ISO Update

Who knew standardization could be this fun?

Léo Perrin

Inria, France

January 20, 2020

Dagstuhl 20041

How are **Streebog** and **Kuznyechik** doing?

Outline

Plan of this Section

Kuznyechik/Streebog

Streebog

Type Hash function

Publication 2012

Kuznyechik

Type Block cipher

Publication 2015

Kuznyechik/Streebog

Streebog

Type Hash function

Publication 2012

Kuznyechik

Type Block cipher

Publication 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.

Timeline

By March 2016, Kuznyechik and Streebog were both **GOST** standards and **IETF** RFCs.

May 2016	Publication of the first decomposition (TU-decomposition)	EC'16
Feb 2017	Publication of the second decomposition (Belarus-like)	FSE'17

Timeline

By March 2016, Kuznyechik and Streebog were both **GOST** standards and **IETF** RFCs.

May 2016	Publication of the first decomposition (TU-decomposition)	EC'16
Feb 2017	Publication of the second decomposition (Belarus-like)	FSE'17
Jun. 2018	Luxembourg representatives at ISO asked me about these	
Oct. 2018	**ISO** standardization of Streebog (ISO 10118-3)	

Timeline

By March 2016, Kuznyechik and Streebog were both **GOST** standards and **IETF** RFCs.

May 2016	Publication of the first decomposition (TU-decomposition)	EC'16
Feb 2017	Publication of the second decomposition (Belarus-like)	FSE'17
Jun. 2018	Luxembourg representatives at ISO asked me about these	
Oct. 2018	**ISO** standardization of Streebog (ISO 10118-3)	
Dec. 2018	**Publication of the TKlog decomposition**	FSE'19
Apr. 2019	**ISO** decision to postpone the inclusion of Kuznyechik	
Apr. 2019	Russian law mandating the use of Russian algorithms	

Timeline

By March 2016, Kuznyechik and Streebog were both **GOST** standards and **IETF** RFCs.

May 2016	Publication of the first decomposition (TU-decomposition)	EC'16
Feb 2017	Publication of the second decomposition (Belarus-like)	FSE'17
Jun. 2018	Luxembourg representatives at ISO asked me about these	
Oct. 2018	**ISO** standardization of Streebog (ISO 10118-3)	
Dec. 2018	Publication of the TKlog decomposition	FSE'19
Apr. 2019	**ISO** decision to postpone the inclusion of Kuznyechik	
Apr. 2019	Russian law mandating the use of Russian algorithms	

