Cryptanalysis, Reverse-Engineering and Design of Symmetric Cryptographic Algorithms

Léo Perrin

SnT, University of Luxembourg

April 25, 2017

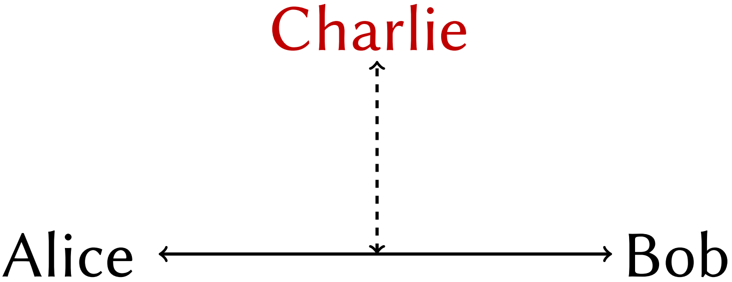
PhD Defence

Outline

Cryptography? (1/2)

Alice $\longleftrightarrow$ Bob

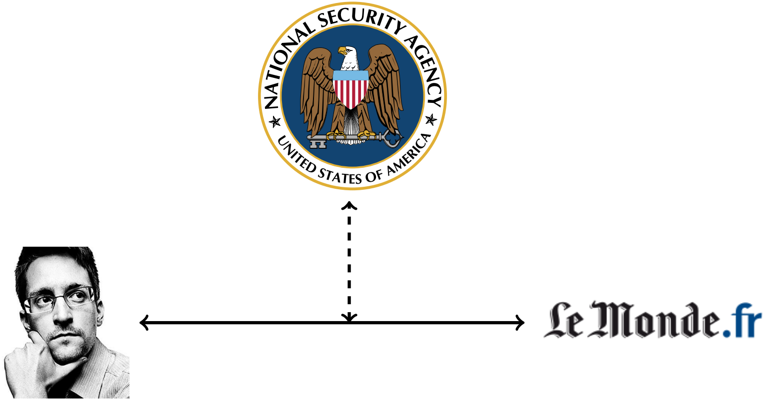
Cryptography? (1/2)



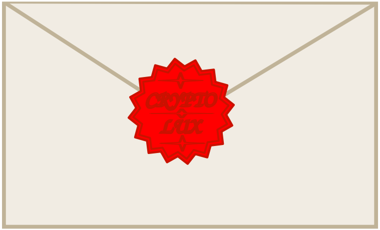
Cryptography? (1/2)

Charlie

Cryptography? (1/2)

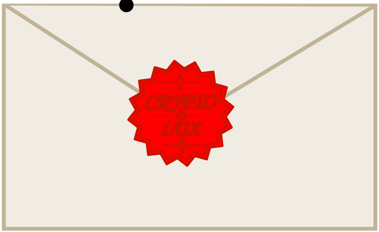


Cryptography? (2/2)

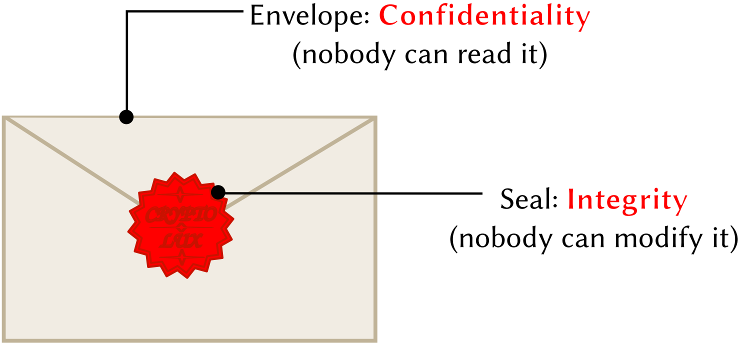


Cryptography? (2/2)

Envelope: **Confidentiality**
(nobody can read it)

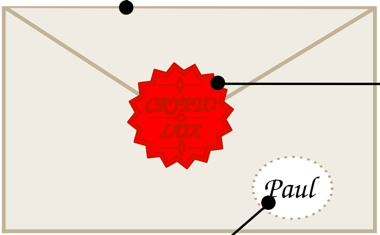


Cryptography? (2/2)



Cryptography? (2/2)

Envelope: **Confidentiality**
(nobody can read it)



Seal: **Integrity**
(nobody can modify it)

Signature: **Authentication**
(it was written by the right person)

Modern Cryptography

	Before
Data encrypted	Letters/Digits
Method	By hand/ machine
Cryptographers	Linguists inventors
Example	

Modern Cryptography

	Before	Now
Data encrypted	Letters/Digits	0,1
Method	By hand/ machine	Computer program
Cryptographers	Linguists inventors	Mathematicians Computer scientists
Example		(code below)

```
void sparx_encrypt(uint16_t * x, uint16_t k[][2*R_S]) {
    unsigned int s, r, b;
    for (s=0 ; s<N_S ; s++) {
        for (b=0 ; b<N_B ; b++)
            for (r=0 ; r<R_S ; r++) {
                x[2*b  ] ^= k[N_B*s + b][2*r    ];
                x[2*b+1] ^= k[N_B*s + b][2*r + 1];
                A(x[2*b], x[2*b+1]);
            }
        L(x);
    }
    for (b=0 ; b<N_B ; b++) {
        x[2*b  ] ^= k[N_B*N_S][2*b  ];
        x[2*b+1] ^= k[N_B*N_S][2*b+1];
    }
}
```

Symmetric Cryptography

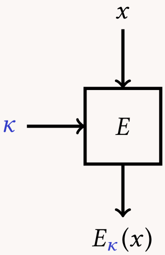
There are many **symmetric** algorithms! Hash functions, MACs...

Symmetric Cryptography

There are many **symmetric** algorithms! Hash functions, MACs...

Definition (Block Cipher)

- Input: n-bit block x
- Parameter: k-bit key κ
- Output: n-bit block $E_\kappa(x)$
- Symmetry: E and E^{-1} use the same κ

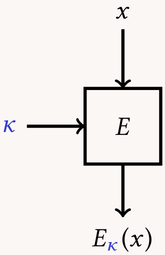


Symmetric Cryptography

There are many **symmetric** algorithms! Hash functions, MACs...

Definition (Block Cipher)

- Input: n-bit block x
- Parameter: k-bit key κ
- Output: n-bit block $E_\kappa(x)$
- Symmetry: E and E^{-1} use the same κ



Properties needed:

Diffusion

Confusion

No cryptanalysis!

Symmetric cryptography is the topic of this thesis.

Symmetric cryptography is the topic of this thesis.

What did I work on?

Lightweight Cryptography

- *Collision spectrum, entropy loss, T-sponges, and cryptanalysis of GLUON-64* (FSE'14) Khovratovich, Perrin; [Perrin and Khovratovich, 2015]
- *Differential analysis and meet-in-the-middle attack against round-reduced TWINE* (FSE'15) Biryukov, Derbez, Perrin ; [Biryukov et al., 2015]
- *Meet-in-the-middle attacks and structural analysis of round-reduced PRINCE* (FSE'15) Derbez, Perrin ; [Derbez and Perrin, 2015]
- *Design strategies for ARX with provable bounds: Sparx and LAX* (ASIACRYPT'16) Dinu, Perrin, Udovenko, Velichkov, Großschädl, Biryukov ; [Dinu et al., 2016]
- On Lightweight Symmetric Cryptography (SoK, Long Paper) (under submission) Biryukov, Perrin; see also `cryptolux.org`

S-Box Reverse-Engineering (1/3)

Actual Results on S-Boxes

- *On reverse-engineering S-boxes with hidden design criteria or structure* (CRYPTO'15) Biryukov, Perrin ; [Biryukov and Perrin, 2015]
- *Reverse-engineering the S-box of Streebog, Kuznyechik and STRIBOBr1* (EUROCRYPT'16) Biryukov, Perrin, Udovenko ; [Biryukov et al., 2016b]
- *Exponential S-boxes: a link between the S-boxes of BelT and Kuznyechik/Streebog* (ToSC'16), Perrin, Udovenko; [Perrin and Udovenko, 2017]

S-Box Reverse-engineering (2/3)

Structural Attacks

- *Cryptanalysis of Feistel networks with secret round functions* (SAC'15) Biryukov, Leurent, Perrin ; [Biryukov et al., 2016a]
- *Algebraic insights into the secret Feistel network* (FSE'16) Perrin, Udovenko ; [Perrin and Udovenko, 2016]
- *Multiset-algebraic cryptanalysis of reduced Kuznyechik, Khazad, and secret SPNs* (ToSC'16), Biryukov, Khovratovich, Perrin; [Biryukov et al., 2017]

S-Box Reverse-engineering (3/3)

Big APN Problem

- *Cryptanalysis of a theorem: Decomposing the only known solution to the big APN problem* (CRYPTO'16) Perrin, Udovenko, Biryukov; [Perrin et al., 2016]
- *A generalisation of Dillon's APN permutation with the best known differential and nonlinear properties for all fields of size* 2^{4k+2} (IEEE Transactions on Information Theory'17) Canteaut, Duval, Perrin; [Canteaut et al., 2017]

Purposefully Hard Cryptography

- A Generic Framework and Examples of Symmetrically and Asymmetrically Hard Functions (under submission) Biryukov, Perrin ;
- Katchup and Katchup-H: Proofs of Work with Different Classes of Users (under submission, a patent was filed) Biryukov, Perrin ;

Outline

Plan of this Section

S-Box?

An S-Box is a small non-linear function mapping m bits to n usually specified via its look-up table.

S-Box?

