Saturnin

A suite of lightweight symmetric algorithms for post-quantum security

Anne Canteaut[1] Sébastien Duval[2] Gaëtan Leurent[1] María Naya-Plasencia[1]
Léo Perrin[1] Thomas Pornin[3] André Schrottenloher[1]

[1] Inria, France [2] UCL Crypto Group, Belgium [3] NCC Group, Canada

Outline

Introduction

Our design goals

Goals

1. Strong security arguments
2. Quantum security
3. Efficient in hardware and software

Design choices

- SPN cipher
- **Wide-trail strategy** (AES-like)
- 256-bit keys **and blocks**
- Carefully chosen modes
- Bitslice design
- Small components

Saturnin in the LWC process

- 13 second-round candidates are based on block ciphers
- Saturnin is the **only block cipher with 256-bit blocks**
- Saturnin is the **only proposal** (cipher + modes) claiming security against superposition queries

Saturnin is the most efficient generalization of the AES wide-trail strategy to a 256-bit block size (in terms of security and implementation).

On quantum security

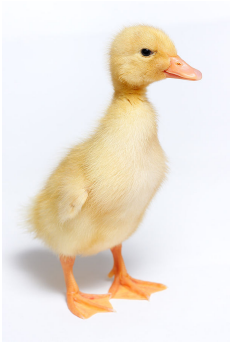
- A key size of **256 bits** mitigates quantum exhaustive search
- A **block size** of 256 bit mitigates attacks (on modes) at the quantum birthday bound ($2^{256/3} \simeq 2^{85.3}$)
  - Also simplifies the design of a hash function

We claim security against **classical and quantum attacks**. Quantum attackers can query the secret-key cipher / the modes in **superposition**.

- This is the strongest model available
- It is non-trivial
- It includes all intermediate definitions, and all use cases

On the name

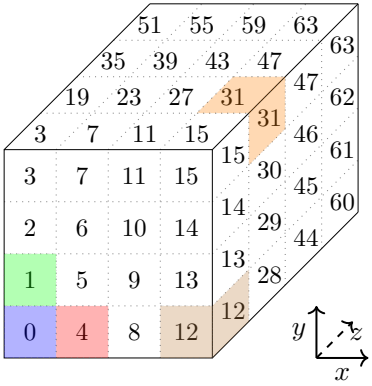
satyʁnɛ̃



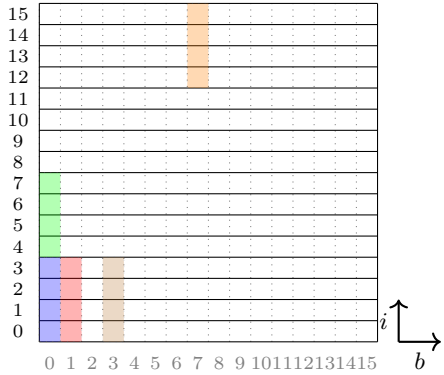
- Saturnin is a **famous french duck**
  - Kids TV show in the 60's
- The **duck** is well known **standard of lightness**
  - Historically used as a weight standard for witches
    [Sir Bedevere, *Monty Python and the Holy Grail*]
- The planet Saturn is associated to the **cube**
  [Kepler, *Mysterium Cosmographicum*]
  - Saturnin's state is represented as a cube

The Block Cipher

The state



A cube of $4 \times 4 \times 4$ nibbles of 4 bits



16 registers of 16 bits

Generic nibble index: $(x, y, z) \mapsto y + 4x + 16z$

The round function

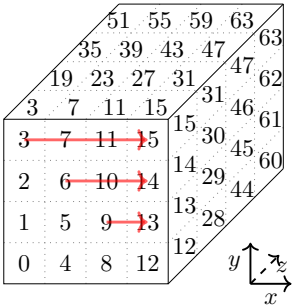
AES-inspired operations:

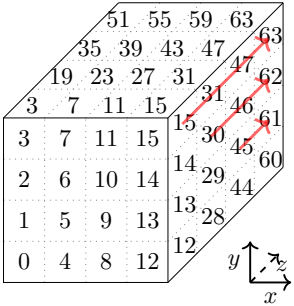
- **S-Box layer:** applies σ_0 to nibbles of even index, σ_1 to nibbles of odd index
- **Nibble permutation** SR: depends on the round number
- **Linear MixColumns:** applies a 4×4 MDS mapping over $\mathbb{F}_{2^4}$ to each column
- **Inverse** of SR
- **Sub-key addition**

The nibble permutation

Let r be the round index (starts at 0).

- $r \mod 4 = 1$: shift rows in "slices" (left)
- $r \mod 4 = 3$: shift rows in "sheets" (right)
- otherwise do nothing

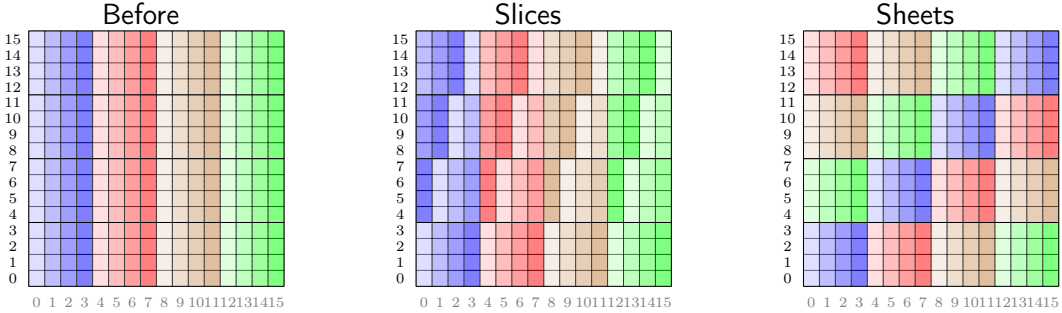




As registers

In the register representation:

- S and MC are bitsliced
- $\mathrm{SR}_{\mathrm{slices}}$ and $\mathrm{SR}_{\mathrm{sheets}}$ correspond to rotations in the registers



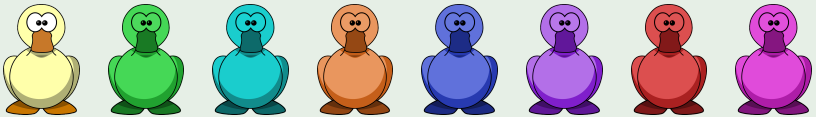
The subkey addition

Only at **odd rounds**.

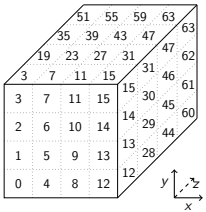
- $r \mod 4 = 3$: XOR the master key K
- $r \mod 4 = 1$: XOR K rotated by 20 nibbles
- otherwise do nothing

Round constants

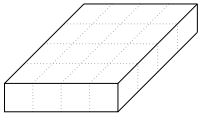
- Two 16-bit words XORed to the state (on 32 nibbles, one bit per nibble).
- Depend on the 4-bit **domain separator**



The Super S-Box representation



Supernibbles: **columns** in the cube



Let's have a look at 4 rounds:

$$\begin{cases} r = 0 \rightarrow \mathsf{S} \rightarrow \text{nothing} \rightarrow \mathsf{MC} \rightarrow \text{nothing} \rightarrow \text{nothing} \\ r = 1 \rightarrow \mathsf{S} \rightarrow \mathsf{SR}_{\text{slices}} \rightarrow \mathsf{MC} \rightarrow \mathsf{SR}^{-1}_{\text{slices}} \rightarrow \mathsf{K}_{\text{rot}} \end{cases}$$

$$\begin{cases} r = 2 \rightarrow \mathsf{S} \rightarrow \text{nothing} \rightarrow \mathsf{MC} \rightarrow \text{nothing} \rightarrow \text{nothing} \\ r = 3 \rightarrow \mathsf{S} \rightarrow \mathsf{SR}_{\text{sheets}} \rightarrow \mathsf{MC} \rightarrow \mathsf{SR}^{-1}_{\text{sheets}} \rightarrow \mathsf{K} \end{cases}$$