Oct. 2019 **ISO** had to make a decision

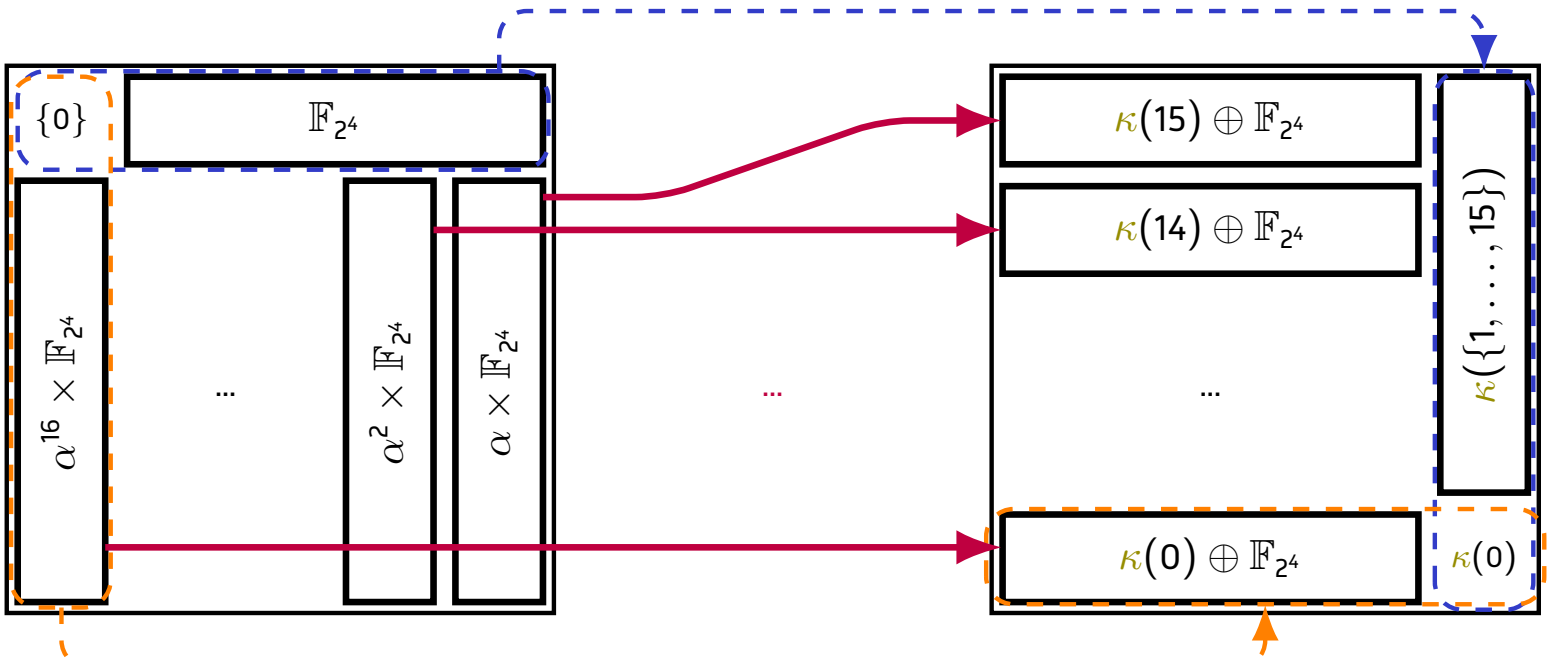
Timeline

By March 2016, Kuznyechik and Streebog were both **GOST** standards and **IETF** RFCs.

May 2016	Publication of the first decomposition (TU-decomposition)	EC'16
Feb 2017	Publication of the second decomposition (Belarus-like)	FSE'17
Jun. 2018	Luxembourg representatives at ISO asked me about these	
Oct. 2018	**ISO** standardization of Streebog (ISO 10118-3)	
Dec. 2018	Publication of the TKlog decomposition	FSE'19
Apr. 2019	**ISO** decision to postpone the inclusion of Kuznyechik	
Apr. 2019	Russian law mandating the use of Russian algorithms	
Summer 2019	Time to act	
Oct. 2019	**ISO** had to make a decision	

The TKlog Structure

$$\pi : \begin{cases} \mathbb{F}_{2^8} & \to \mathbb{F}_{2^8} & \\ 0 & \mapsto \kappa(0) & \\ \alpha^{17j} & \mapsto \kappa(16-j) & \text{for } 1 \le j \le 15 \\ \alpha^{i+17j} & \mapsto \kappa(16-i) \oplus (\alpha^{17})^{s(j)} & \text{for } 0 < i, 0 \le j < 16 \end{cases}$$