An S-Box is a small non-linear function mapping m bits to n usually specified via its look-up table.

- Typically, $n = m, n \in \{4, 8\}$
- Used by many block ciphers/hash functions/stream ciphers.
- Necessary for the wide trail strategy.

Example

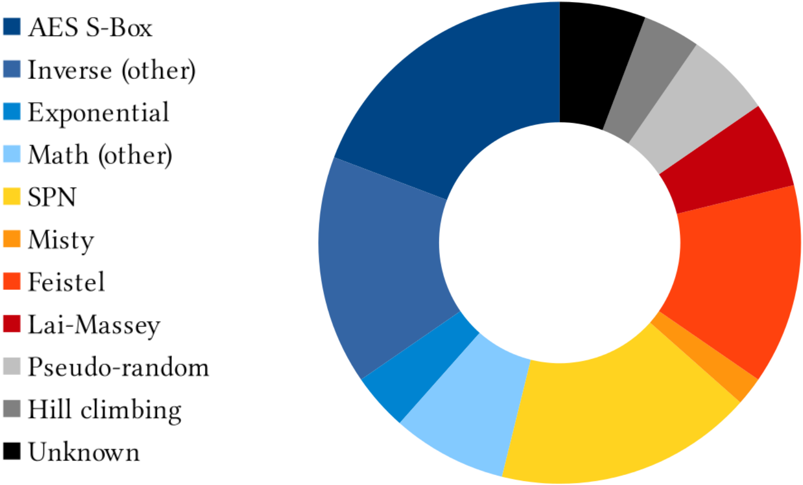
π' = (252, 238, 221, 17, 207, 110, 49, 22, 251, 196, 250, 218, 35, 197, 4, 77, 233, 119, 240, 219, 147, 46, 153, 186, 23, 54, 241. 187, 20, 205, 95, 193, 249, 24, 101, 90, 226, 92, 239, 33, 129, 28, 60, 66, 139, 1, 142, 79, 5, 132, 2, 174, 227, 106, 143, 160, 6, 11, 237, 152, 127, 212, 211, 31, 235, 52, 44, 81, 234, 200, 72, 171, 242, 42, 104, 162, 253, 58, 206, 204, 181, 112, 14, 86, 8, 12, 118, 18, 191, 114, 19, 71, 156, 183, 93, 135, 21, 161, 150, 41, 16, 123, 154, 199, 243, 145, 120, 111, 157, 158, 178, 177, 50, 117, 25, 61, 255, 53, 138, 126, 109, 84, 198, 128, 195, 189, 13, 87, 223, 245, 36, 169, 62, 168, 67, 201, 215, 121, 214, 246, 124, 34, 185, 3, 224, 15, 236, 222, 122, 148, 176, 188, 220, 232, 40, 80, 78, 51, 10, 74, 167, 151, 96, 115, 30, 0, 98, 68, 26, 184, 56, 130, 100, 159, 38, 65, 173, 69, 70, 146, 39, 94, 85, 47, 140, 163, 165, 125, 105, 213, 149, 59, 7, 88, 179, 64, 134, 172, 29, 247, 48, 55, 107, 228, 136, 217, 231, 137, 225, 27, 131, 73, 76, 63, 248, 254, 141, 83, 170, 144, 202, 216, 133, 97, 32, 113, 103, 164, 45, 43, 9, 91, 203, 155, 37, 208, 190, 229, 108, 82, 89, 166, 116, 210, 230, 244, 180, 192, 209, 102, 175, 194, 57, 75, 99, 182).

Screen capture from [GOST, 2015].

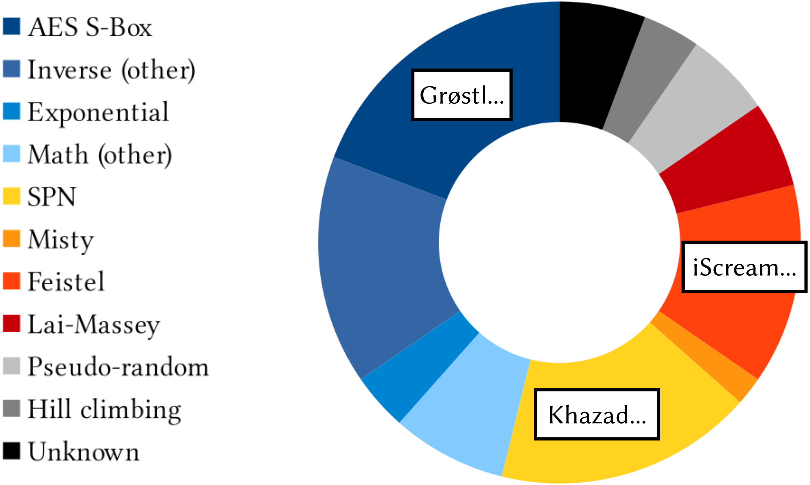
S-Box Design

- AES S-Box
- Inverse (other)
- Exponential
- Math (other)
- SPN
- Misty
- Feistel
- Lai-Massey
- Pseudo-random
- Hill climbing
- Unknown

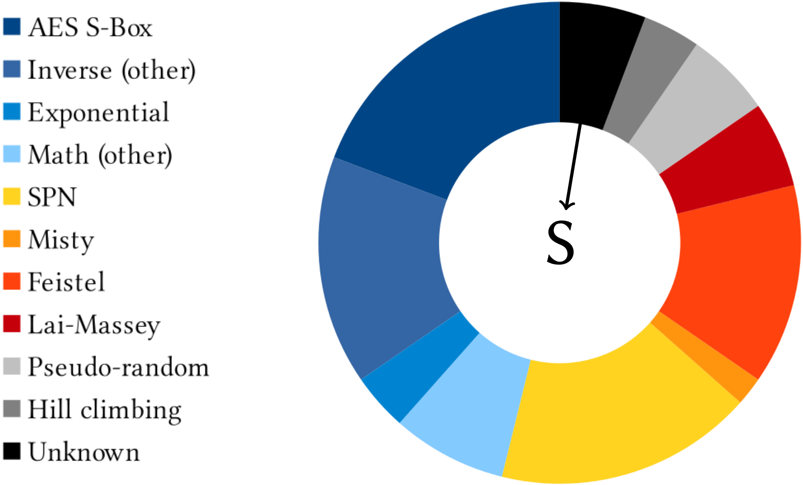
S-Box Design


AES S-Box
Inverse (other)
Exponential
Math (other)
SPN
Misty
Feistel
Lai-Massey
Pseudo-random
Hill climbing
Unknown

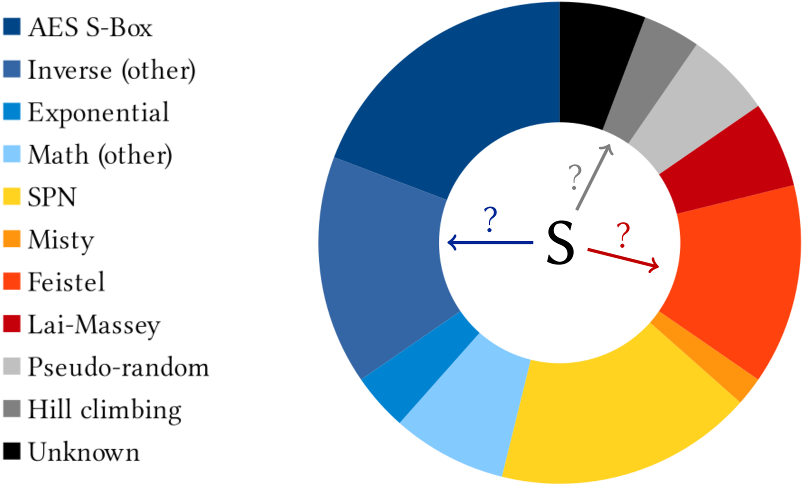
S-Box Design


AES S-Box
Inverse (other)
Exponential
Math (other)
SPN
Misty
Feistel
Lai-Massey
Pseudo-random
Hill climbing
Unknown
Grøstl...
iScream...
Khazad...

S-Box Reverse-Engineering



S-Box Reverse-Engineering



Motivation

A malicious designer can easily hide a structure in an S-Box.

Motivation

A malicious designer can easily hide a structure in an S-Box.

To keep an advantage in implementation (WB crypto)...

Motivation

A malicious designer can easily hide a structure in an S-Box.

To keep an advantage in implementation (WB crypto)...
... or an advantage in cryptanalysis (backdoor).

The Two Tables

Let $S : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ be an S-Box.

The Two Tables

Let $S : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ be an S-Box.

Definition (DDT)

The *Difference Distribution Table* of S is a matrix of size $2^n \times 2^n$ such that

$$\mathrm{DDT}[a, b] \ = \ \#\{x \in \mathbb{F}_2^n \mid S(x \oplus a) \oplus S(x) = b\}.$$

The Two Tables

Let $S : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ be an S-Box.

