

Advanced Block Cipher Design

*My crazy boss asked me to design a new
block cipher. What's next?*

Pascal Junod

University of Applied Sciences
Western Switzerland

Outline

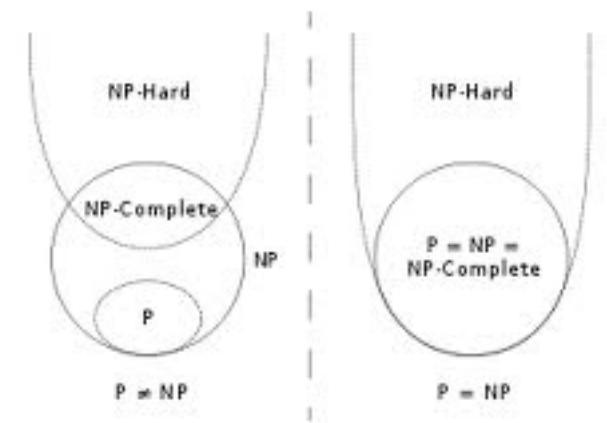
- High-Level Schemes
- Confusion
- Diffusion
- Key-Schedule
- Beyond the Design

Introduction



Some Simple Facts

- As of today, nobody knows how to design a (mathematically proven) secure block cipher.
- Problem related to fundamental open questions in mathematics / computer science
- A secure block cipher is a block cipher that nobody can break...
- A good block cipher is a secure block cipher that people like to implement.



So many Designs in the Wild...

Hierocrypt

G-DES

MacGuffin

LION

LOKI

RC2

Akellare

Coconut98

DFC

Square

E0

Twofish

Anubis

CAST

Skipjack

CS-Cipher

DEAL

Shark

RC5

Rijndael

IDEA

Camellia

Aria

Present

Noekeon

Magenta

DES-X

Threefish

RC6

Seed

Mars

FOX

Serpent

BassOmatic

GOST

DES

MESH

3-Way

E2

TEA

Blowfish

Misty

Triple DES

XTEA

Cipherunicorn

BEAR

CLEFIA

FEAL

heig-vd

Haute Ecole d'Ingénierie et de Gestion
du Canton de Vaud

XXTEA

Madryga

Designing a New Block Cipher

- Several good and bad reasons:
 - Faster / smaller than any other one ✓
 - With «better» security guarantees than any other one ✓✓
 - My boss crazily asked me to design a new, *secret (!) and patented (!!)* block cipher ~
 - Not enough proposals / diversity in the wild ✗
 - I desperately need to publish something to finish my PhD thesis ! ✗

Designing a New Block Cipher




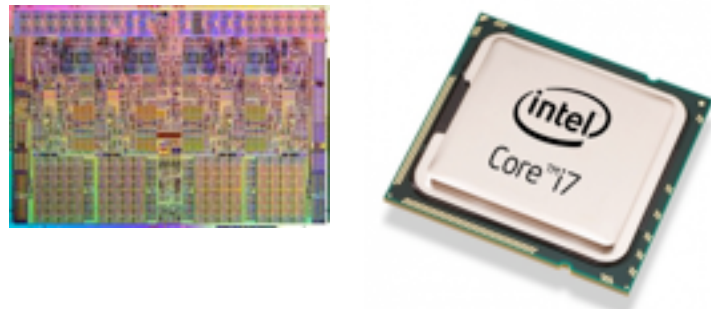
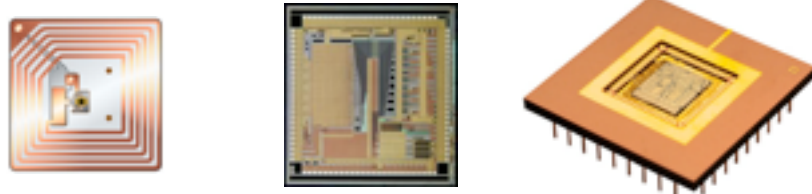
- Claude E. Shannon somewhat defined how to build a good cipher:

Two methods (other than recourse to ideal systems) suggest themselves for frustrating a statistical analysis. These we may call the methods of diffusion and confusion.

Designing a New Block Cipher

- Several decisions to take
 - Platform target
 - Security target
 - High-level scheme
 - Inner confusion / diffusion elements
 - Key-Schedule

Designing a New Block Cipher

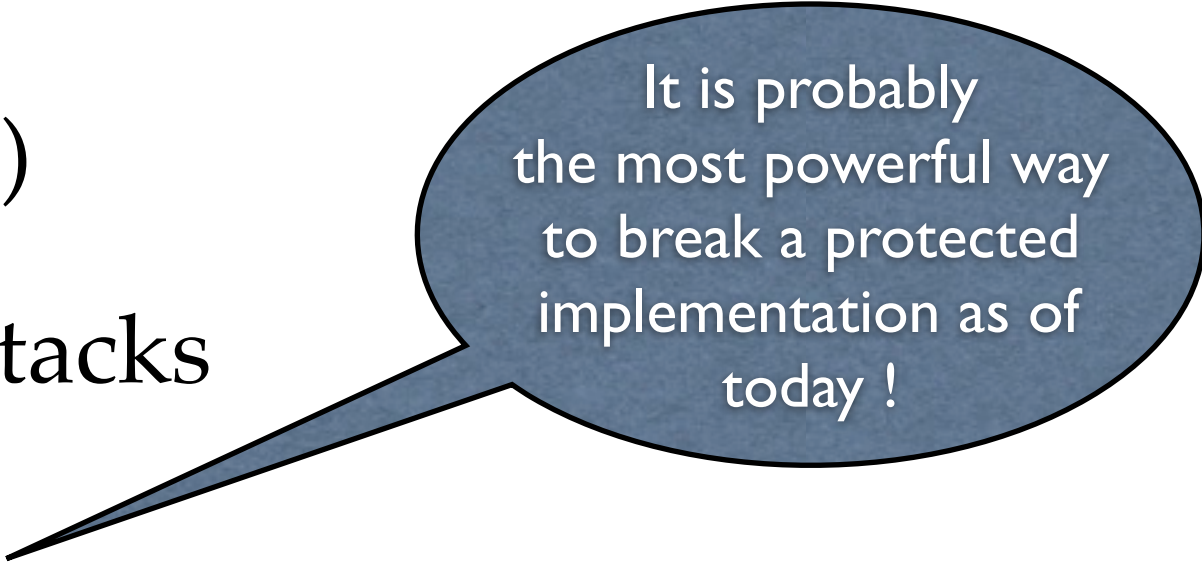
- Platform target
 - low-end CPU (4-bit, 8-bit, 16-bit, 32-bit micro-controller)
 - RAM / ROM / code size
 - high-end CPU (Intel / AMD / ...)
 - SIMD instructions / L1 cache size
- FPGA / ASIC
 - low / high gate / cells budget (RFID vs. high-speed encryption card)

Designing a New Block Cipher

- Security target (1)
 - Encryption
 - Authenticated encryption
 - Hashing
- Key size (... , 64, 80, 128, 256, 512, 1024, ...)
- Block size (... , 32, 48, 64, 96, 128, 256, 512, 1024, ...)

Designing a New Block Cipher

- Security target (2)
 - Side-channel attacks
 - Fault attacks
- (Resistance to reverse engineering, software emulation, ...)



It is probably
the most powerful way
to break a protected
implementation as of
today !