RUnet

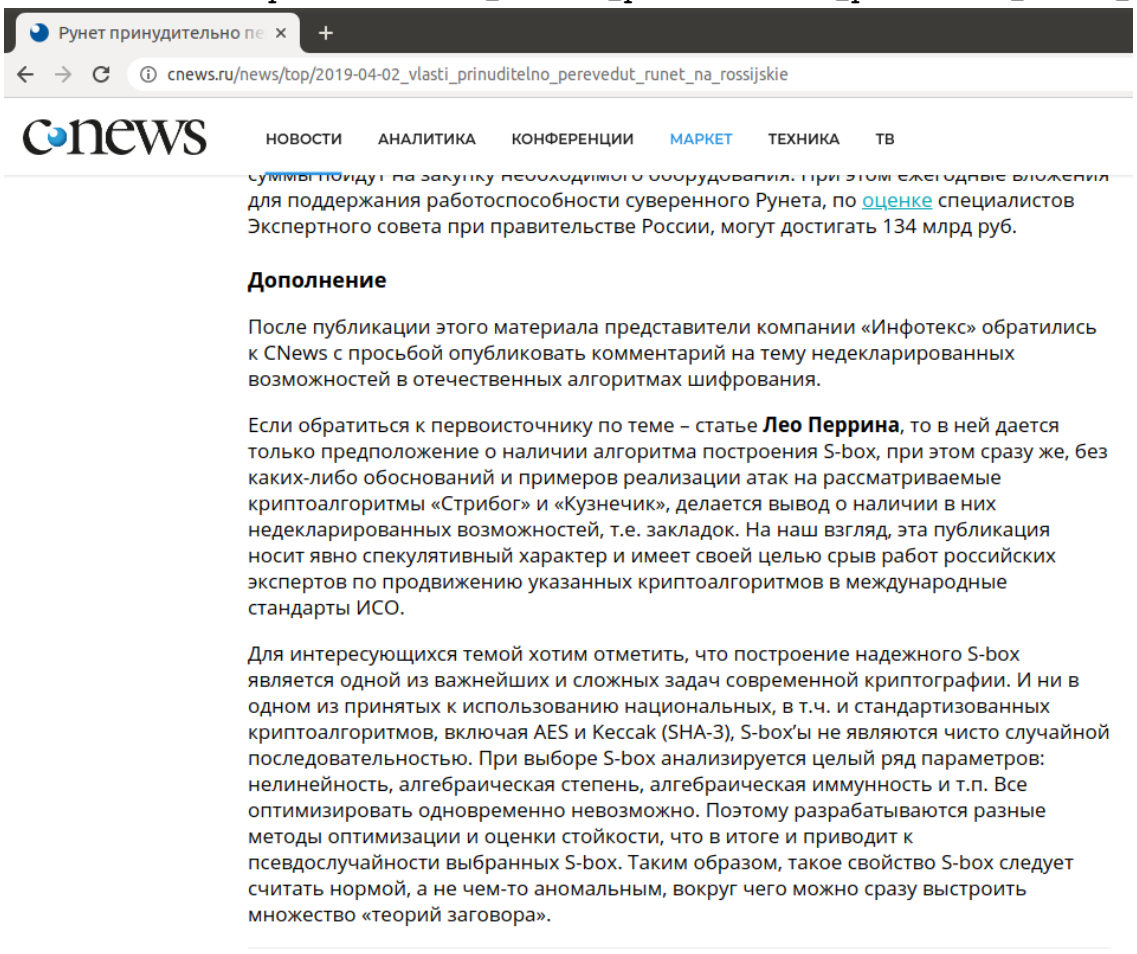
The use of national encryption standards is being made **mandatory** in Russia.

https://www.cnews.ru/news/top/2019-04-02_vlasti_prinuditelno_perevedut_runet_na_rossijskie

RUnet

The use of national encryption standards is being made **mandatory** in Russia.

https://www.cnews.ru/news/top/2019-04-02_vlasti_prinuditelno_perevedut_runet_na_rossijskie



Рунет принудительно пе ×

cnews.ru/news/top/2019-04-02_vlasti_prinuditelno_perevedut_runet_na_rossijskie

cnews НОВОСТИ АНАЛИТИКА КОНФЕРЕНЦИИ МАРКЕТ ТЕХНИКА ТВ

суммы пойдут на закупку необходимого оборудования. При этом ежегодные вложения для поддержания работоспособности суверенного Рунета, по оценке специалистов Экспертного совета при правительстве России, могут достигать 134 млрд руб.

Дополнение

После публикации этого материала представители компании «Инфотекс» обратились к CNews с просьбой опубликовать комментарий на тему недекларированных возможностей в отечественных алгоритмах шифрования.

Если обратиться к первоисточнику по теме – статье **Лео Перрина**, то в ней дается только предположение о наличии алгоритма построения S-box, при этом сразу же, без каких-либо обоснований и примеров реализации атак на рассматриваемые криптоалгоритмы «Стрибог» и «Кузнечик», делается вывод о наличии в них недекларированных возможностей, т.е. закладок. На наш взгляд, эта публикация носит явно спекулятивный характер и имеет своей целью срыв работ российских экспертов по продвижению указанных криптоалгоритмов в международные стандарты ИСО.

Для интересующихся темой хотим отметить, что построение надежного S-box является одной из важнейших и сложных задач современной криптографии. И ни в одном из принятых к использованию национальных, в т.ч. и стандартизованных криптоалгоритмов, включая AES и Keccak (SHA-3), S-box'ы не являются чисто случайной последовательностью. При выборе S-box анализируется целый ряд параметров: нелинейность, алгебраическая степень, алгебраическая иммунность и т.п. Все оптимизировать одновременно невозможно. Поэтому разрабатываются разные методы оптимизации и оценки стойкости, что в итоге и приводит к псевдослучайности выбранных S-box. Таким образом, такое свойство S-box следует считать нормой, а не чем-то аномальным, вокруг чего можно сразу выстроить множество «теорий заговора».