Definition (DDT)

The *Difference Distribution Table* of S is a matrix of size $2^n \times 2^n$ such that

$$\mathrm{DDT}[a,b] = \#\{x \in \mathbb{F}_2^n \mid S(x \oplus a) \oplus S(x) = b\}.$$

Definition (LAT)

The *Linear Approximations Table* of S is a matrix of size $2^n \times 2^n$ such that

$$\mathrm{LAT}[a,b] = \#\{x \in \mathbb{F}_2^n \mid x \cdot a = S(x) \cdot b\} - 2^{n-1}.$$

Example

$$S = [4, 2, 1, 6, 0, 5, 7, 3]$$

The DDT of S.

$$\begin{bmatrix} 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 2 & 2 & 2 \\ 0 & 0 & 0 & 0 & 2 & 2 & 2 & 2 \\ 0 & 0 & 4 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 2 & 2 & 2 \\ 0 & 4 & 4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 2 & 2 & 2 \end{bmatrix}$$

The LAT of S.

$$\begin{bmatrix} 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 2 & 0 & 0 & 2 & -2 \\ 0 & 2 & 2 & 0 & 0 & 2 & -2 & 0 \\ 0 & 2 & 0 & 2 & 0 & -2 & 0 & 2 \\ 0 & 2 & 0 & -2 & 0 & -2 & 0 & -2 \\ 0 & -2 & 2 & 0 & 0 & -2 & -2 & 0 \\ 0 & 0 & -2 & 2 & 0 & 0 & -2 & -2 \\ 0 & 0 & 0 & 0 & -4 & 0 & 0 & 0 \end{bmatrix}$$

Coefficient Distribution in the DDT

If an n-bit S-Box is bijective, then its DDT coefficients behave like independent and identically distributed random variables following a Poisson distribution:

$$\Pr\left[\mathrm{DDT}[a,b] = 2z\right] = \frac{e^{-1/2}}{2^z z} .$$

Coefficient Distribution in the DDT

If an n-bit S-Box is bijective, then its DDT coefficients behave like independent and identically distributed random variables following a Poisson distribution:

$$\Pr\left[\mathrm{DDT}[a,b] = 2z\right] = \frac{e^{-1/2}}{2^z z} .$$

- Always even, ≥ 0
- Typically between 0 and 16.
- Lower is better.

Coefficient Distribution in the LAT

If an n-bit S-Box is bijective, then its LAT coefficients behave like independent and identically distributed random variables following this distribution:

$$\Pr\left[\mathrm{LAT}[a,b] = 2z\right] = \frac{\binom{2^{n-1}}{2^{n-2+z}}}{\binom{2^n}{2^{n-1}}}.$$

Coefficient Distribution in the LAT

If an n-bit S-Box is bijective, then its LAT coefficients behave like independent and identically distributed random variables following this distribution:

$$\Pr\left[\mathrm{LAT}[a,b] = 2z\right] = \frac{\binom{2^{n-1}}{2^{n-2+z}}}{\binom{2^n}{2^{n-1}}} .$$

- Always even, signed.
- Typically between -40 and 40.
- Lower absolute value is better.

Looking Only at the Maximum

δ	$\log_2 (\Pr[\max(\mathcal{D}) \leq \delta])$
14	-0.006
12	-0.094
10	-1.329
8	-16.148
6	-164.466
4	-1359.530

DDT

ℓ	$\log_2 (\Pr[\max(\mathcal{L}) \leq \ell])$
38	-0.084
36	-0.302
34	-1.008
32	-3.160
30	-9.288
28	-25.623
26	-66.415
24	-161.900
22	-371.609

LAT

Probability that the maximum coefficient in the DDT/LAT of an 8-bit permutation is at most equal to a certain threshold.

Looking Only at the Maximum

δ	$\log_2 (\Pr[\max(\mathcal{D}) \leq \delta])$
14	-0.006
12	-0.094
10	-1.329
8	-16.148
6	-164.466
4	-1359.530

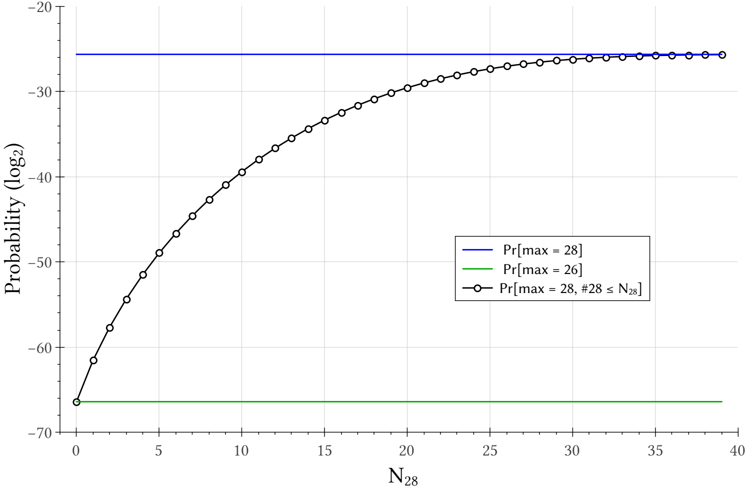
DDT

ℓ	$\log_2 (\Pr[\max(\mathcal{L}) \leq \ell])$
38	-0.084
36	-0.302
34	-1.008
32	-3.160
30	-9.288
28	-25.623
26	-66.415
24	-161.900
22	-371.609

LAT

Probability that the maximum coefficient in the DDT/LAT of an 8-bit permutation is at most equal to a certain threshold.

Taking Number of Maximum Values into Account

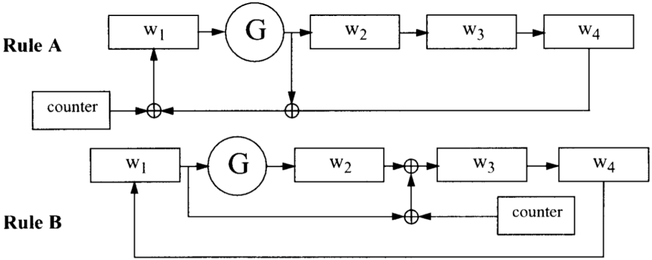


Application of this Analysis?

We applied this method on the S-Box of Skipjack.

What is Skipjack? (1/2)

Type	Block cipher
Bloc	64 bits
Key	80 bits
Authors	NSA
Publication	1998



What is Skipjack? (2/2)

- Skipjack was supposed to be secret...
- ... but eventually published in 1998 [NIST, 1998],

What is Skipjack? (2/2)

- Skipjack was supposed to be secret...
- ... but eventually published in 1998 [NIST, 1998],
- It uses an 8×8 S-Box (F) specified only by its LUT,

What is Skipjack? (2/2)

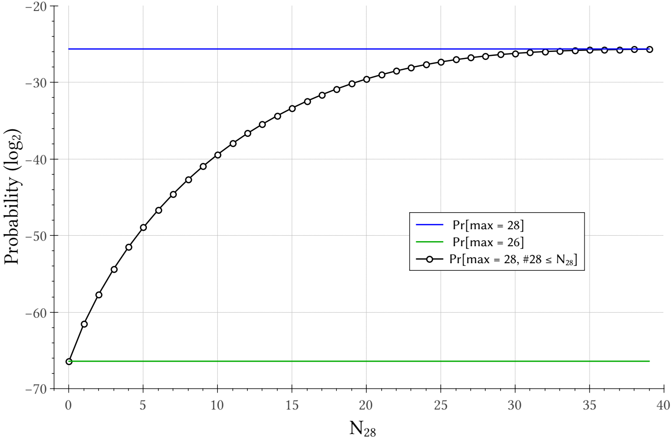
- Skipjack was supposed to be secret...
- ... but eventually published in 1998 [NIST, 1998],
- It uses an 8×8 S-Box (F) specified only by its LUT,
- Skipjack was to be used by the *Clipper Chip*.

Reverse-Engineering F

For Skipjack's F, $\max(\mathrm{LAT}) = 28$ and $\#28 = 3$.

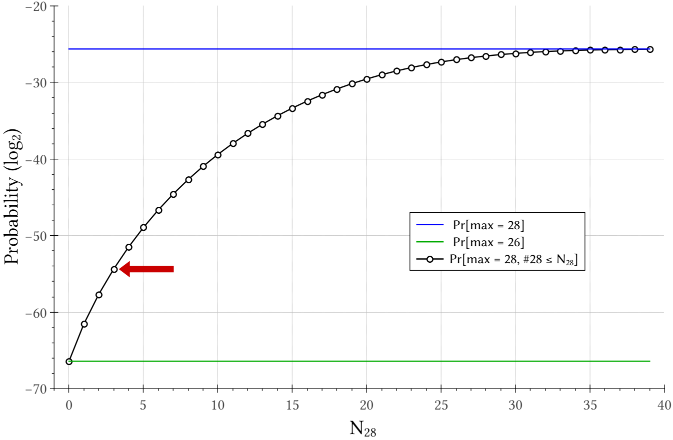
Reverse-Engineering F

For Skipjack's F, max(LAT) = 28 and #28 = 3.



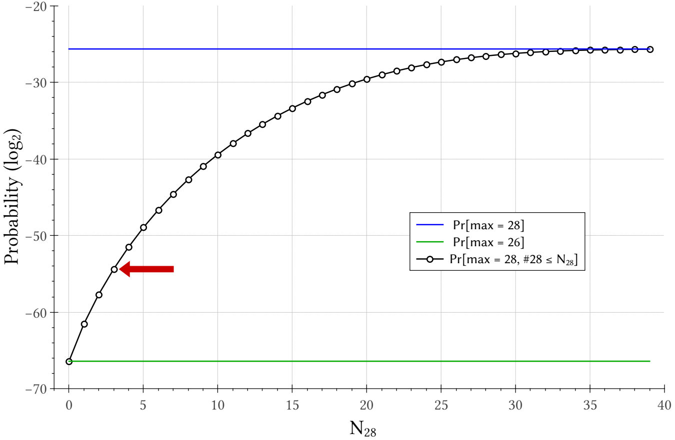
Reverse-Engineering F

For Skipjack's F, max(LAT) = 28 and #28 = 3.