Designing a New Block Cipher

- High-Level Scheme
 - None (?)
 - Iterated
 - Feistel
 - Generalized Feistel
 - Substitution-Permutation Network
 - Lai-Massey

Designing a New Block Cipher

- Inner confusion / diffusion elements
 - Substitution boxes
 - Key-dependent non-linear operations
 - (Non-)linear diffusion layers

Designing a New Block Cipher

- Key-schedule algorithm
 - Light
 - Diffusive
 - Diffusive and non-linear
 - One-way
 - Efficient in both directions

High-Level Schemes



Iterated Schemes

- Main principle:

- Take a (rather weak) keyed permutation, i.e., a round function
- Iterate this function several times, by adding new randomness
- Hopefully get something more secure !
- Well illustrated e.g. by Vaudenay's decorrelation theory (information-theoretic setting) and Tessaro et al. (computational setting) very recent results

Security Amplification for the Cascade of Arbitrarily Weak PRPs:
Tight Bounds via the Interactive Hardcore Lemma

Stefano Tessaro
Department of Computer Science and Engineering
University of California, San Diego
9500 Gilman Drive
La Jolla, CA
stessaro@cs.ucsd.edu

Theorem 4. Let C_1, \dots, C_r be independent ciphers over \mathcal{M} . We consider $C = C_r \circ \dots \circ C_1$ the product cipher. We let C^* be the perfect cipher over \mathcal{M} . For the distance D defined by either $\|\cdot\|_2$, $\|\cdot\|_\infty$, $\|\cdot\|_a$, or N_∞ we have

$$D([C]^d, [C^*]^d) \leq \prod_{i=1}^r D([C_i]^d, [C^*]^d).$$

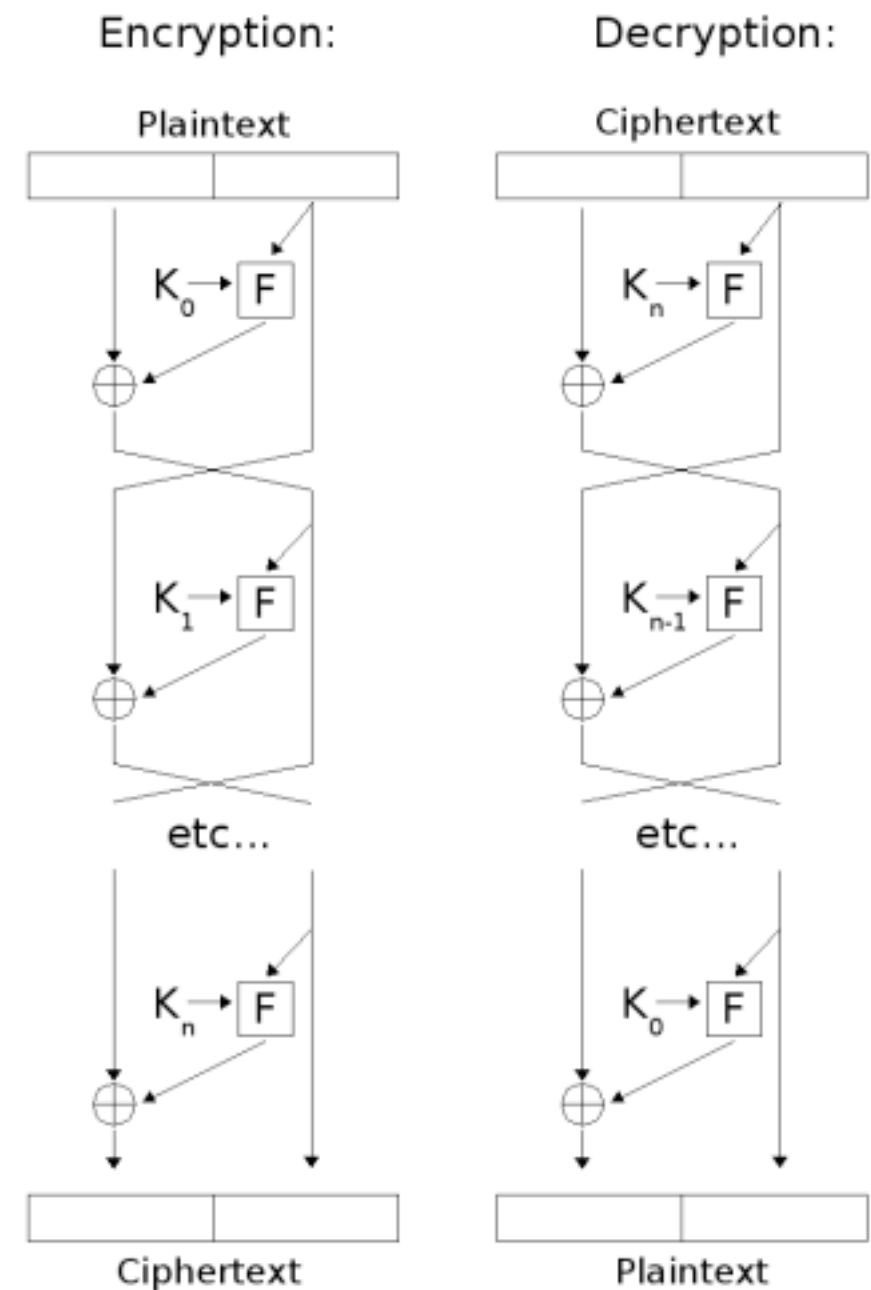
Iterated Schemes

- Well-known «Zürcher» cryptographer joke:
- «*Most ciphers are secure after sufficiently many rounds*» (L. O'Connor)
- «*Most ciphers are too slow after sufficiently many rounds*» (J. Massey)



Feistel Scheme

- Feistel Scheme (aka Feistel Network, Feistel Cipher, ...)
- Named after his inventor, Horst Feistel
- Scheme behind the DES
- Allow to transform any (possibly non-invertible function) in a permutation



Feistel Cipher

Feistel Scheme

- Has «provable security» properties [LubyRackoff, Patarin,...]
- PRP after 3 (7) rounds and less than $O(2^{\frac{n}{2}})$ ($O(2^{n(1-\varepsilon)})$) queries
- SPRP after 4 (10) rounds and less than $O(2^{\frac{n}{2}})$ ($O(2^{n(1-\varepsilon)})$) queries