What its Designers Said (at ISO)

questioned is the S-box π. This S-box was chosen from Streebog hash-function and it was synthesized in 2007. Note that through many years of cryptanalysis no weakness of this S-box was found. The S-box π was obtained by pseudo-random search and the following properties were taken into account.

[...]

No secret structure was enforced during construction of the S-box. At the same time, it is obvious that for any transformation a lot of representations are possible (see, for example, a lot of AES S-box representations).

What its Designers Said (at ISO)

questioned is the S-box π. This S-box was chosen from Streebog hash-function and it was synthesized in 2007. Note that through many years of cryptanalysis no weakness of this S-box was found. The S-box π was obtained by pseudo-random search and the following properties were taken into account.

[...]

No secret structure was enforced during construction of the S-box. At the same time, it is obvious that for any transformation a lot of representations are possible (see, for example, a lot of AES S-box representations).

In private conversations, they explicitely said they used a Fisher-Yates shuffle to generate random S-boxes.

Plan of this Section

General Question

How "far" is the behaviour of a specific S-box from that of a "random S-box"?

General Question

How "far" is the behaviour of a specific S-box from that of a "random S-box"?

How likely is it for a random S-box to have a "structure"?

Definition

```
p(x){unsigned char*k="@`rFTDVbpPB
vdtfR@\xacp?\xe2>4\xa6\xe9{z\xe3q
5\xa7\xe8",a=2,l=0,b=17;while(x&&
(l++,a^x))a=2*a^a/128*29;return l
%b?k[l%b]^k[b+l/b]^b:k[l/b]^188;}
```

165 ASCII characters that fit on **7** bits: this program is **1155**-bit long.

`https://codegolf.stackexchange.com/questions/186498/`

`proving-that-a-russian-cryptographic-standard-is-too-structured`

Let $P(S)$ be the bitlength of a C implementation of $S \in \mathfrak{S}_{2^n}$.

Definition (Kolmogorov Anomaly)

The Kolmogorov Anomaly of S for C is the opposite of the $\log_2$ of **the probability that a random S-box has a C implementation at most as long** as that of S.

Estimating the Kolmogorov Anomaly

How to estimate it?

- (≤ 1155)-bit C programs implementing 8-bit permutations
- (≤ 1155)-bit strings
- $\mathfrak{S}_{2^8}$

For π, we get:

$$\frac{\#(\leq 1155)\text{-bit C prog.}}{|\mathfrak{S}_{2^8}|} \leq \frac{\#(\leq 1155)\text{-bit strings.}}{|\mathfrak{S}_{2^8}|} = \frac{2^{1156}-1}{256!} \approx 2^{-528},$$

meaning that the Kolmogorov anomaly of π for C is at least 528.

Plan of this Section

Artist Rendition

Discussions with the Alleged Designers, Allegory.
Python M., 1969.

An S-box is always like this (1/2)

1. Unfortunately, we lost theg generation program so we can't show it to you
2. S-boxes always have a structure, why do you complain about this one and not about this AES?
3. If you optimize the differential/linear properties, a structure will appear
4. You are just a mathematician, in the *real world*™ we don't phase out algorithms unless we have an attack.

[1] See excellent write up at `https://crypto.stackexchange.com/questions/75456/how-to-check-whether-the-permutation-is-random-or-not`

An S-box is always like this (1/2)

1. Unfortunately, we lost theg generation program so we can't show it to you

   Quite convenient

2. S-boxes always have a structure, why do you complain about this one and not about this AES?

   No claims of randomness from the AES designers

3. If you optimize the differential/linear properties, a structure will appear

   Simply not true, it also does not match other anomalies[1]

4. You are just a mathematician, in the *real world*™ we don't phase out algorithms unless we have an attack.

   I never said I had an attack, but I do think **lying** *is bad (even in the* real world™*).*

[1] See excellent write up at `https://crypto.stackexchange.com/questions/75456/how-to-check-whether-the-permutation-is-random-or-not`

An S-box is always like this (2/2)

5 There is something about C that allows you to find this implementation, it merely says something about the C language and not π.

