Towards New International Cryptographic Standards

Designing and Breaking Cryptography

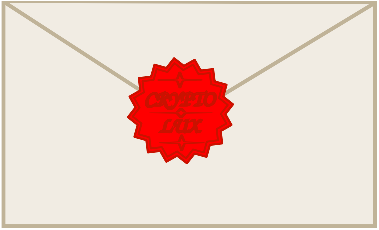
Léo Perrin
Cosmiq TEAM
Inria, Paris, France

FIC 2020, Lille

We (the **Cosmiq** team) are working on the foundations of cryptography.

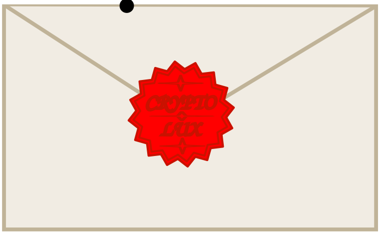
1. What kind of algorithms do we study?
2. Why do we design new ones?
3. What kind of flaws do we find in other ones?

What is Cryptography?

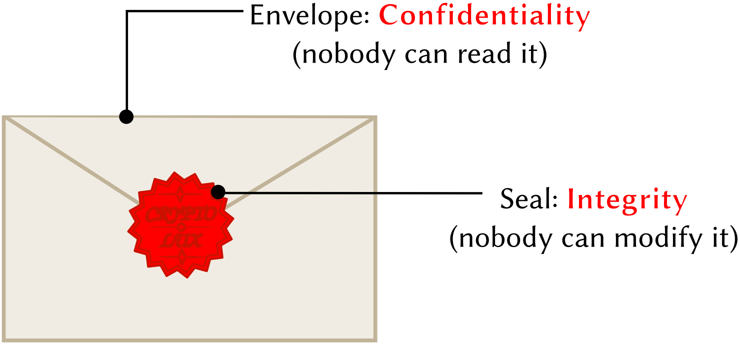


What is Cryptography?

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

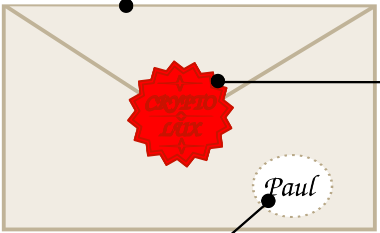


What is Cryptography?



What is Cryptography?

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



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

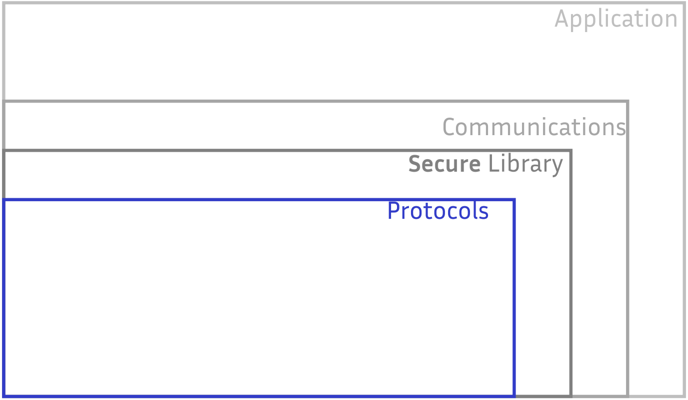
Signature: **Authentication**
(it was written by the right person)

How Is It Used?

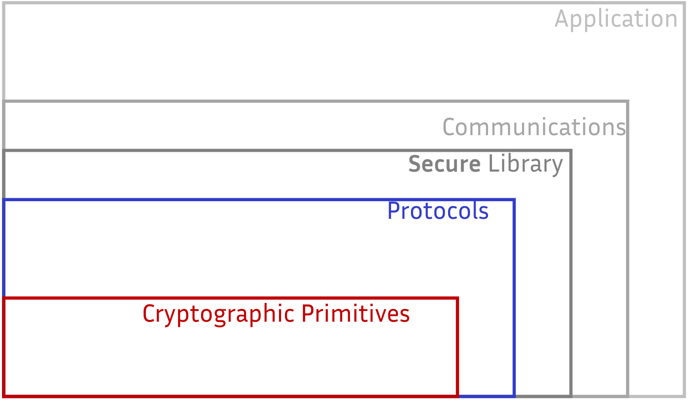
How Is It Used?