Reverse-Engineering F

For Skipjack's F, $\max(\mathrm{LAT}) = 28$ and $\#28 = 3$.



$$\Pr\left[\max(\mathrm{LAT}) = 28 \text{ and } \#28 \leq 3\right] \approx \mathbf{2^{-55}}$$

What Can We Deduce?

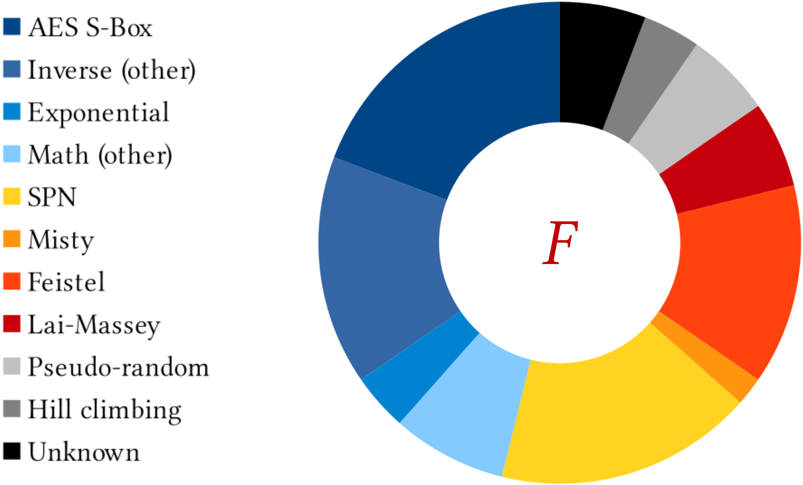
- F has not been picked uniformly at random.
- F has not been picked among a feasibly large set of random S-Boxes.
- Its linear properties were optimized (though poorly).

What Can We Deduce?

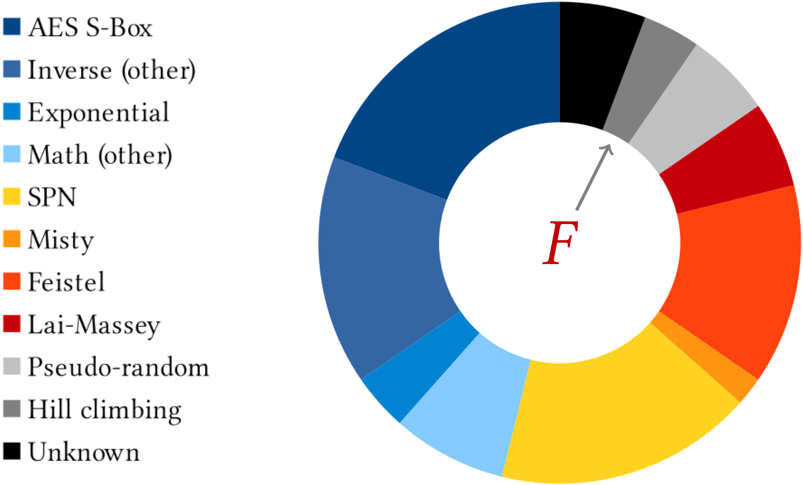
- F has not been picked uniformly at random.
- F has not been picked among a feasibly large set of random S-Boxes.
- Its linear properties were optimized (though poorly).

The S-Box of Skipjack was built using a dedicated algorithm.

Conclusion on Skipjack



Conclusion on Skipjack



Distinguisher vs. Decomposition

We have figured out that F is not random...

Distinguisher vs. Decomposition

We have figured out that F is not random...

But what can we do to find actual structures?

Structural Attacks

Attacks against structures regardless of their details. Examples:

- Integral attacks against SPNs,
- Yoyo game against Feistel Networks,
- Looking at the Pollock representations of the DDT/LAT,

Distinguisher vs. Decomposition

We have figured out that F is not random...

But what can we do to find actual structures?

Structural Attacks

Attacks against structures regardless of their details. Examples:

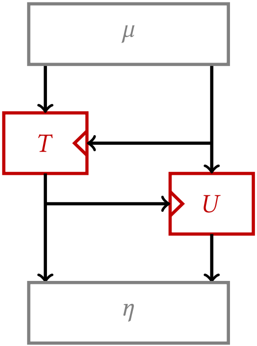
- Integral attacks against SPNs,
- Yoyo game against Feistel Networks,
- Looking at the Pollock representations of the DDT/LAT,
- TU-Decomposition.

TU-Decomposition in a Nutshell

1. Identify linear patterns in zeroes of LAT;

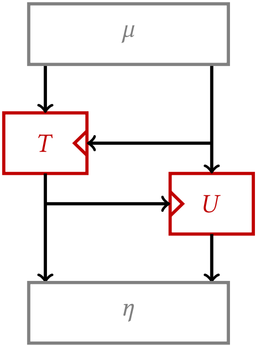
TU-Decomposition in a Nutshell

1. Identify linear patterns in zeroes of LAT;
2. Deduce linear layers μ, η such that π is decomposed as in right picture;



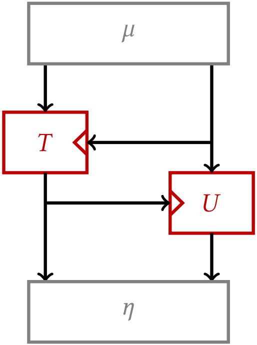
TU-Decomposition in a Nutshell

1. Identify linear patterns in zeroes of LAT;
2. Deduce linear layers μ, η such that π is decomposed as in right picture;
3. Decompose U, T;



TU-Decomposition in a Nutshell

1. Identify linear patterns in zeroes of LAT;
2. Deduce linear layers μ, η such that π is decomposed as in right picture;
3. Decompose U, T;
4. Put it all together.



Kuznyechik/Stribog

Stribog

Type	Hash function
Publication	[GOST, 2012]

Kuznyechik

Type	Block cipher
Publication	[GOST, 2015]

Kuznyechik/Stribog

Stribog

Type Hash function

Publication [GOST, 2012]

Kuznyechik

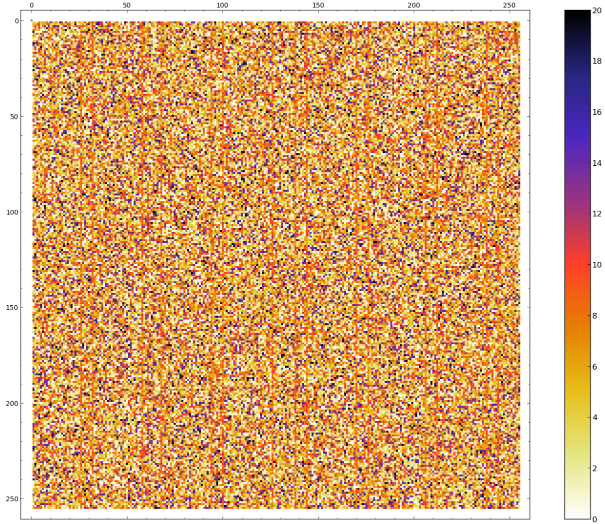
Type Block cipher

Publication [GOST, 2015]

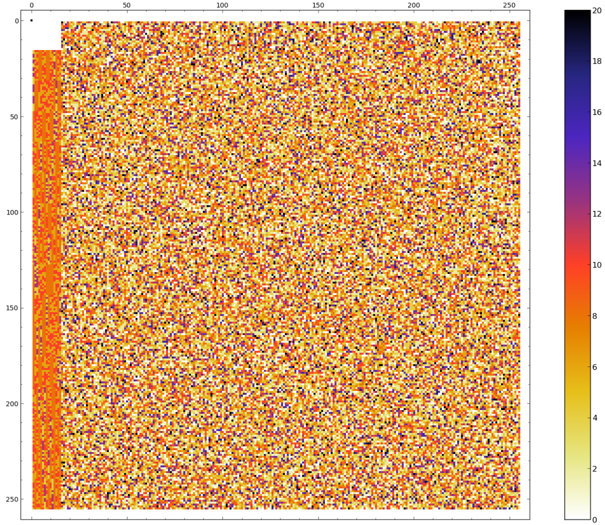
Common ground

- Both are standard symmetric primitives in Russia.
- Both were designed by the FSB (TC26).
- Both use the same 8×8 S-Box, π.

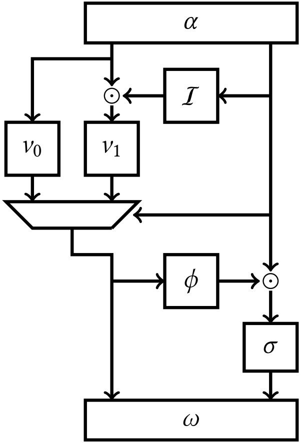
The LAT of π



The LAT of $\eta \circ \pi \circ \mu$



Final Decomposition Number 1



- $\odot$ Multiplication in $\mathbb{F}_{2^4}$
- α Linear permutation
- $\mathcal{I}$ Inversion in $\mathbb{F}_{2^4}$
- ν_0, ν_1, σ 4×4 permutations
- ϕ 4×4 function
- ω Linear permutation

Conclusion for Kuznyechik/Stribog?

The Russian S-Box was built like a strange Feistel...

Conclusion for Kuznyechik/Stribog?

The Russian S-Box was built like a strange Feistel...

... or was it?

Conclusion for Kuznyechik/Stribog?

The Russian S-Box was built like a strange Feistel...

... or was it?

Belarussian inspiration

- The last standard of Belarus [Bel. St. Univ., 2011] uses an 8-bit S-box,
- somewhat similar to π...

Conclusion for Kuznyechik/Stribog?