How to Construct Pseudo-random Permutations
from Pseudo-random Functions

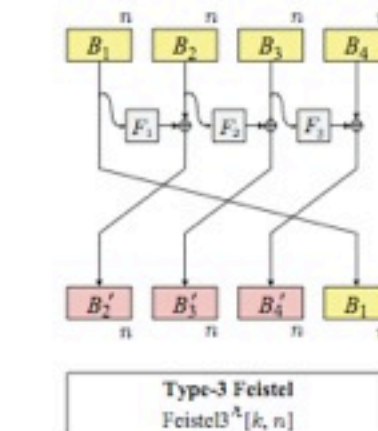
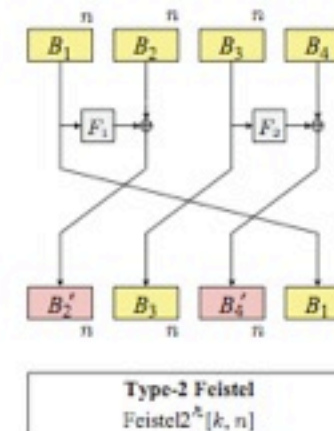
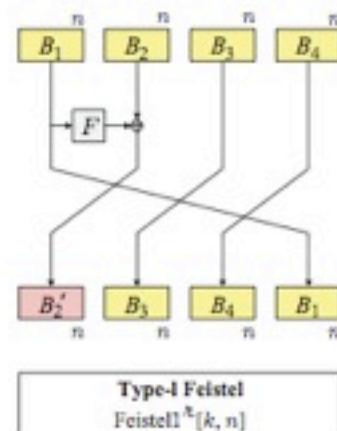
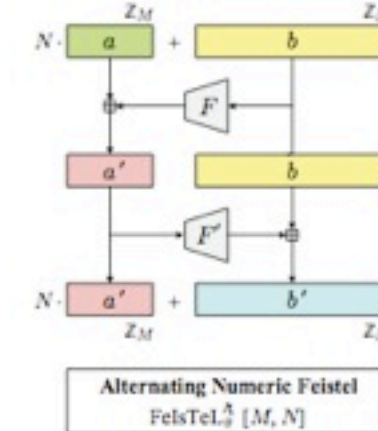
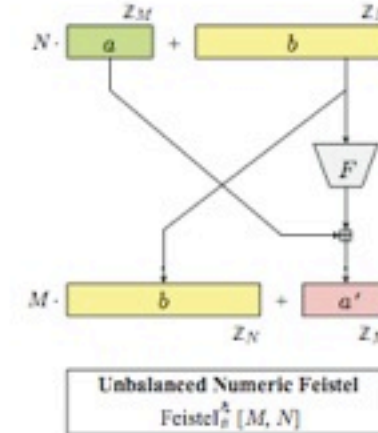
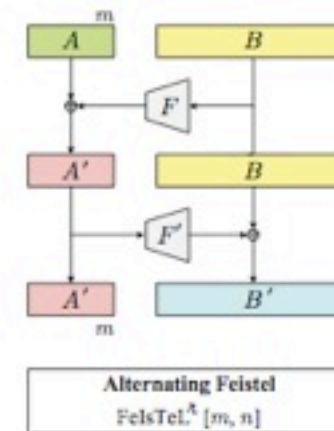
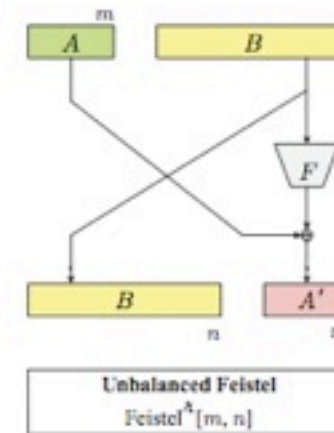
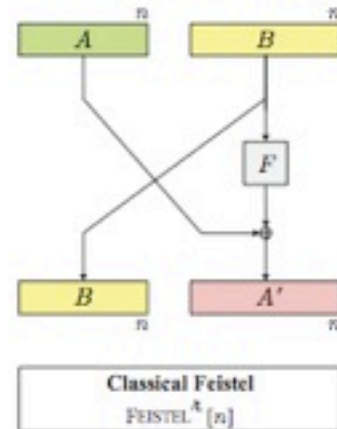
Michael Luby
Charles Rackoff
Department of Computer Science
University of Toronto
Toronto, Canada M5S 1A4

Luby-Rackoff: 7 Rounds Are Enough for $2^{n(1-\varepsilon)}$
Security

Jacques Patarin
University of Versailles

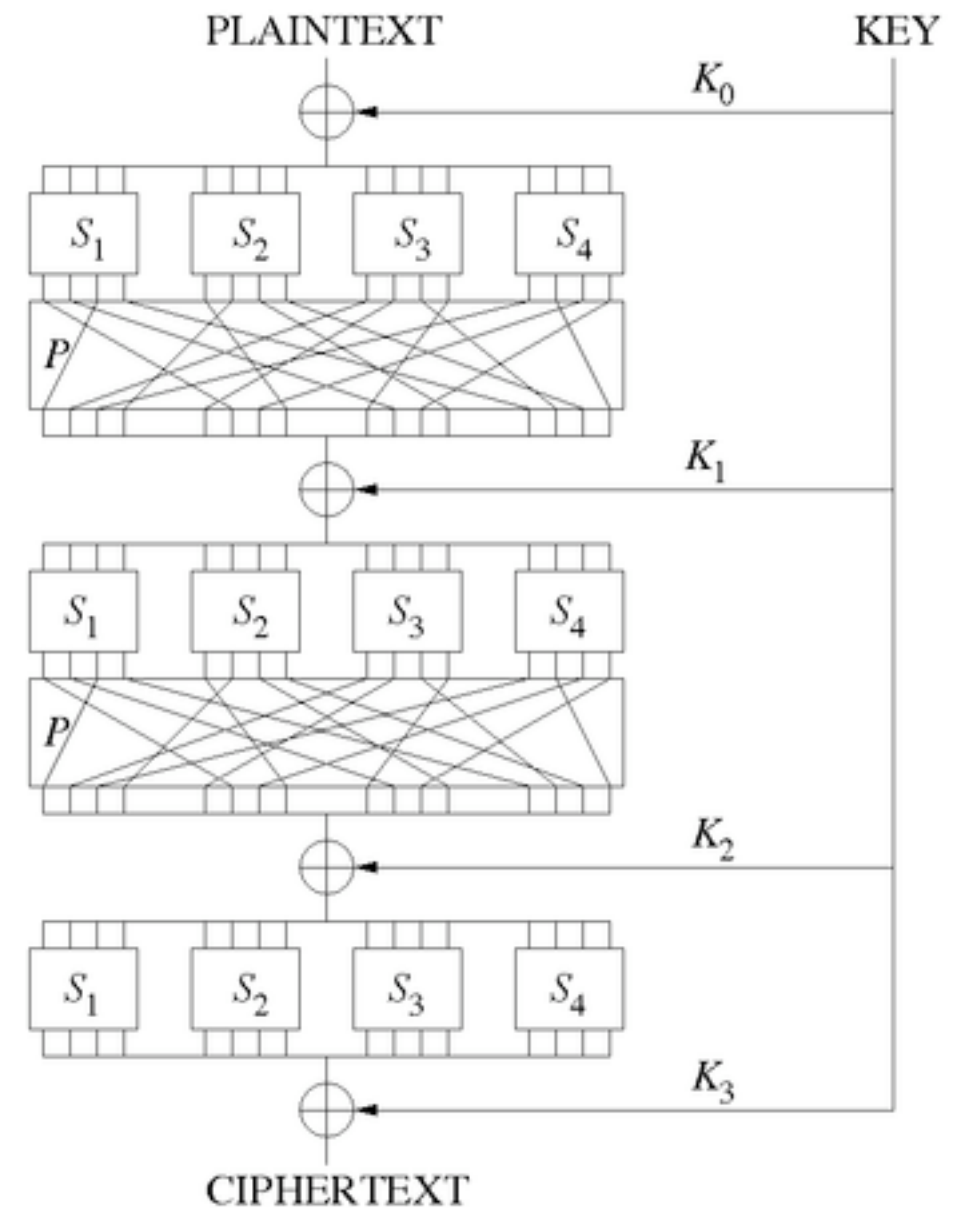
Generalized Feistel Schemes

- Many, many different variants (see e.g. [HoangRogaway-2010])
- Rather slow diffusion



Substitution Permutation Networks

- Used by AES, Present, Square and many others.
- Works on the full cipher width
- Large body of literature available on its security towards various attacks (linear, differential, saturation, ...)