How Is It Used?

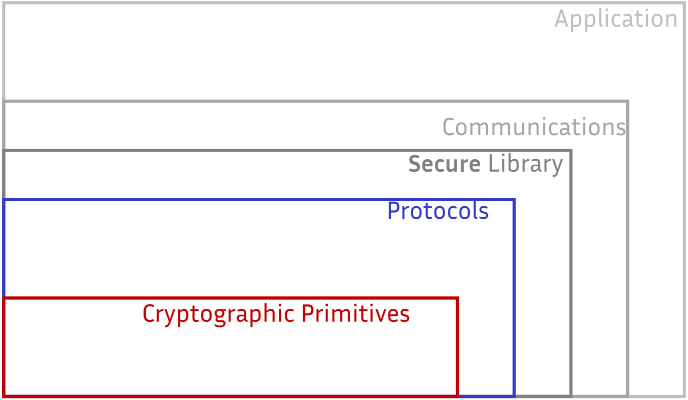
How Is It Used?



How Is It Used?



How Is It Used?



RSA, AES, SHA-256, ECDSA...

What Do Primitives Do?

A cryptographic primitive is a **basic building block** ; it has a very simple API but very sophisticated inner workings!

The block cipher

For any k-bit long key κ, E_κ is a **permutation** of $\{0, 1\}^n$.
Typically, $n \in \{64, 128\}$ and $k \in \{128, 256\}$.

To ensure **security**: no matter how many pairs $(x, E_\kappa(x))$ are known, it is impossible to recover k[1]

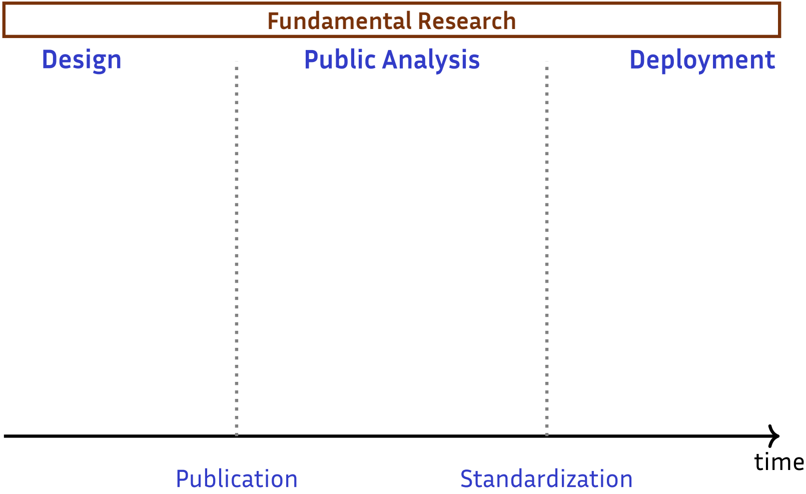
[1] Except by trying all possible κ which has 2^k possible values.

How are the primitives used in practice chosen?

Life Cycle of a Cryptographic Primitive

Fundamental Research

Life Cycle of a Cryptographic Primitive



Life Cycle of a Cryptographic Primitive

Fundamental Research

Design
Small teams

Public Analysis
Academic community

Deployment
Industry

time

Publication
Conf., competition

Standardization
NIST, ISO, IETF...

Life Cycle of a Cryptographic Primitive

Fundamental Research

Design

Small teams

- Scope statement
- Algorithm specification
- Design choices justifications
- Security analysis

Public Analysis

Academic community

Deployment

Industry

time

Publication

Conf., competition

Standardization

NIST, ISO, IETF...

Life Cycle of a Cryptographic Primitive

Fundamental Research

Design
Small teams

- Scope statement
- Algorithm specification
- Design choices justifications
- Security analysis

Public Analysis
Academic community

Deployment
Industry

time

Publication
Conf., competition

Standardization
NIST, ISO, IETF...

Life Cycle of a Cryptographic Primitive

Fundamental Research

Design

Small teams

- Scope statement
- Algorithm specification
- Design choices justifications
- Security analysis

Public Analysis

Academic community

Try and break published algorithms

Deployment

Industry

time

Publication

Conf., competition

Standardization

NIST, ISO, IETF...