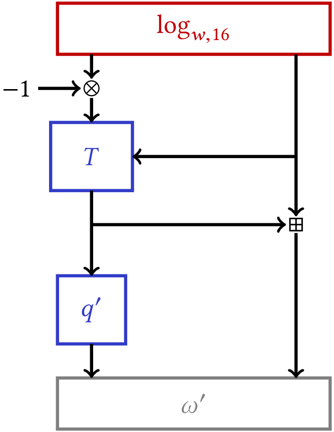
The Russian S-Box was built like a strange Feistel...

... or was it?

Belarussian inspiration

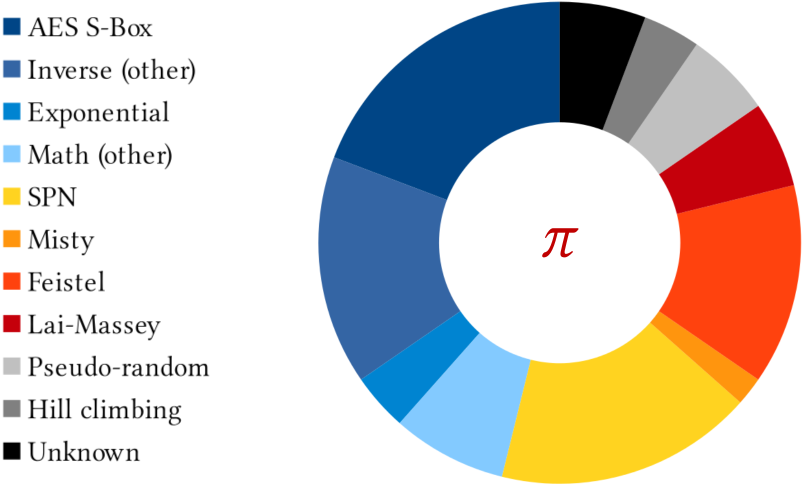
- The last standard of Belarus [Bel. St. Univ., 2011] uses an 8-bit S-box,
- somewhat similar to π...
- ... based on a finite field exponential!

Final Decomposition Number 2 (!)

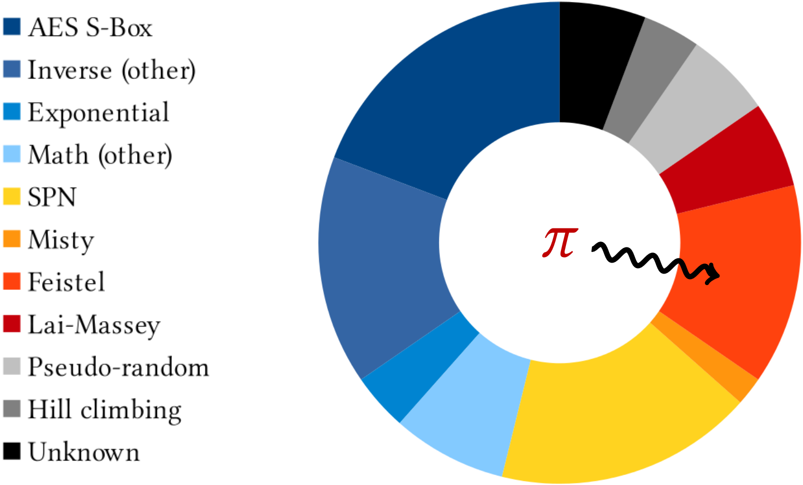


	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
T_0	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
T_1	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
T_2	0	1	2	3	4	5	6	7	8	9	a	b	c	d	f	e
T_3	0	1	2	3	4	5	6	7	8	9	a	b	c	f	d	e
T_4	0	1	2	3	4	5	6	7	8	9	a	b	f	c	d	e
T_5	0	1	2	3	4	5	6	7	8	9	a	f	b	c	d	e
T_6	0	1	2	3	4	5	6	7	8	9	f	a	b	c	d	e
T_7	0	1	2	3	4	5	6	7	8	f	9	a	b	c	d	e
T_8	0	1	2	3	4	5	6	7	f	8	9	a	b	c	d	e
T_9	0	1	2	3	4	5	6	f	7	8	9	a	b	c	d	e
T_a	0	1	2	3	4	5	f	6	7	8	9	a	b	c	d	e
T_b	0	1	2	3	4	f	5	6	7	8	9	a	b	c	d	e
T_c	0	1	2	3	f	4	5	6	7	8	9	a	b	c	d	e
T_d	0	1	2	f	3	4	5	6	7	8	9	a	b	c	d	e
T_e	0	1	f	2	3	4	5	6	7	8	9	a	b	c	d	e
T_f	0	f	1	2	3	4	5	6	7	8	9	a	b	c	d	e

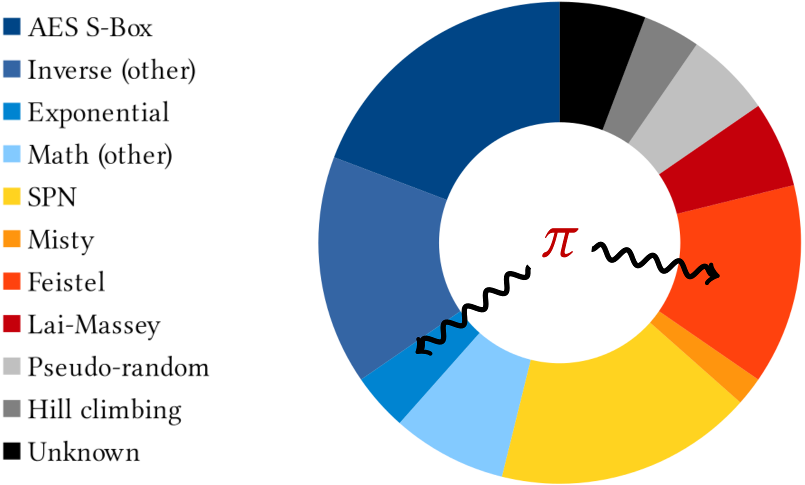
Conclusion on Kuznyechik/Stribog



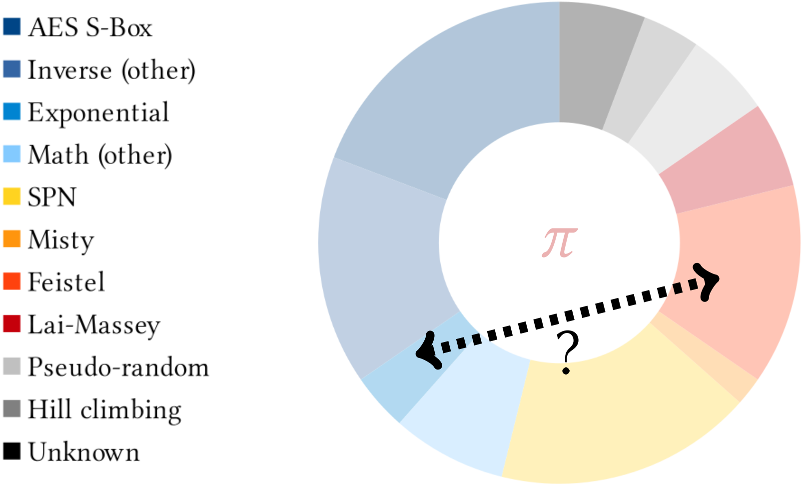
Conclusion on Kuznyechik/Stribog



Conclusion on Kuznyechik/Stribog



Conclusion on Kuznyechik/Stribog



Outline

Plan of this Section

What Things?

Everything is being connected to the internet.

What Things?

Everything

What Things?

Everything

What Things?

Everything

Security

"In IoT, the S is for Security."

- Internet-enabled devices have security flaws.
- Security is an afterthought (at best).
- Security has a cost in terms of engineering...
- ... and computationnal resources!

Lightweight Cryptography

Lightweight cryptography uses little resources.

Lightweight Cryptography from the Industry

Stream ciphers, unless †(BC) or ‡(MAC)

- A5/1
- A5/2
- CMEA †
- ORYX
- A5-GMR-1
- A5-GMR-2
- DSC
- SecureMem.
- CryptoMem.
- Hitag2
- Megamos
- Keeloq †
- DST40 †
- iClass
- Crypto-1
- CSS
- Cryptomeria †
- CSA-BC †
- CSA-SC
- PC-1
- SecurID ‡
- E0
- RC4

Lightweight Cryptography from the Industry

Stream ciphers, unless †(BC) or ‡(MAC)

- A5/1
- A5/2
- CMEA †
- ORYX
- A5-GMR-1
- A5-GMR-2
- DSC
- SecureMem.

- CryptoMem.
- Hitag2
- Megamos
- Keeloq †
- DST40 †
- iClass
- Crypto-1
- CSS

- Cryptomeria †
- CSA-BC †
- CSA-SC
- PC-1
- SecurID ‡
- E0
- RC4

They're all dead (attacks in less than 2^{64}).

Lightweight Block Ciphers from Academia

- 3-Way
- RC5
- Misty1
- XTEA
- AES
- Khazad
- Noekeon
- Iceberg
- mCrypton
- HIGHT
- SEA
- CLEFIA
- DESLX
- PRESENT
- MIBS
- KATAN
- GOST rev.
- PRINTCipher
- EPCBC
- KLEIN
- LBlock
- LED
- Piccolo
- PICARO
- PRINCE
- ITUbee
- TWINE
- Zorro
- Chaskey
- PRIDE
- Joltik
- LEA
- iScream
- LBlock-s
- Scream
- Lilliput
- RECTANGLE
- Fantomas
- Robin
- Midori
- SIMECK
- RoadRunneR
- FLY
- Mantis
- SKINNY
- SPARX
- Mysterion
- Qarma

48 distinct block ciphers!