Lai-Massey Scheme

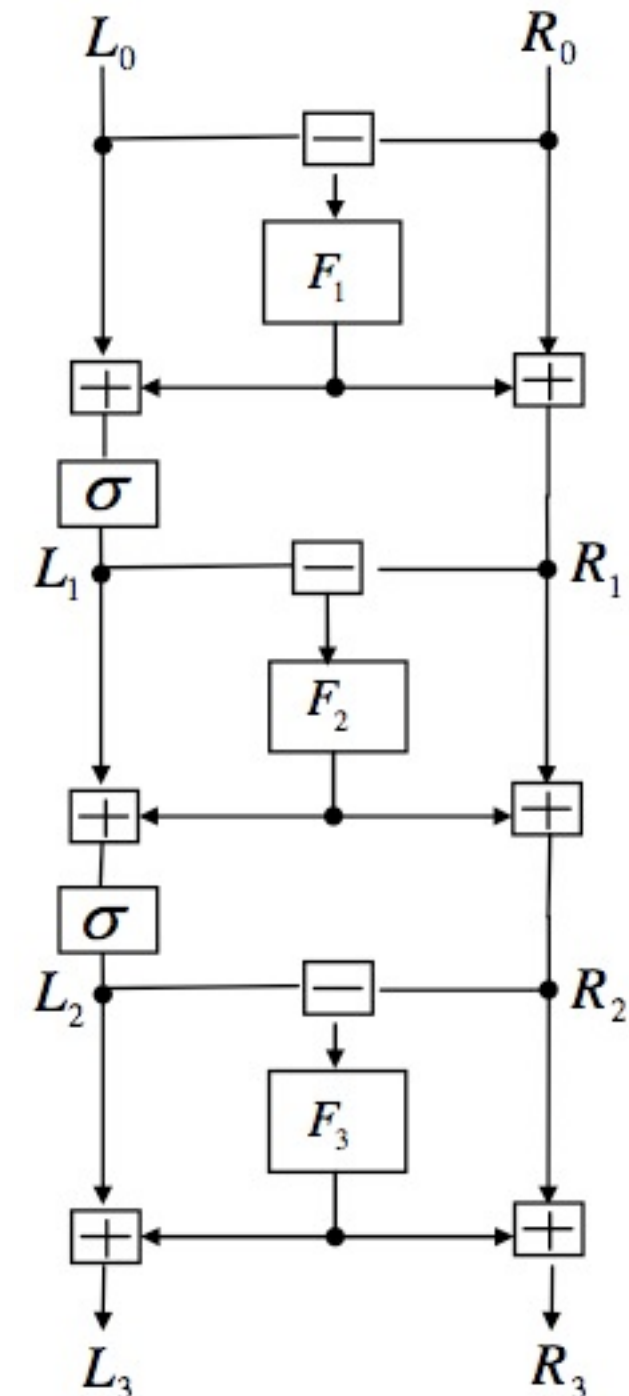
- High-level structure behind the IDEA cipher
- Recycled e.g. by FOX
- Has some provable properties (see e.g. [Vaudenay-1999])

Definition 1. In a given group G of order g , a permutation σ is called an α -almost orthomorphism if the function $\sigma'(x) = \sigma(x) - x$ is such that there are at most α elements in G with no preimage by σ' .

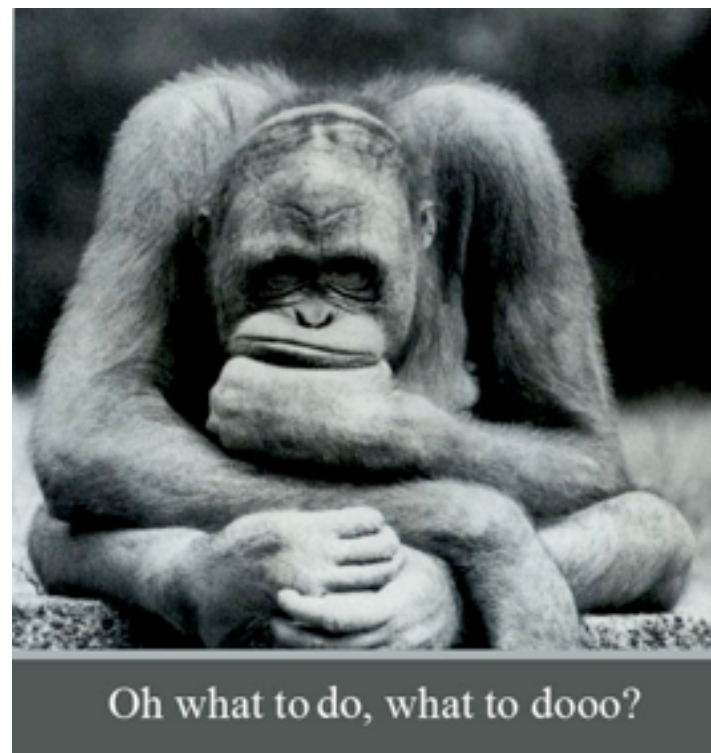
Theorem 4. Let F_1^*, F_2^*, F_3^* be three independent random functions on a group G with a uniform distribution. Let σ be an α -almost orthomorphism on G . For any distinguisher limited to d chosen plaintexts ($d < g^2$) between $\Lambda^\sigma(F_1^*, F_2^*, F_3^*)$ and a random permutation C^* with a uniform distribution, we have

$$\text{Adv}(\Lambda^\sigma(F_1^*, F_2^*, F_3^*), C^*) \leq d(d-1)(g^{-1} + g^{-2}) + f(\alpha)$$

where g is the cardinality of G and $f(\alpha)$ is defined as in Lemma 3.



Confusion



Substitution Boxes

- Substitution boxes
 - Non-linear mapping $n \longrightarrow m$ bits
 - Usual values:

3 \longrightarrow 3

4 \longrightarrow 4

6 \longrightarrow 4

7 \longrightarrow 7

8 \longrightarrow 8

9 \longrightarrow 9

8 \longrightarrow 32

S ₅		Middle 4 bits of input															
		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
Outer bits	00	0010	1100	0100	0001	0111	1010	1011	0110	1000	0101	0011	1111	1101	0000	1110	1001
	01	1110	1011	0010	1100	0100	0111	1101	0001	0101	0000	1111	1010	0011	1001	1000	0110
	10	0100	0010	0001	1011	1010	1101	0111	1000	1111	1001	1100	0101	0110	0011	0000	1110
	11	1011	1000	1100	0111	0001	1110	0010	1101	0110	1111	0000	1001	1010	0100	0101	0011

Substitution Boxes

- Main criteria to look at:
 - DP and LP coefficients
 - Algebraic degree
 - + many, many others...

Substitution Boxes

- Differential (Linear) Probability coefficient
- Measures the resistance of an S-box to differential (linear) cryptanalysis

Definition 1. Let $F_k(x)$ be a function with an n -bit input x and an ℓ -bit parameter k . We define average differential probability DP^F and average linear probability LP^F of the function F as

$$DP^F \stackrel{def}{=} \frac{1}{2^\ell} \sum_k \max_{\Delta x \neq 0, \Delta y} \frac{\#\{x | F_k(x) \oplus F_k(x \oplus \Delta x) = \Delta y\}}{2^n}, \quad (1)$$

$$LP^F \stackrel{def}{=} \frac{1}{2^\ell} \sum_k \max_{\Gamma x, \Gamma y \neq 0} \left(2^{\frac{\#\{x | x \bullet \Gamma x = F_k(x) \bullet \Gamma y\}}{2^n}} - 1 \right)^2, \quad (2)$$

respectively. We also apply this definition to a function $F(x)$ without the parameter k by setting $\ell = 0$.

Substitution Boxes

- Algebraic Degree
 - Measures the «complexity» of the Boolean equations representing the S-box
 - Is equal to the number of variables of the largest monomial in the polynomial representation of the S-box.

$$f(x) = \sum_{y \in \mathbb{F}_2^n} \hat{f}(y) x_0^{y_0} x_1^{y_1} \cdots x_{n-1}^{y_{n-1}}.$$

Definition 3. The algebraic degree $\deg(f)$ of a function $f : \mathbb{F}_2^n \mapsto \mathbb{F}_2$ is the maximal weight $wt(x)$ that satisfies $\hat{f}(x) \neq 0$.

Substitution Boxes

- Other criteria:
 - No single-bit difference
 - Efficient Boolean representation
 - Efficient Boolean representation of the inverse mapping
 - ...