6 There are all kind of 8-bit bijective S-box structures in the literature!

	Special polynomials	2^{22}
	Generation using paths (?)	2^{255}
†	TU_4-decomposition (w/ mult)	2^{88}
→	**TU_4-decomposition (called "F-construction")**	$\mathbf{2^{1417}}$
†	Feistel 1r	2^{64}
	Feistel 1r (weird)	2^{130}
†	Misty 2r	2^{88}
	SPN 1r (balanced or not)	2^{781}
	SPN 3r (Iceberg-like)	2^{104}
	SPN 3r (Khazad-like)	2^{88}
	SPN 2r (Crypton v1)	2^{152}
†	SPN 2r (CLEFIA-style)	2^{177}
†	Lai-Massey (FLY-style)	2^{152}
†	Lai-Massey (Whirlpool-style)	2^{88}
†	Perrin (neither mine nor a permutation)	2^{304}
	LFSRs	2^{12}
	Total (with affine-equivalence)	$\approx 2^{1488}$

2^{1488} **"is approaching"** 2^{1683}, so the presence of a structure is normal.

An S-box is always like this (2/2)

5. There is something about C that allows you to find this implementation, it merely says something about the C language and not π.

That's not even wrong.

6. There are all kind of 8-bit bijective S-box structures in the literature!

	Special polynomials	2^{22}
	Generation using paths (?)	2^{255}
†	TU_4-decomposition (w/ mult)	2^{88}
→	**TU_4-decomposition (called "F-construction")**	$\mathbf{2^{1417}}$
†	Feistel 1r	2^{64}
	Feistel 1r (weird)	2^{130}
†	Misty 2r	2^{88}
	SPN 1r (balanced or not)	2^{781}
	SPN 3r (Iceberg-like)	2^{104}
	SPN 3r (Khazad-like)	2^{88}
	SPN 2r (Crypton v1)	2^{152}
†	SPN 2r (CLEFIA-style)	2^{177}
†	Lai-Massey (FLY-style)	2^{152}
†	Lai-Massey (Whirlpool-style)	2^{88}
†	Perrin (neither mine nor a permutation)	2^{304}
	LFSRs	2^{12}
	Total (with affine-equivalence)	$\approx 2^{1488}$

2^{1488} **"is approaching"** 2^{1683}, so the presence of a structure is normal.

2^{1488} is in fact $\approx 2^{196}$ times smaller than 256! $\approx 2^{1683.996}$.

They Actually Said That (see ISO/IEC JTC 1/SC 27/WG2 N 2063)

2.3 Shift registers

One more way of substitution generation is shifting number $x \in \mathrm{GF}(2^8)$ by a linear feedback shift register (see Fig. 1) by a number of steps $n \in \{0, 255\}$. Since it is necessary that the substitution is full-round, the polynomial of degree 8, whose coefficients determine the feedback function, is required to be primitive. Then the number of substitutions is set by the choice of the number n and the number of primitive polynomials.

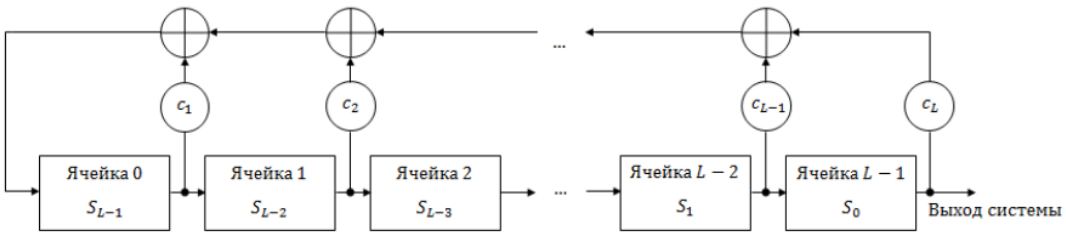


Figure 1

The number of polynomials over $\mathrm{GF}(2)$ is $\frac{\varphi(2^8-1)}{8} = \frac{128}{8} = 16$, so approximately $2^8 \cdot 2^4 = 2^{12}$ substitutions may be obtained this way.

[...]

this word. Based on this remark we apply the affine transformation only to the output. The result is 2^{1488} substitutions in total. And this size is approaching the total number of 16 element substitutions, which equals to $256! \approx 2^{1683}$.

Best Argument

7 **Anti-Russia bias !!1!**

No other country would be treated like this!

Best Argument

7 Anti-Russia bias !!1!

No other country would be treated like this!

Except for the US

less than a year ago

who said the same thing

Plan of this Section

Conclusion

How are **Streebog** and **Kuznyechik** doing?