Common Trade-Offs in LWC

- Small internal state size.

Common Trade-Offs in LWC

- Small internal state size.
- Small key.

Common Trade-Offs in LWC

- Small internal state size.
- Small key.
- Simple key schedule.

Common Trade-Offs in LWC

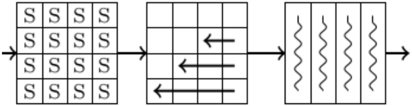
- Small internal state size.
- Small key.
- Simple key schedule.
- No table look-ups (instead, ARX or bit-sliced S-Box).

How did we design SPARX?

Block Cipher Design (1/2)

Requirement	S-Box-based	ARX-based
Confusion	S	$\boxplus$
Diffusion	L	$\boxplus, \lll, \oplus$

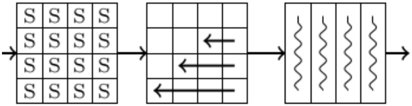
Block Cipher Design (2/2)



$$P_{\text{diff}} \leq \left(\frac{\Delta_S}{2^b}\right)^{\#\text{ active S-Boxes}}$$

Design of an S-Box based SPN (wide trail strategy)

Block Cipher Design (2/2)



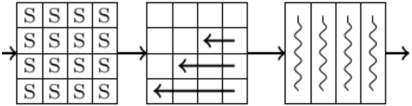
$$P_{\text{diff}} \leq \left(\frac{\Delta_S}{2^b}\right)^{\#\text{ active S-Boxes}}$$

Design of an S-Box based SPN (wide trail strategy)

Design of an ARX-cipher (allegory)

source: Wiki Commons

Block Cipher Design (2/2)



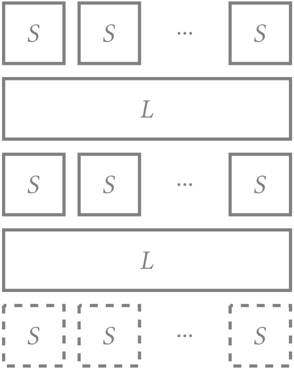
$$P_{\text{diff}} \leq \left(\frac{\Delta_S}{2^b}\right)^{\#\text{ active S-Boxes}}$$

Design of an S-Box based SPN (wide trail strategy)

Design of an ARX-cipher (allegory)
source: Wiki Commons

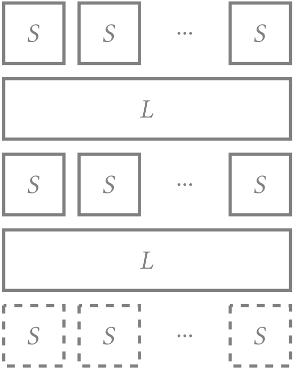
Can we use ARX *and* have provable bounds?

Trail Based Argument



Bouding 2-round differential probability.

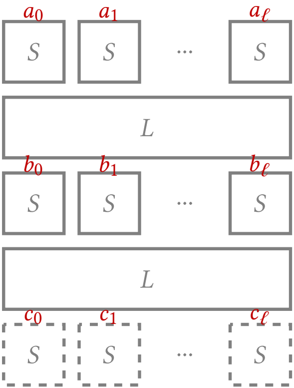
Trail Based Argument



Bouding 2-round differential probability.

1. Consider all trails $A \rightsquigarrow B \rightsquigarrow C$, where $A = (a_0, ..., a_\ell)$, etc.

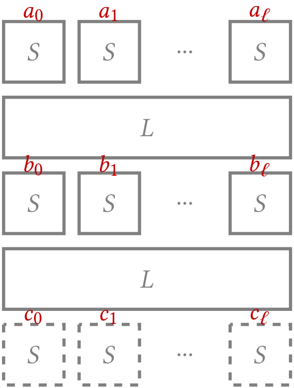
Trail Based Argument



Bouding 2-round differential probability.

1. Consider all trails $A \rightsquigarrow B \rightsquigarrow C$, where $A = (a_0, \ldots, a_\ell)$, etc.

Trail Based Argument

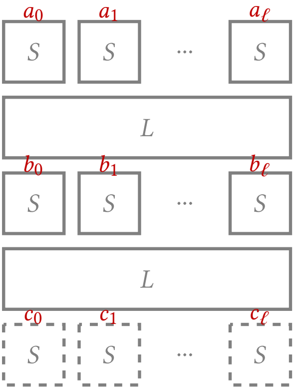


Bouding 2-round differential probability.

1. Consider all trails $A \rightsquigarrow B \rightsquigarrow C$, where $A = (a_0, \dots, a_\ell)$, etc.

2. Markov assumption:
$\Pr[A \rightsquigarrow B \rightsquigarrow C] = \Pr[A \rightsquigarrow B] \times \Pr[B \rightsquigarrow C]$

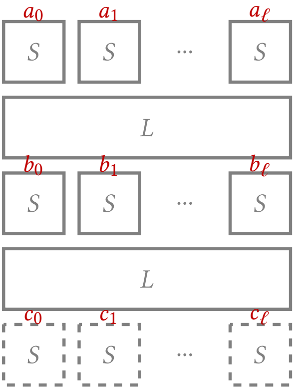
Trail Based Argument



Bouding 2-round differential probability.

1. Consider all trails $A \rightsquigarrow B \rightsquigarrow C$, where $A = (a_0, \ldots, a_\ell)$, etc.
2. Markov assumption: $\Pr[A \rightsquigarrow B \rightsquigarrow C] = \Pr[A \rightsquigarrow B] \times \Pr[B \rightsquigarrow C]$
3. Show that, for all A, B, C:
   - if $\Pr[A \rightsquigarrow B]$ is high,
   - then $\Pr[B \rightsquigarrow C]$ is low.

Trail Based Argument



Bouding 2-round differential probability.

1. Consider all trails $A \rightsquigarrow B \rightsquigarrow C$, where $A = (a_0, \ldots, a_\ell)$, etc.
2. Markov assumption:
   $\Pr[A \rightsquigarrow B \rightsquigarrow C] = \Pr[A \rightsquigarrow B] \times \Pr[B \rightsquigarrow C]$
3. Show that, for all A, B, C:
   - if $\Pr[A \rightsquigarrow B]$ is high,
   - then $\Pr[B \rightsquigarrow C]$ is low.
4. Conclude that $\Pr[A \rightsquigarrow B \rightsquigarrow C]$ can't be high.

Proving Point 3: *Wide* Trail Argument

Wide Trail Argument

- At the S-Box level, $\Pr[a_i \rightsquigarrow b_i] \leq p$.
- At the trail level, if $\#\{i, a_i \neq 0\}$ is *low* then $\#\{i, b_i \neq 0\}$ is *high* because their sum is $\geq B(L)$.

Conclusion: best trail over 2 rounds has probability at most

$$p^{B(L)} .$$

Proving Point 3: *Long* Trail Argument

Long Trail Argument

- At the S-Box level, use heuristic to show

$$\Pr\left[a_i \rightsquigarrow b_i\right] \leq p_1 \ ,$$

$$\Pr\left[a_i \rightsquigarrow b_i \rightsquigarrow c_i\right] \leq p_2 \ll {p_1}^2 \ \ldots$$

Proving Point 3: *Long* Trail Argument

Long Trail Argument

- At the S-Box level, use heuristic to show

$$\Pr\left[a_i \rightsquigarrow b_i\right] \le p_1 ,$$

$$\Pr\left[a_i \rightsquigarrow b_i \rightsquigarrow c_i\right] \le p_2 \ll {p_1}^2 \ldots$$

- At the trail level, decompose $A \rightsquigarrow B \rightsquigarrow C$ into independent trails at the S-Box level, e.g. $a_0 \rightsquigarrow b_1 \rightsquigarrow c_0$, $a_1 \rightsquigarrow b_0$, ...

Proving Point 3: *Long* Trail Argument

Long Trail Argument

- At the S-Box level, use heuristic to show

$$\Pr\left[a_i \rightsquigarrow b_i\right] \leq p_1 ,$$

$$\Pr\left[a_i \rightsquigarrow b_i \rightsquigarrow c_i\right] \leq p_2 \ll {p_1}^2 \ldots$$

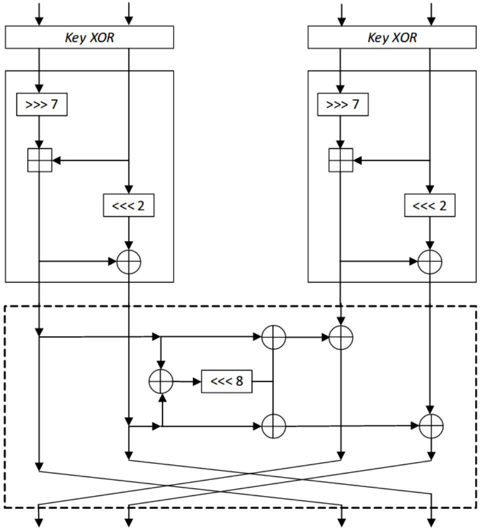
- At the trail level, decompose $A \rightsquigarrow B \rightsquigarrow C$ into independent trails at the S-Box level, e.g. $a_0 \rightsquigarrow b_1 \rightsquigarrow c_0$, $a_1 \rightsquigarrow b_0$, ...
- Bound probability using product of p_1, p_2, etc. depending on the lengths of the S-Box-level trails.