Life Cycle of a Cryptographic Primitive

Fundamental Research

Design

Small teams

- Scope statement
- Algorithm specification
- Design choices justifications
- Security analysis

Public Analysis

Academic community

Try and break published algorithms

Deployment

Industry

time

Publication

Conf., competition

Standardization

NIST, ISO, IETF...

Life Cycle of a Cryptographic Primitive

Fundamental Research

Design

Small teams

- Scope statement
- Algorithm specification
- Design choices justifications
- Security analysis

Public Analysis

Academic community

Try and break published algorithms

Unbroken algorithms are eventually trusted

Deployment

Industry

time

Publication

Conf., competition

Standardization

NIST, ISO, IETF...

Life Cycle of a Cryptographic Primitive

Fundamental Research

Design
Small teams
- Scope statement
- Algorithm specification
- Design choices justifications
- Security analysis

Public Analysis
Academic community

Try and break published algorithms

Unbroken algorithms are eventually trusted

Deployment
Industry

Implements algorithms in actual products...
...unless a new attack is found

time

Publication
Conf., competition

Standardization
NIST, ISO, IETF...

Breaking the Pipeline

Fundamental Research

Design

Small teams

- Scope statement
- Algorithm specification
- Design choices justifications
- Security analysis

Public Analysis

Academic community

Try and break published algorithms

Unbroken algorithms are eventually trusted

Deployment

Industry

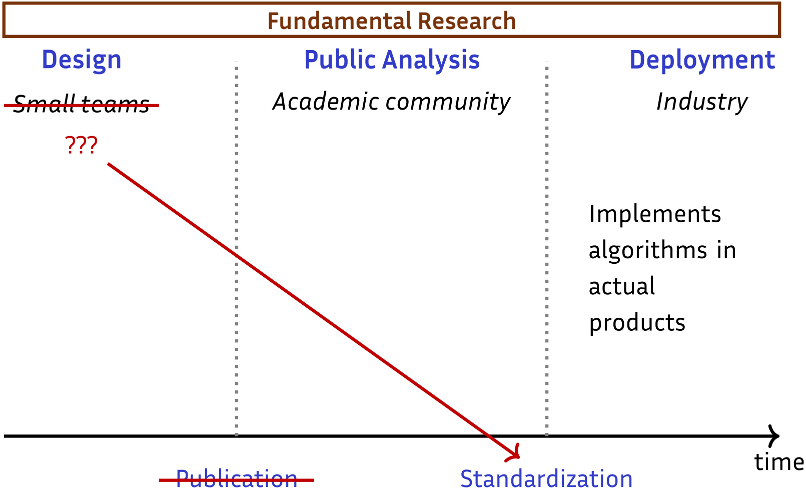
Implements algorithms in actual products

Publication

Standardization

time

Breaking the Pipeline



Breaking the Pipeline

Fundamental Research

Design	**Public Analysis**	**Deployment**
Small teams	*Academic community*	*Industry*

ts

s in

Hidden defect?

time

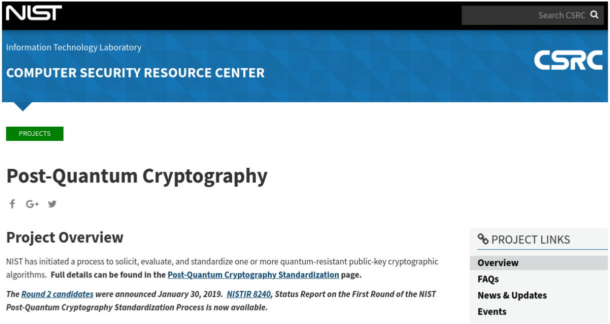
Publication | Standardization

Primitives we designed
Primitives we attacked

Primitives we designed

Primitives we attacked

Post-Quantum Public Key



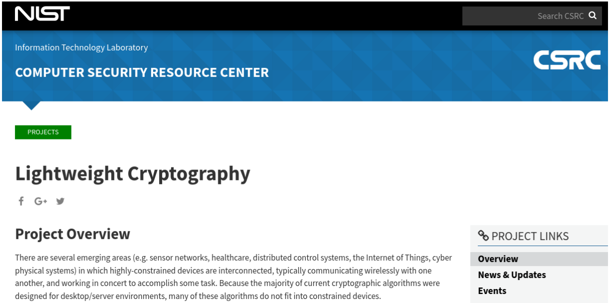
Quantum computers will **break** current public key algorithms

$\Longrightarrow$ we need new algorithms!