Substitution Boxes

- How to find «good» S-boxes ?
- Three main approaches:
 - Random search
 - Algebraic construction
 - Iterated construction

Substitution Boxes

- Random search
 - Plug an AES in counter mode to a Knuth shuffle
 - Generate random permutations
 - Test for your preferred criteria
 - Repeat the process until you are happy !

```
To initialize an array a of n elements to a randomly shuffled copy of source, both 0-based:  
a[0] ← source[0]  
for i from 1 to n - 1 do  
  j ← random integer with 0 ≤ j ≤ i  
  a[i] ← a[j]  
  a[j] ← source[i]
```

Substitution Boxes

- Algebraic approach
 - Proposed by Nyberg in 1993
 - Used by AES, among many others
 - Example: inversion operation in $GF(2^8)$
 - Usually combined with an affine mapping over bits to break the algebraic structure
 - Might (???) cause troubles with respect to algebraic attacks

Differentially uniform mappings for cryptography

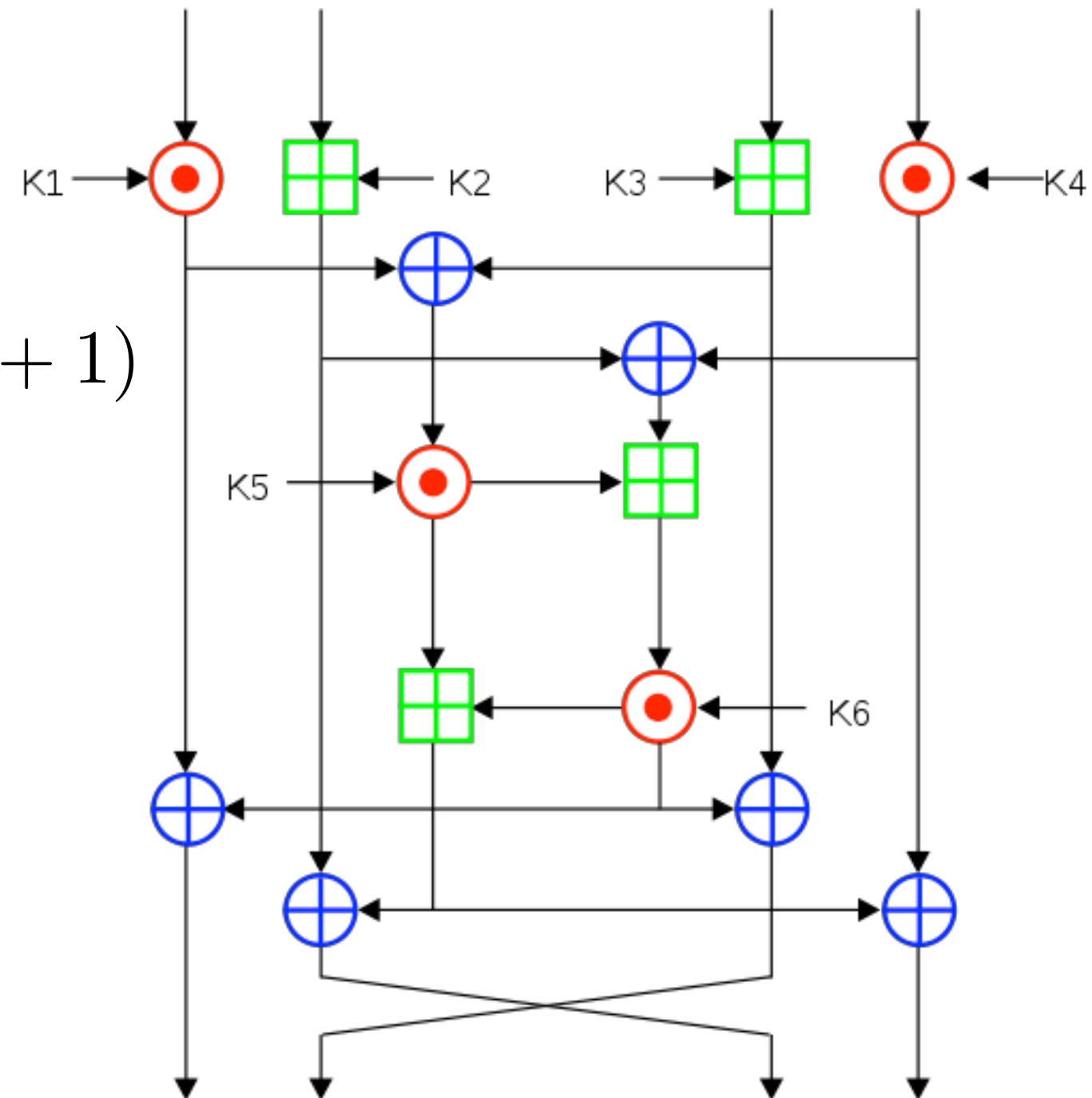
KAISA NYBERG*

Institute of Computer Technology, Vienna Technical University

Abstract. This work is motivated by the observation that in DES-like ciphers it is possible to choose the round functions in such a way that every non-trivial one-round characteristic has small probability. This gives rise to the following definition. A mapping is called differentially uniform if for every non-zero input difference and any output difference the number of possible inputs has a uniform upper bound. The examples of differentially uniform mappings provided in this paper have also other desirable cryptographic properties: large distance from affine functions, high nonlinear order and efficient computability.

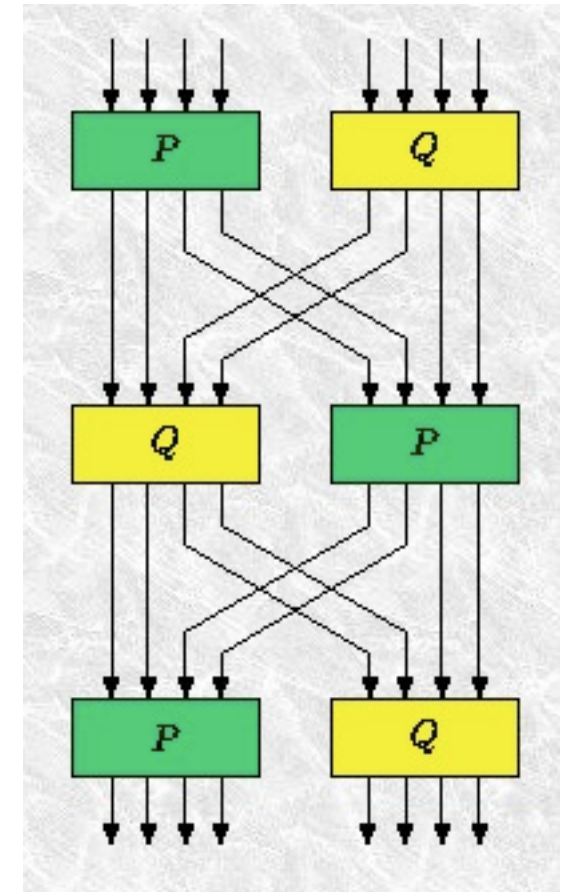
Key-Dependent Non-Linear Operations

- Example: IDEA
- Multiplication in $GF(2^{16} + 1)$
- Involves a subkey value
- Sensitive to weak key classes
- Nice down-scaling properties