	Streebog	Kuznyechik
IETF	**Good**	**Good**
ISO	**Good**	**Bad**

$\Longrightarrow$ 3 open problems

Conclusion

How are **Streebog** and **Kuznyechik** doing?

	Streebog	Kuznyechik
IETF	**Good**	**Good**
ISO	**Good**	**Bad**

$\Longrightarrow$ 3 open problems

TBC "debate", IETF procedures...
Standardization is a lot more fun than I thought!

Conclusion

How are **Streebog** and **Kuznyechik** doing?

	Streebog	Kuznyechik
IETF	**Good**	**Good**
ISO	**Good**	**Bad**

$\Longrightarrow$ 3 open problems

TBC "debate", IETF procedures...
Standardization is a lot more fun than I thought!

Thank you!

Translation

(with thanks to google translate)

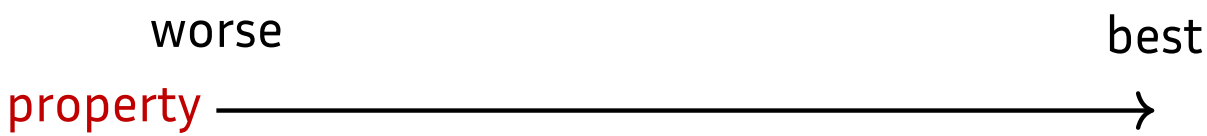
[...], representatives of the **Infotex company** asked CNews to publish a comment on the topic of undeclared capabilities in domestic encryption algorithms.

Leo Perrin's article [...] only conjectures that there is an algorithm for constructing an S-box, while immediately, without any justification and examples of attacks to "Stribog" and "Grasshopper", it is concluded that there are undeclared functionalities in them, i.e. backdoors. In our opinion, this publication is clearly speculative in nature and aims to disrupt the work of Russian experts in promoting these cryptographic algorithms in international ISO standards.

[...] in standard encryption algorithms, including AES and Keccak (SHA-3), S-boxes are not purely random sequences. When choosing an S-box, a number of parameters are taken into account: nonlinearity, algebraic degree, algebraic immunity, etc. [...] Thus, such an S-box property should be considered the norm, and not something abnormal, around which you can immediately build a lot of "**conspiracy theories**."

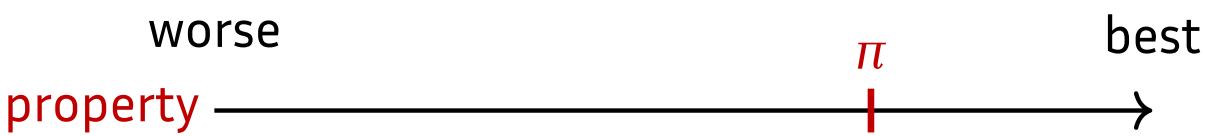
General Approach

1. Choose an S-box property with a value in a partially ordered set (i.e. $\mathbb{N}$)



General Approach

1. Choose an S-box property with a value in a partially ordered set (i.e. $\mathbb{N}$)
2. Compute it for the specific target



General Approach

1. Choose an S-box property with a value in a partially ordered set (i.e. $\mathbb{N}$)
2. Compute it for the specific target
3. Evaluate the number of S-boxes with a worse and a better property

General Approach

1. Choose an S-box property with a value in a partially ordered set (i.e. $\mathbb{N}$)
2. Compute it for the specific target
3. Evaluate the number of S-boxes with a worse and a better property

Negative Anomaly

$$\overline{\mathcal{A}}(\pi) = -\log_2\left(\frac{\#\text{worse S-boxes}}{(2^n)!}\right)$$

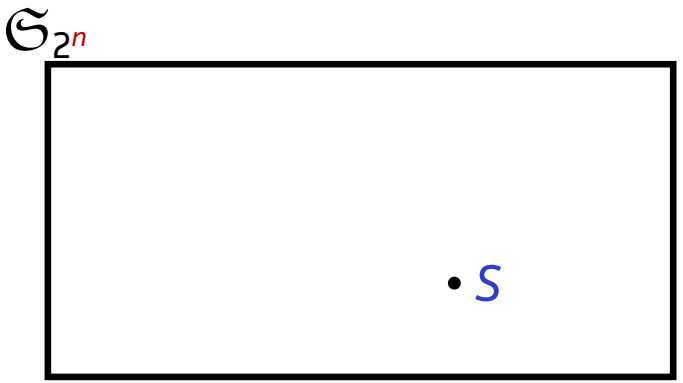
Positive Anomaly

$$\mathcal{A}(\pi) = -\log_2\left(\frac{\#\text{better S-boxes}}{(2^n)!}\right)$$