SPARX

1. Substitution-Permutation ARX.
2. Built using a wide-trail strategy...
3. ... thus, provably secure against differential/linear attacks!
4. Quite efficient on micro-controllers.

n/k	64/128	128/128	128/256
# Rounds/Step	3	4	4
# Steps	8	8	10
Best Attack (# rounds)	15/24	22/32	24/40

High Level View of SPARX-64/128



Impossible differential attack on reduced round SPARX-64/128
(AFRICACRYPT'2017)
Abdelkhalek, A., Tolba, M., and Youssef, A;
[Abdelkhalek et al., 2017]

Outline

Plan of this Section

Conclusion

1. We can recover the majority of known S-Box structures and derive new results about Skipjack and Kuznyechik.

Conclusion

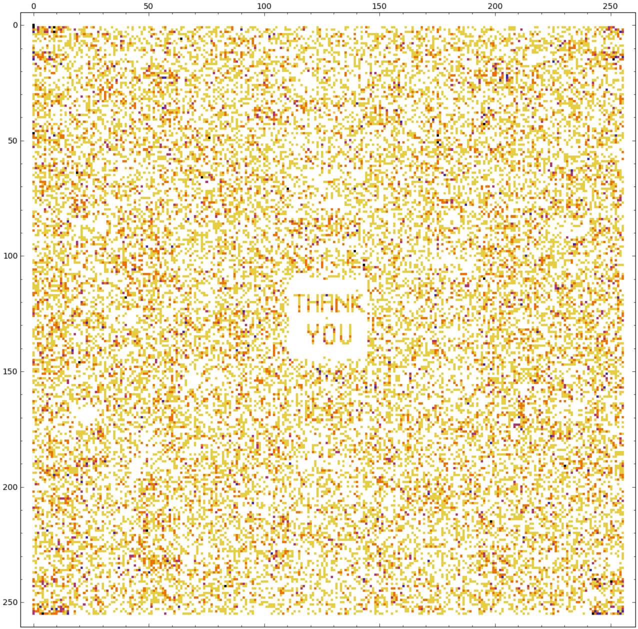
1. We can recover the majority of known S-Box structures and derive new results about Skipjack and Kuznyechik.

2. We can design an efficient ARX-based lightweight block ciphers with provable security against differential/linear attacks.

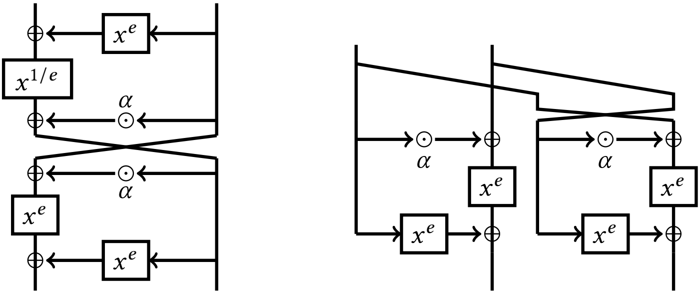
The Last S-Box

14	11	60	6d	e9	10	e3	2	b	90	d	17	c5	b0	9f	c5
d8	da	be	22	8	f3	4	a9	fe	f3	f5	fc	bc	30	be	26
bb	88	85	46	f4	2e	e	fd	76	fe	b0	11	4e	de	35	bb
30	4b	30	d6	dd	df	df	d4	90	7a	d8	8c	6a	89	30	39
e9	1	da	d2	85	87	d3	d4	ba	2b	d4	9f	9c	38	8c	55
d3	86	bb	db	ec	e0	46	48	bf	46	1b	1c	d7	d9	1b	e0
23	d4	d7	7f	16	3f	3	3	44	c3	59	10	2a	da	ed	e9
8e	d8	d1	db	cb	cb	c3	c7	38	22	34	3d	db	85	23	7c
24	d1	d8	2e	fc	44	8	38	c8	c7	39	4c	5f	56	2a	cf
d0	e9	d2	68	e4	e3	e9	13	e2	c	97	e4	60	29	d7	9b
d9	16	24	94	b3	e3	4c	4c	4f	39	e0	4b	bc	2c	d3	94
81	96	93	84	91	d0	2e	d6	d2	2b	78	ef	d6	9e	7b	72
ad	c4	68	92	7a	d2	5	2b	1e	d0	dc	b1	22	3f	c3	c3
88	b1	8d	b5	e3	4e	d7	81	3	15	17	25	4e	65	88	4e
e4	3b	81	81	fa	1	1d	4	22	0	6	1	27	68	27	2e
3b	83	c7	cc	25	9b	d8	d5	1c	1f	e5	59	7f	3f	3f	ef

Conclusion



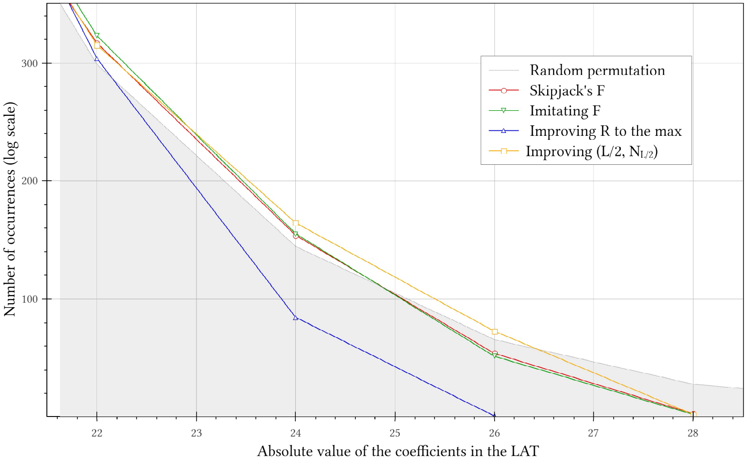
On the Butterfly Structure



(a) Open (bijective) butterfly H^e_α. (b) Closed (non-bijective) butterfly V^e_α.

Figure : The two types of butterfly structure with coefficient α and exponent e.

Details About Skipjack



High Level View of SPARX (algo)

Algorithm 7.1 SPARX encryption
Inputs plaintext $(x_0, \ldots, x_{w-1})$; key $(k_0, \ldots, k_{v-1})$
Output ciphertext $(y_0, \ldots, y_{w-1})$

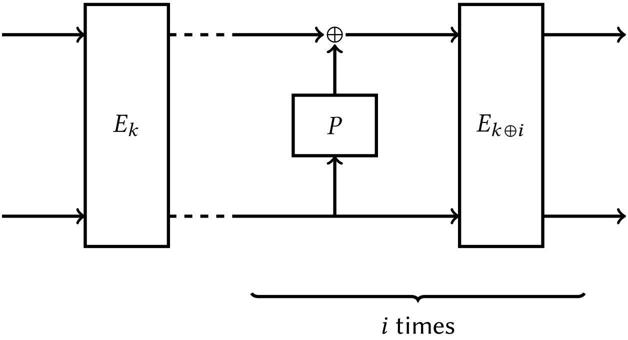
Let $y_i \leftarrow x_i$ for all $i \in [0, \ldots, w-1]$
for all $s \in [0, n_s - 1]$ **do**
  for all $i \in [0, w-1]$ **do**
    for all $r \in [0, r_a - 1]$ **do**
      $y_i \leftarrow y_i \oplus k_r$
      $y_i \leftarrow A(y_i)$
    end for
    $(k_0, \ldots, k_{v-1}) \leftarrow K_v\Big((k_0, \ldots, k_{v-1})\Big)$ ▷ Update key state
  end for
  $(y_0, \ldots, y_{w-1}) \leftarrow \lambda_w\Big((y_0, \ldots, y_{w-1})\Big)$ ▷ Linear mixing layer
end for
Let $y_i \leftarrow y_i \oplus k_i$ for all $i \in [0, \ldots, w-1]$ ▷ Final key addition
return $(y_0, \ldots, y_{w-1})$

Details About ULW vs. IoT Crypto

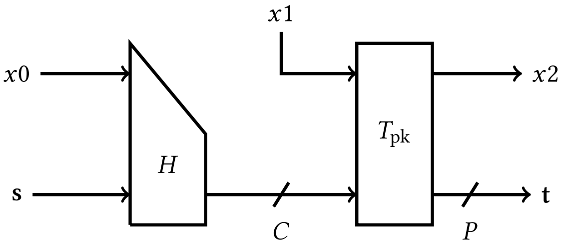
	Ultra-Lightweight	IoT
Block size	64 bits	$\geq$ 128 bits
Security level	$\geq$ 80 bits	$\geq$ 128 bits
Relevant attacks	low data complexity	Same as “regular” crypto
Intended platform	dedicated circuit	low-end CPUs
SCA resilience	important	important
Functionality	one per device	encryption, authentication...
Connection	to a central hub	to a global network

Table : A summary of the differences between ultra-lightweight and IoT cryptography.

Hard Block Cipher



Katchup-H



Fixing Justification of Attack 11.5.4 (1/2)

Lemma

Let $F : \mathbb{F}_2^n \to \mathbb{F}_2$ *be a Boolean function and let* $G : \mathbb{F}_2^n \to \mathbb{F}_2^n$ *be a permutation. Then:*

$$\deg(F \circ G) = n - 1 \quad \Longrightarrow \quad \deg(F) + \deg(G^{-1}) \geq n\ .$$

Fixing Justification of Attack 11.5.4 (2/2)

If $\deg(F \circ G) = n - 1$, then $\exists i \leq n$ such that $\bigoplus_{x \in C_i} (F \circ G)(x) = 1$.

Fixing Justification of Attack 11.5.4 (2/2)

If $\deg(F \circ G) = n - 1$, then $\exists i \leq n$ such that $\bigoplus_{x \in C_i} (F \circ G)(x) = 1$.