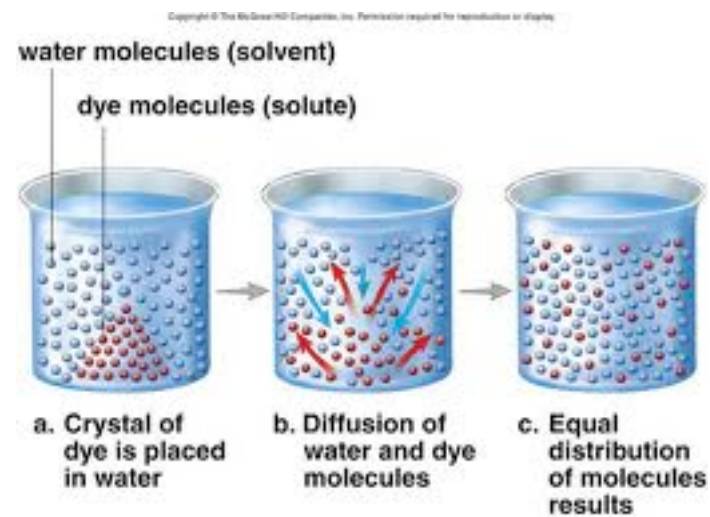


Iterated Construction

- Examples: Khazad, FOX
- Construct a large S-box out of smaller ones
- A few rounds of Feistel / SPN / Lai-Massey with smaller «good» S-boxes as round function
- «Nice» when implemented in hardware
 - Less GE, more delay



Diffusion



Strong Diffusion Layers

● Concept of multipermutation [Vaudenay]

Definition 4.1.4 (Multipermutation). *An (r, n) -multipermutation over an alphabet \mathcal{A} is a function f from \mathcal{A}^r to \mathcal{A}^n such that two different $(r + n)$ -tuples of the form $(x, f(x))$ cannot collide in any r positions.*

On the Need for Multipermutations: Cryptanalysis of MD4 and SAFER

Serge VAUDENAY

Laboratoire d'Informatique, URA 1327 du CNRS
Département de Mathématiques et d'Informatique
Ecole Normale Supérieure

LIENS - 94 - 23

November 1994

Strong Diffusion Layers

- Concept of branch number of a (diffusive) mapping [Daemen]

Definition 4.1.5 (Branch number). Let $\theta : \text{GF}(2^n)^m \rightarrow \text{GF}(2^n)^m$ be a mapping. Then

$$\mathfrak{B}(\theta) = \min_{x \neq 0} \{ \delta_W(x) + \delta_W(\theta(x)) \}$$

is called the branch number $\mathfrak{B}(\theta)$ of θ .

Definition 4.1.6 (Optimal mapping). A mapping

$$\theta : \text{GF}(2^n)^m \rightarrow \text{GF}(2^n)^m$$

is called optimal if $\mathfrak{B}(\theta) = m + 1$.

Strong Diffusion Layers

- Maximum Distance Separable (MDS) matrices
 - Square invertible matrix with elements of $\text{GF}(2^n)$
 - Every sub-matrix is non-singular
 - Maximum branch number equal to $n + 1$

Strong Diffusion Layers

- MDS matrices constructions

$$= \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & 2 & 3 & \dots & \ell \\ 1^2 & 2^2 & 3^2 & \dots & \ell^2 \\ 1^3 & 2^3 & 3^3 & \dots & \ell^3 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1^{\ell-2k-1} & 2^{\ell-2k-1} & 3^{\ell-2k-1} & \dots & \ell^{\ell-2k-1} \end{pmatrix}$$

- Parity-check matrix of a Reed-Solomon code

$$\begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix}$$

- Circulant matrices

- Hand-crafted matrices

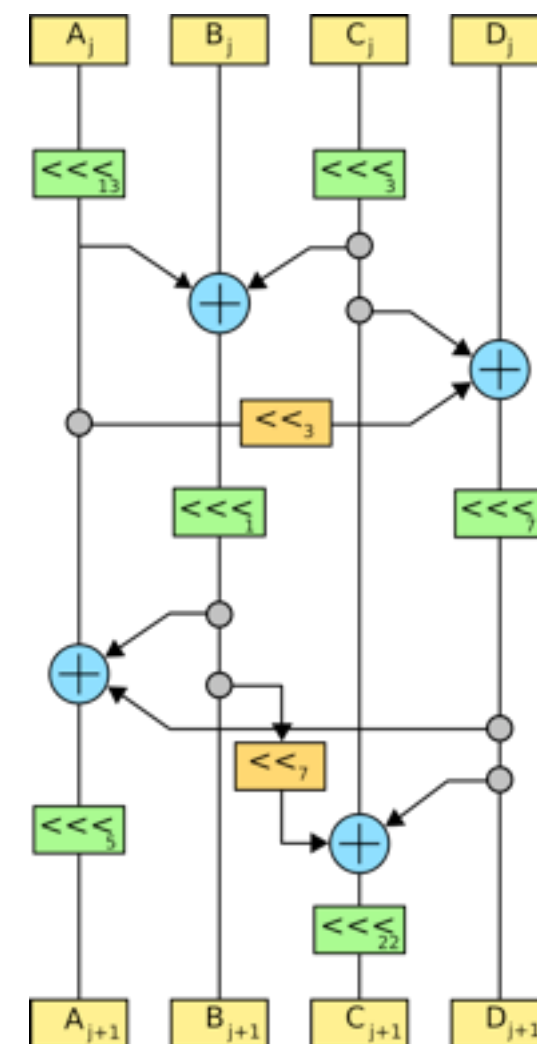
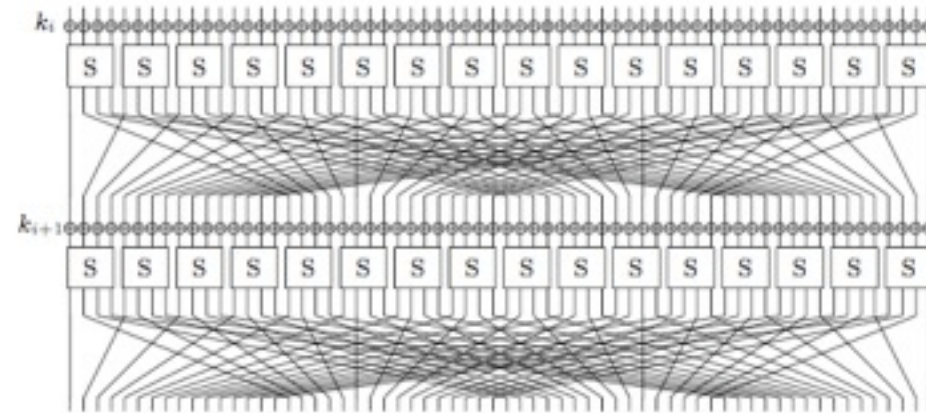
$$\begin{pmatrix} 1 & 1 & 1 & \alpha \\ 1 & z & \alpha & 1 \\ z & \alpha & 1 & 1 \\ \alpha & 1 & z & 1 \end{pmatrix} \quad \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & a \\ 1 & a & b & c & d & e & f & 1 \\ a & b & c & d & e & f & 1 & 1 \\ b & c & d & e & f & 1 & a & 1 \\ c & d & e & f & 1 & a & b & 1 \\ d & e & f & 1 & a & b & c & 1 \\ e & f & 1 & a & b & c & d & 1 \\ f & 1 & a & b & c & d & e & 1 \end{pmatrix}$$

- Cauchy matrices

- ...

Lighter Diffusion Layers

- Perfect diffusion
- Can be quite heavy to implement on constrained environments
- Alternative
 - Use lighter diffusion, but more rounds



Key-Schedule Algorithm

Key-Schedule Basics

- Responsible to derive several subkeys out of the master key
 - E.g., for AES128, derive eleven 128-bit round subkeys out of the 128-bit master key.
 - E.g., for IDEA, derive fifty-two 16-bit round subkeys out of the 128-bit master key.