Bad Idea: Using Instance-Tailored Properties

Let $S \in \mathfrak{S}_{2^n}$ be the studied S-box. We define a property P_S as

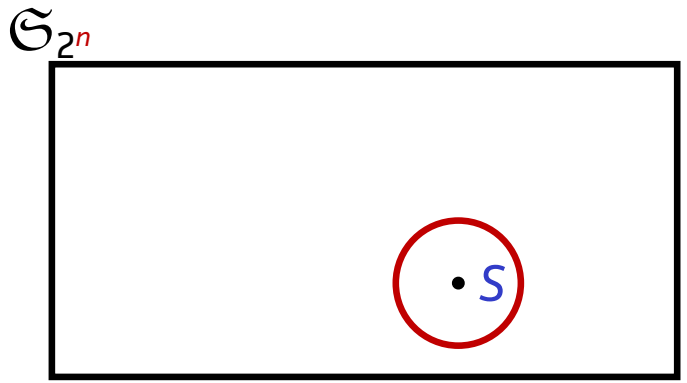
$$P_S : \begin{cases} \mathfrak{S}_{2^n} & \to \mathbb{N} \\ F & \mapsto \#\left\{x \in \mathbb{F}_2^n, F(x) = S(x)\right\} \end{cases}.$$



Bad Idea: Using Instance-Tailored Properties

Let $S \in \mathfrak{S}_{2^n}$ be the studied S-box. We define a property P_S as

$$P_S : \begin{cases} \mathfrak{S}_{2^n} & \to \mathbb{N} \\ F & \mapsto \#\left\{x \in \mathbb{F}_2^n, F(x) = S(x)\right\} \end{cases}.$$

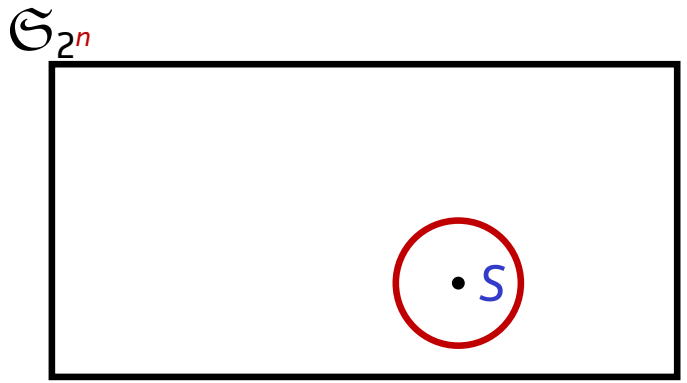


$$\left\{F, P_S(F) \geq 5\right\}$$

Bad Idea: Using Instance-Tailored Properties

Let $S \in \mathfrak{S}_{2^n}$ be the studied S-box. We define a property P_S as

$$P_S : \begin{cases} \mathfrak{S}_{2^n} & \to \mathbb{N} \\ F & \mapsto \#\left\{x \in \mathbb{F}_2^n, F(x) = S(x)\right\} \end{cases}.$$



$$\left\{F, P_S(F) \geq 5\right\}$$

The corresponding anomaly is useless: we can choose S arbitrarily!

Experimental Results

Type	Cipher	Differential		Linear		Boomerang	
		$A^d(s)$	$\overline{A}^d(s)$	$A^\ell(s)$	$\overline{A}^\ell(s)$	$A^b(s)$	$\overline{A}^b(s)$
Inverse	AES	7382.1	0.00	3329.4	0.00	9000.1	0.0
TKlog	Kuznyechik	80.6	0.00	34.4	0.00	14.2	0.0
SPN (2S)	CLEFIA_S0	2.6	0.2	25.6	0.0	0.0	15.6
	Twofish_p0	1.4	0.7	3.2	0.2	0.0	33.8
Feistel	ZUC_S0	16.2	0.0	3.2	0.2	0.0	NaN
Hill climbing	Kalyna_pi0	104.2	0.0	235.8	0.00	29.7	0.00
Random	MD2	1.4	0.7	0.1	2.4	1.0	0.4
Unknown	Skipjack	0.2	1.9	54.4	0.0	1.0	0.4