256. AES-128 provides more than enough security margin for the foreseeable future. But if you're already using AES-256, there's no reason to change*".

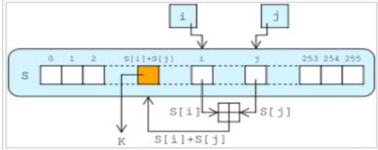
Side-channel cryptanalysis

Timing attacks in AES

- Cryptanalysis using additional information from the implementation of the algorithm
- The MixColumns function of AES may have different execution times, depending on the corresponding values
- Measuring the execution time provides information for the secret key
  - Most powerful timing attack on AES: Bernstein, 2005
- Conclusion: Not only mathematics, but implementation issues also need to be considered

Weaknesses depending on algorithm's parameters

The RC4 cipher (Rivest, 1987)



- i, j are updated via a specific procedure
- The weaknesses mainly stem from the keystream generator
  - The first bytes of K do not possess pseudorandom characteristics
- Proper choice of parameters is needed (key size, discarding first bytes of keystream,...)
- Academically not secure (distinguishing attacks)
  - Still secure though in practice (under a proper choice of parameters)
  - But....

Secure cipher does not imply secure protocol

RC4 in encryption protocols

- WEP is not considered as secure (see e.g. Tews et. al. - 2007: Breaking 104 bit WEP in less than 60 seconds)
  - Flaws rest with implementation (e.g. not proper choices of Initilization Vectors)
- AlFardan, Bernstein, Paterson, Poettering, Schuldt - 2013: Security of RC4 encryption in TLS and WPA/TKIP has been compromised
- From `www.isg.rhul.ac.uk/tls`:
  - The attacks in TLS arise from statistical flaws in the keystream generated
    - Most effective countermeasure: Stop using RC4 in TLS
  - One of the attacks also applies to WPA/TKIP
    - Most effective countermeasure: Upgrade to WPA2

Even AES may not provide a secure protocol

Attacks in IPSec

- IPSec provides security at the IP Layer (mainly used in Virtual Private Networks)
- Paterson-Yau, 2006 - Degabriel-Paterson, 2007: Active ciphertext-only attacks, if only encryption (and not data authentication) is implemented
  - Encryption with AES
  - Even following RFCs may not be enough!
- The attacks rest with the CBC mode of AES
  - Flipping bits in a ciphertext block leads to controlled changes in the subsequent decrypted plaintext block
  - Example: Appropriate modification of headers so that error messages, carrying plaintext data, are sent to attacker's machine

Conclusions

- Mathematics for ensuring cryptographic properties that are prerequisite to withstand any type of known attacks
  - Always leave a security margin - the attacks are getting better and better
- A secure cipher does not imply secure protocol
  - A proper design of the protocol is needed
- B. Schneier, Sep. 2013 (after disclosures of NSA eavesdropping on the Internet):
  - *"(...) Remember this: The math is good, but math has no agency. Code has agency, and the code has been subverted (...)"*
  - *"(...) Trust the math. Encryption is your friend. Use it well, and do your best to ensure that nothing can compromise it (...)"*

Questions & Answers

Thank you for your attention!