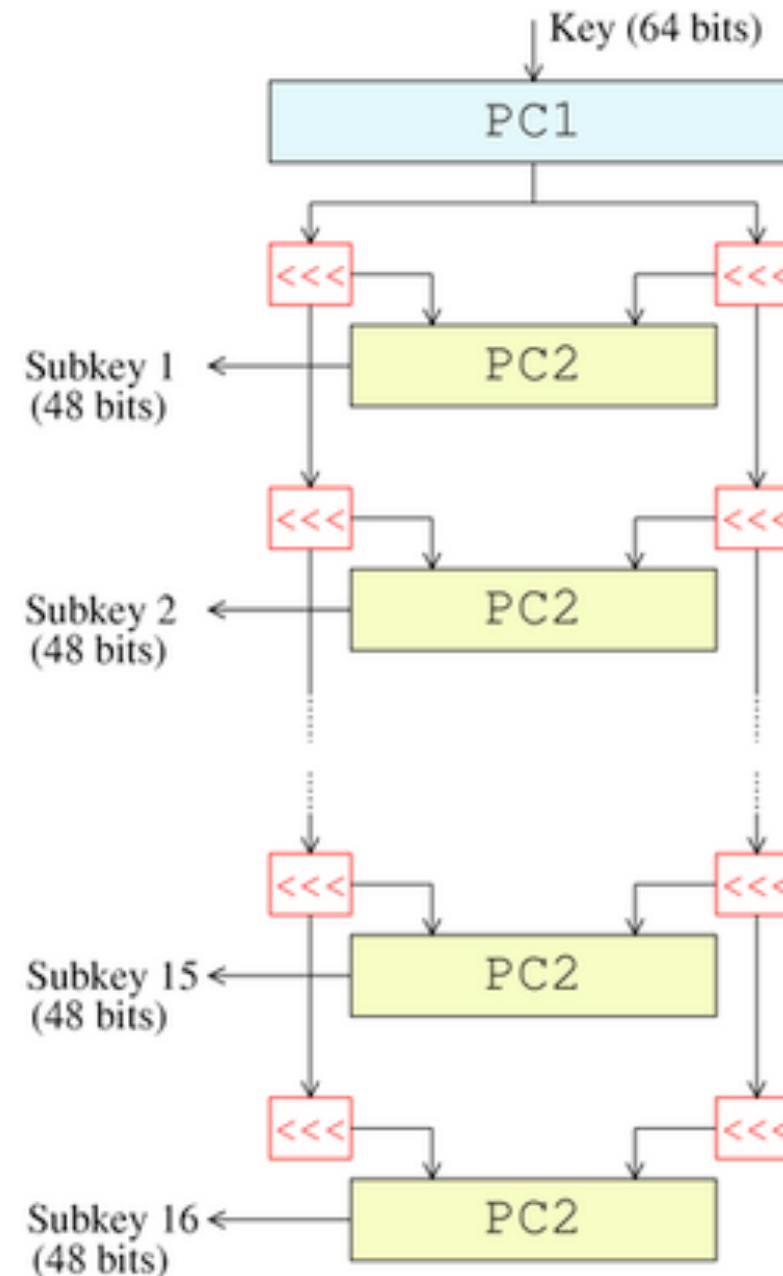
Light Key-Schedule

- GOST

«Break the 256-bit key into eight 32-bit subkeys, and each subkey is used four times in the algorithm; the first 24 rounds use the key words in order, the last 8 rounds use them in reverse order.»

Light Key-Schedule

- DES
- Two rotating registers
- Bit selection



Light Key-Schedule

● IDEA

● Bit selection through rotation of the key

Round r	$Z_1^{(r)}$	$Z_2^{(r)}$	$Z_3^{(r)}$	$Z_4^{(r)}$	$Z_5^{(r)}$	$Z_6^{(r)}$
1	$Z_{[0...15]}$	$Z_{[16...31]}$	$Z_{[32...47]}$	$Z_{[48...63]}$	$Z_{[64...79]}$	$Z_{[80...95]}$
2	$Z_{[96...111]}$	$Z_{[112...127]}$	$Z_{[25...40]}$	$Z_{[41...56]}$	$Z_{[57...72]}$	$Z_{[73...88]}$
3	$Z_{[89...104]}$	$Z_{[105...120]}$	$Z_{[121...8]}$	$Z_{[9...24]}$	$Z_{[50...65]}$	$Z_{[66...81]}$
4	$Z_{[82...97]}$	$Z_{[98...113]}$	$Z_{[114...1]}$	$Z_{[2...17]}$	$Z_{[18...33]}$	$Z_{[34...49]}$
5	$Z_{[75...90]}$	$Z_{[91...106]}$	$Z_{[107...122]}$	$Z_{[123...10]}$	$Z_{[11...26]}$	$Z_{[27...42]}$
6	$Z_{[43...58]}$	$Z_{[59...74]}$	$Z_{[100...115]}$	$Z_{[116...3]}$	$Z_{[4...19]}$	$Z_{[20...35]}$
7	$Z_{[36...51]}$	$Z_{[52...67]}$	$Z_{[68...83]}$	$Z_{[84...99]}$	$Z_{[125...12]}$	$Z_{[13...28]}$
8	$Z_{[29...44]}$	$Z_{[45...60]}$	$Z_{[61...76]}$	$Z_{[77...92]}$	$Z_{[93...108]}$	$Z_{[109...124]}$
8.5	$Z_{[22...37]}$	$Z_{[38...53]}$	$Z_{[54...69]}$	$Z_{[70...85]}$		

Stronger Key-Schedule

- AES

- First subkey is the key
- Non-linear Feedback Shift Register
- Recycling the AES S-box
- Use of round constants
- Possible to compute it sequentially in both directions
- Cost is less than one cipher

```
KeyExpansion(byte key[4*Nk], word w[Nb*(Nr+1)], Nk)
begin
  word temp

  i = 0

  while (i < Nk)
    w[i] = word(key[4*i], key[4*i+1], key[4*i+2], key[4*i+3])
    i = i+1
  end while

  i = Nk

  while (i < Nb * (Nr+1))
    temp = w[i-1]
    if (i mod Nk = 0)
      temp = SubWord(RotWord(temp)) xor Rcon[i/Nk]
    else if (Nk > 6 and i mod Nk = 4)
      temp = SubWord(temp)
    end if
    w[i] = w[i-Nk] xor temp
    i = i + 1
  end while
end
```

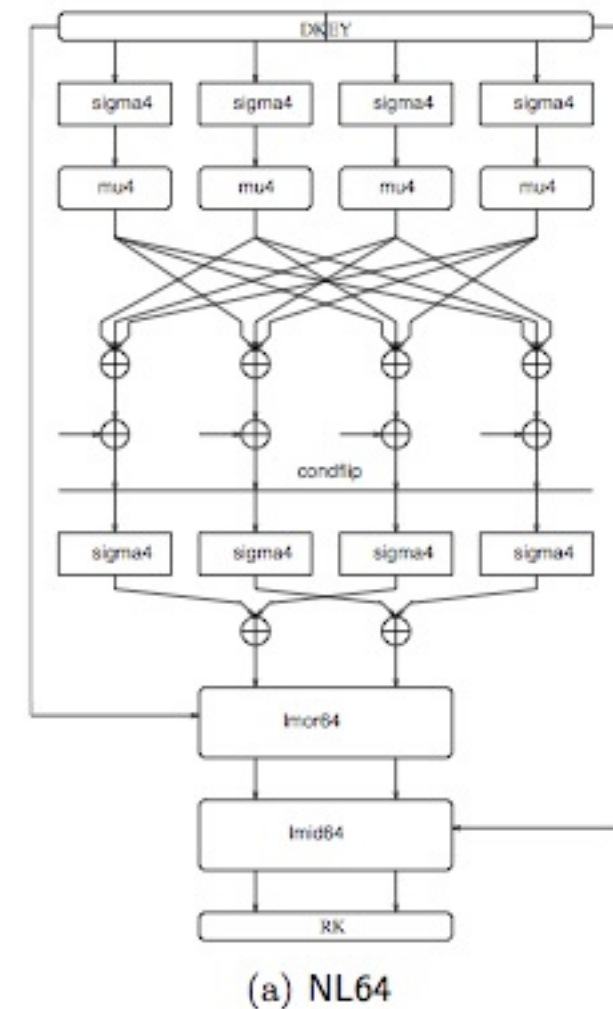
Note that $Nk=4, 6,$ and 8 do not all have to be implemented; they are all included in the conditional statement above for conciseness. Specific implementation requirements for the Cipher Key are presented in Sec. 6.1.