Cosmiq Involvement

3 Cosmiq candidates made it to the second round! (**Bike**, **Classic McEliece**, and **Rollo**)

Ligthweight Secret Key



IoT devices cannot handle the (low!) **complexity** of current symmetric ciphers.

$\Longrightarrow$ we need new algorithms!

Cosmiq Involvement

3 Cosmiq candidates made it to the second round! (**Saturnin**, **Sparkle**, **Spook**)

Primitives we designed

Primitives we attacked

Breaking SHA-1

SHA-1 is a hash function.

Collision Resistance

For a hash function H, it should not be possible to find messages x and y such that

$$H(x) = H(y)\ .$$

Cosmiq Involvement

It is possible **in practice** to find **meaningful** messages $a||x$ and $a||y$ where a and b are meaningful and such that

$$H(a||x) = H(a||y)$$

G. Leurent, T. Peyrin. *From Collisions to Chosen-Prefix Collisions – Application to Full SHA-1*. Eurocrypt 2019.

Finding Weird Patterns in Russian Standards

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).

Cosmiq Involvement

The designers of Streebog and Kuznyechik **are lying**. The probability that a **random** S-box is as **structured** as theirs is $< 2^{-1000}$ ($\approx$ winning the "loto" 60 times in a row).

Scientific publication: X. Bonnetain, L. Perrin, S. Tian. *Anomalies and Vector Space Search: Tools for S-box Analysis*. Asiacrypt 2019.

Conclusion

Cryptography is an **active** research area motivated by concrete needs for **standard** algorithms.

Conclusion

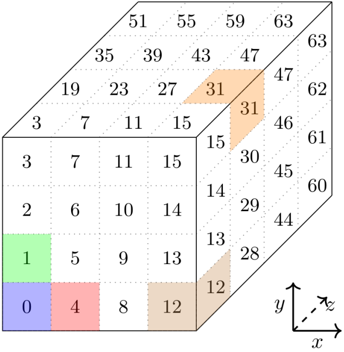
Cryptography is an **active** research area motivated by concrete needs for **standard** algorithms.

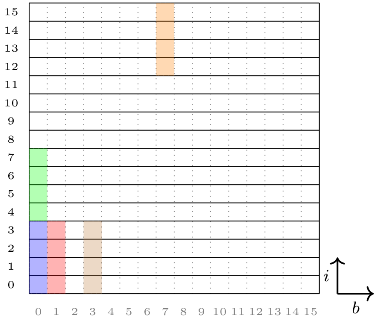
Thank you!

Delenda Russian Algo

Appendix

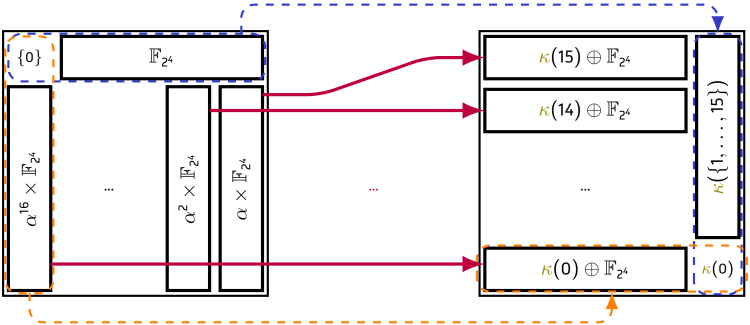
Saturnin


51 55 59 63
35 39 43 47
19 23 27 31
3 7 11 15
3 7 11 15
2 6 10 14
1 5 9 13
0 4 8 12
63 62 61 60
47 46 45 44
31 30 29 28
15 14 13 12
y
z
x


15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
i
b

The TKlog Structure

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



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.