Let $I_i : \mathbb{F}_2^n \to \mathbb{F}_2$ be such that $I_i(x) = 1 \Leftrightarrow x \in C_i$:

$$\bigoplus_{x \in C_i} (F \circ G)(x) = \bigoplus_{x \in \mathbb{F}_2^n} F\big(G(x)\big) \times I_i(x) \, ,$$

Fixing Justification of Attack 11.5.4 (2/2)

If $\deg(F \circ G) = n - 1$, then $\exists i \leq n$ such that $\bigoplus_{x \in C_i} (F \circ G)(x) = 1$.

Let $I_i : \mathbb{F}_2^n \to \mathbb{F}_2$ be such that $I_i(x) = 1 \Leftrightarrow x \in C_i$:

$$\bigoplus_{x \in C_i} (F \circ G)(x) = \bigoplus_{x \in \mathbb{F}_2^n} F\Big(G(x)\Big) \times I_i(x) \ ,$$

and let $y = G(x)$. Then:

$$\bigoplus_{x \in C_i} (F \circ G)(x) = \bigoplus_{y \in \mathbb{F}_2^n} F(y) \times I_i\Big(G^{-1}(y)\Big) \ .$$

Fixing Justification of Attack 11.5.4 (2/2)

If $\deg(F \circ G) = n - 1$, then $\exists i \leq n$ such that $\bigoplus_{x \in C_i} (F \circ G)(x) = 1$.

Let $I_i : \mathbb{F}_2^n \to \mathbb{F}_2$ be such that $I_i(x) = 1 \Leftrightarrow x \in C_i$:

$$\bigoplus_{x \in C_i} (F \circ G)(x) \; = \; \bigoplus_{x \in \mathbb{F}_2^n} F\big(G(x)\big) \times I_i(x) \,,$$

and let $y = G(x)$. Then:

$$\bigoplus_{x \in C_i} (F \circ G)(x) \; = \; \bigoplus_{y \in \mathbb{F}_2^n} F(y) \times I_i\big(G^{-1}(y)\big) \,.$$

This sum is equal to 1 if and only if $x \mapsto F(x) \times I_i\big(G^{-1}(x)\big)$ has degree n.

Fixing Justification of Attack 11.5.4 (2/2)

If $\deg(F \circ G) = n - 1$, then $\exists i \leq n$ such that $\bigoplus_{x \in C_i} (F \circ G)(x) = 1$.

Let $I_i : \mathbb{F}_2^n \to \mathbb{F}_2$ be such that $I_i(x) = 1 \Leftrightarrow x \in C_i$:

$$\bigoplus_{x \in C_i} (F \circ G)(x) = \bigoplus_{x \in \mathbb{F}_2^n} F\big(G(x)\big) \times I_i(x) ,$$

and let $y = G(x)$. Then:

$$\bigoplus_{x \in C_i} (F \circ G)(x) = \bigoplus_{y \in \mathbb{F}_2^n} F(y) \times I_i\big(G^{-1}(y)\big) .$$

This sum is equal to 1 if and only if $x \mapsto F(x) \times I_i\big(G^{-1}(x)\big)$ has degree n.
I_i is affine ($I_i(x) = 1 + x_i$).

Fixing Justification of Attack 11.5.4 (2/2)

If $\deg(F \circ G) = n - 1$, then $\exists i \leq n$ such that $\bigoplus_{x \in C_i} (F \circ G)(x) = 1$.

Let $I_i : \mathbb{F}_2^n \to \mathbb{F}_2$ be such that $I_i(x) = 1 \Leftrightarrow x \in C_i$:

$$\bigoplus_{x \in C_i} (F \circ G)(x) = \bigoplus_{x \in \mathbb{F}_2^n} F\big(G(x)\big) \times I_i(x) \; ,$$

and let $y = G(x)$. Then:

$$\bigoplus_{x \in C_i} (F \circ G)(x) = \bigoplus_{y \in \mathbb{F}_2^n} F(y) \times I_i\big(G^{-1}(y)\big) \; .$$

This sum is equal to 1 if and only if $x \mapsto F(x) \times I_i\big(G^{-1}(x)\big)$ has degree n.
I_i is affine ($I_i(x) = 1 + x_i$). Thus, the sum can be equal to 1 only if

$$\deg(F) + \deg(G^{-1}) \geq n \; .$$

$\square$

Proposed Updates to the Thesis

- Better justification for HDIM-based attack against SPNs.
- Add S-Boxes of Skinny-64 and Skinny-128.
- Add Chiasmus to the list of broken S-Boxes; add CSA-BC to the list of unknown S-Boxes. Add CSS?
- Update LWC review.
- Add brief description of SPARX external cryptanalysis.

Bibliography I

Abdelkhalek, A., Tolba, M., and Youssef, A. (2017).
Impossible differential attack on reduced round SPARX-64/128.
In Joye, M. and Nitaj, A., editors, *Progress in Cryptology – AFRICACRYPT 2017*, volume To appear of *Lecture Notes in Computer Science*, page To appear. Springer International Publishing.

Bel. St. Univ. (2011).
"Information technologies. Data protection. Cryptographic algorithms for encryption and integrity control.".
State Standard of Republic of Belarus (STB 34.101.31-2011).
http://apmi.bsu.by/assets/files/std/belt-spec27.pdf.

Biryukov, A., Derbez, P., and Perrin, L. (2015).
Differential analysis and meet-in-the-middle attack against round-reduced TWINE.
In [Leander, 2015], pages 3–27.

Biryukov, A., Khovratovich, D., and Perrin, L. (2017).
Multiset-algebraic cryptanalysis of reduced Kuznyechik, Khazad, and secret SPNs.
IACR Transactions on Symmetric Cryptology, 2016(2):226–247.

Bibliography II

Biryukov, A., Leurent, G., and Perrin, L. (2016a).
Cryptanalysis of Feistel networks with secret round functions.
In Dunkelman, O. and Keliher, L., editors, *Selected Areas in Cryptography – SAC 2015*, volume 9566 of *Lecture Notes in Computer Science*, pages 102–121, Cham. Springer International Publishing.

Biryukov, A. and Perrin, L. (2015).
On reverse-engineering S-boxes with hidden design criteria or structure.
In Gennaro, R. and Robshaw, M. J. B., editors, *Advances in Cryptology – CRYPTO 2015, Part I*, volume 9215 of *Lecture Notes in Computer Science*, pages 116–140. Springer, Heidelberg.

Biryukov, A., Perrin, L., and Udovenko, A. (2016b).
Reverse-engineering the S-box of streebog, kuznyechik and STRIBOBr1.
In Fischlin, M. and Coron, J.-S., editors, *Advances in Cryptology – EUROCRYPT 2016, Part I*, volume 9665 of *Lecture Notes in Computer Science*, pages 372–402. Springer, Heidelberg.

Bibliography III

Canteaut, A., Duval, S., and Perrin, L. (2017).
A generalisation of Dillon's APN permutation with the best known differential and nonlinear properties for all fields of size 2^{4k+2}.
IEEE Transactions on Information Theory, (to appear).

Derbez, P. and Perrin, L. (2015).
Meet-in-the-middle attacks and structural analysis of round-reduced PRINCE.
In [Leander, 2015], pages 190–216.

Dinu, D., Perrin, L., Udovenko, A., Velichkov, V., Großschädl, J., and Biryukov, A. (2016).
Design strategies for ARX with provable bounds: Sparx and LAX.
In Cheon, J. H. and Takagi, T., editors, *Advances in Cryptology – ASIACRYPT 2016, Part I*, volume 10031 of *Lecture Notes in Computer Science*, pages 484–513. Springer, Heidelberg.

GOST (2012).
Gost r 34.11-2012: Streebog hash function.
https://www.streebog.net/.

Bibliography IV

GOST (2015).
(GOST R 34.12–2015) information technology – cryptographic data security – block ciphers.
http://tc26.ru/en/standard/gost/GOST_R_34_12_2015_ENG.pdf.

Leander, G., editor (2015).
Fast Software Encryption – FSE 2015, volume 9054 of *Lecture Notes in Computer Science*. Springer, Heidelberg.

NIST (1998).
Skipjack and KEA algorithms specifications, v2.0.
http://csrc.nist.gov/groups/ST/toolkit/documents/skipjack/skipjack.pdf.

Perrin, L. and Khovratovich, D. (2015).
Collision spectrum, entropy loss, T-sponges, and cryptanalysis of GLUON-64.
In Cid, C. and Rechberger, C., editors, *Fast Software Encryption – FSE 2014*, volume 8540 of *Lecture Notes in Computer Science*, pages 82–103. Springer, Heidelberg.

Bibliography V

Perrin, L. and Udovenko, A. (2016).
Algebraic insights into the secret feistel network.
In Peyrin, T., editor, *Fast Software Encryption – FSE 2016*, volume 9783 of *Lecture Notes in Computer Science*, pages 378–398. Springer, Heidelberg.

Perrin, L. and Udovenko, A. (2017).
Exponential S-boxes: a link between the S-boxes of BelT and Kuznyechik/Streebog.
IACR Transactions on Symmetric Cryptology, 2016(2):99–124.

Perrin, L., Udovenko, A., and Biryukov, A. (2016).
Cryptanalysis of a theorem: Decomposing the only known solution to the big APN problem.
In Robshaw, M. and Katz, J., editors, *Advances in Cryptology – CRYPTO 2016, Part II*, volume 9815 of *Lecture Notes in Computer Science*, pages 93–122. Springer, Heidelberg.