One-Way Key-Schedule

- Blowfish
 - Key-schedule is responsible to generate
 - Constants
 - S-boxes
 - Encryption core is recycled
 - Cost is up to 521 Blowfish iterations (!)

One-Way Key-Schedule

- FOX
- Requirements
 - Bi-directional without key processing
 - One-way
 - Not very (in-)efficient (the cost of about 6 encryptions)



Perfect Key-Schedule

- Theoreticians
 - Subkeys decorrelated from the key, statistically independent subkeys
 - One-way (e.g., leakage-resilient crypto)
- Implementers
 - Light, fast, small, easy to understand, free
 - Secure in all situations
- Depends on the cipher's use, too
 - Encryption vs. compression function

Beyond a Design



Security Analysis

- Designer has to provide (some) evidence of security against every possible known attack...
- «Provable security» towards
 - Differential cryptanalysis
 - Linear cryptanalysis
- Out of AES specifications:

We prove that the minimum number of active S-boxes in any 4-round differential or linear trail is 25. This gives a maximum prop ratio of 2^{-150} for any 4-round differential trail and a maximum of 2^{-75} for the correlation for any 4-round linear trail. This holds for all block lengths of Rijndael and is independent of the value of the Round Keys.

Security Analysis

- How not to get broken ?
 - Rely on bullet-proof components
 - High-level scheme
 - Confusion / diffusion elements
 - Double or triple the number of rounds that are supposed to resist linear and differential cryptanalysis
 - Be somewhat lucky !

Research Directions

- Field of «block ciphers» could / has become slightly boring...
- More and more difficult to find attacks in standard models
- More and more difficult to find new attack directions
- As of today, we know how to «engineer» a secure, general-purpose block cipher



Research Directions

- Ways to explore
 - Lightweight cryptography
 - More provable security for practical designs

KFC - The Krazy Feistel Cipher

Thomas Baignères* and Matthieu Finiasz

EPFL

CH-1015 Lausanne – Switzerland

<http://lasecwww.epfl.ch>

Abstract.

Theorem 9. Assume that the advantage of the best 2-limited distinguisher on

$F_{KFC[r_1, r_2]}$ is bounded by ϵ . For any d and set of integers $\{t_3, \dots, t_d\}$ such that

Extra Crispy KFC: $N = 8$, $q = 2^{16}$, $r_1 = 3$, $r_2 = 1000$. Using these quite extreme parameters, we manage to obtain provable security against 70-limited adaptive adversaries, but encryption rate could probably never reach more than 1 Mbit/s. Also, the key schedule should produce 2^{25} pseudo random bits, which means that Extra Crispy KFC requires at least 4 GB of memory.

$$\sum_{i=3}^d \left(1 - \left(1 - \frac{i-1}{q} \right)^N \right)^{t_i}.$$

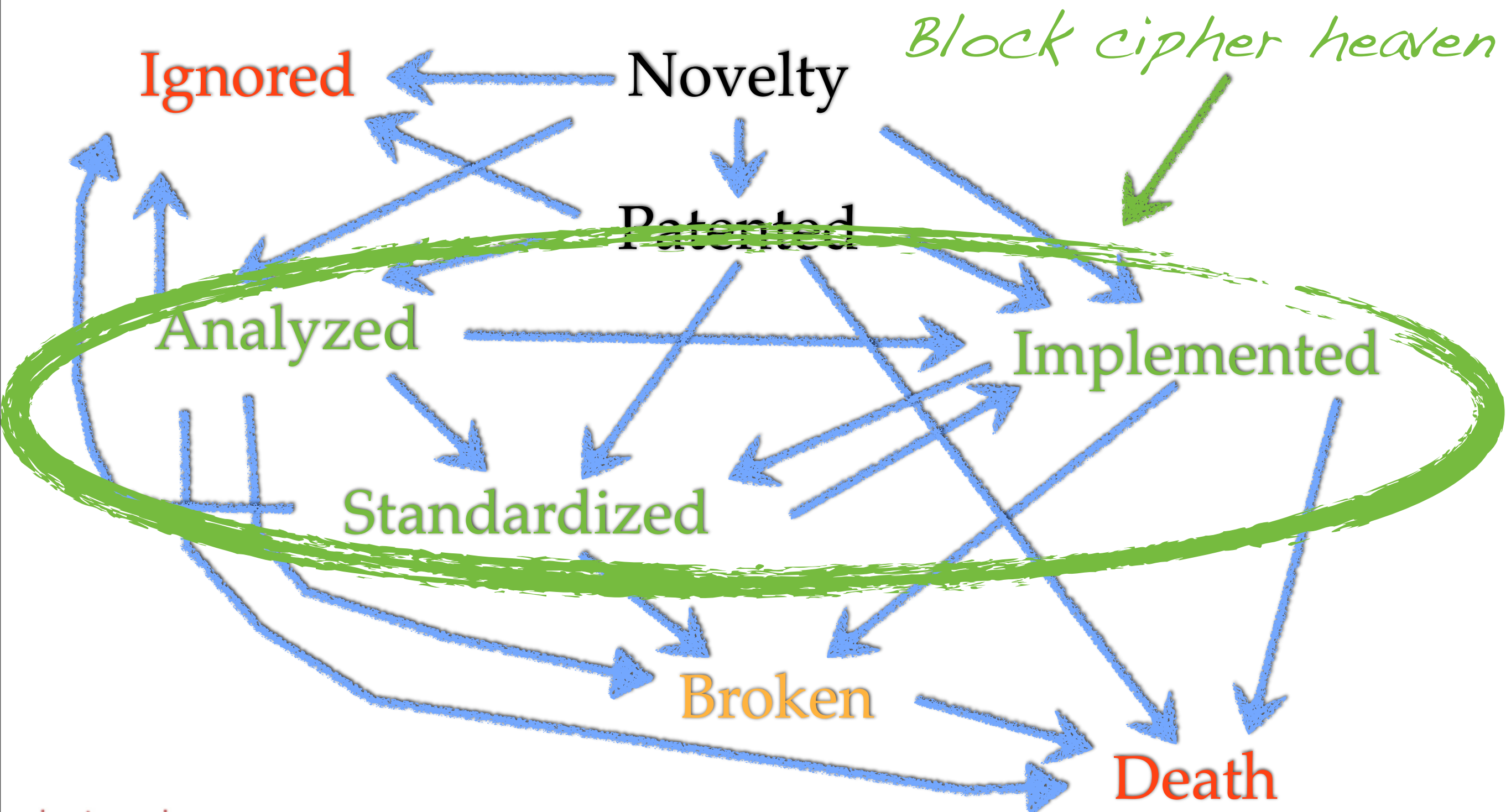
effect in co

sis, the boomerang attack, differential-linear cryptanalysis, and others).

Research Directions

- Other potential ways to explore
 - Efficient, large-block ciphers
 - Finding the perfect key-schedule
 - Intrinsically fault/leakage-resistant designs
 - Designs resistant to reverse-engineering (white-box cryptography)

Fate of a Block Cipher



Thank you !

Questions ?



Credits for pictures: shamelessly stolen from all over the Internet