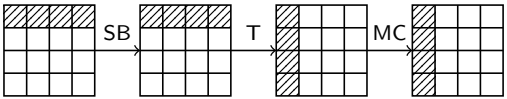
The Super S-Box representation (ctd.)

4 rounds of Saturnin apply:

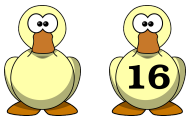
- A Super S-Box
- A linear transformation on the Super-columns
- A rotated key addition
- A Super S-Box
- The same linear transformation on the Super-rows
- A key addition

The Super S-Box representation (ctd.)

2 rounds of Saturnin (a Super-round) $\iff$ a single round of an AES on 16-bit nibbles, with a transposition (*i.e.* the block cipher Square).



- We use 10 Super-rounds for standard Saturnin
- We recommend 16 Super-rounds for related-key security (Faturnin)
- Our best key-recovery targets 7.5 Super-rounds



Security overview

- **Extensive analysis** of the AES is transferable to Saturnin
- 125 active S-Boxes for 8 rounds
- **4-bit S-Box** has optimal properties
  - $\delta = 4$
  - $\mathcal{L} = 8$
  - degree 3
- **Super S-Box** has good properties thanks to the MDS layer:
  - $\delta = 80$
  - $\mathcal{L} = 3072$
  - degree 9
- **Bounds** on 8-rounds trails
  - Differential: $p \leq 2^{-241.9}$
  - Linear: $p \leq 2^{-220.7}$

Modes of operation

Overview

The submission includes **three** modes of operation:

- **Saturnin-CTR-Cascade** for AEAD
- **Saturnin-Short** for small AE (< 128 bits)
- **Saturnin-Hash** for hashing

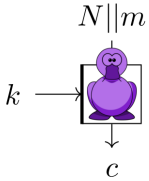
We use separate round constants for **domain separation**.

Known quantum security proofs:

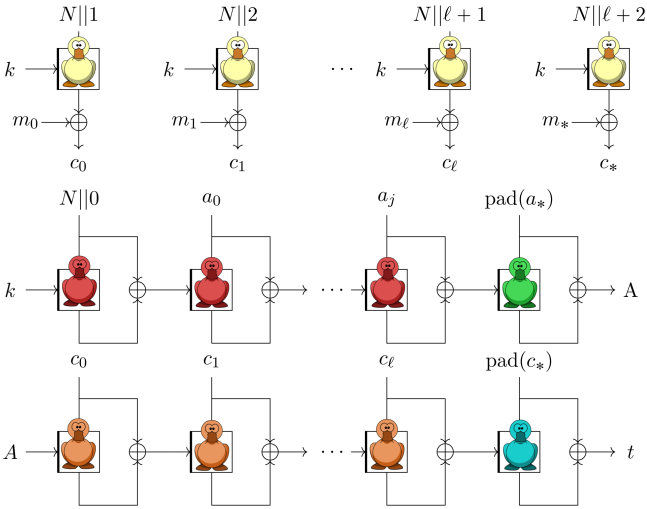
- Encrypt then MAC [Soukharev, Jao & Seshadri, PQCrypto 2016]
- CTR mode for encryption [Anand, Targhi, Tabia, Unruh, PQCrypto 2016]
- Cascade MAC [Song & Yun, Crypto '17]
- Quantum indifferentiability of Merkle-Damgård [Zhandry, Crypto '19]

Saturnin-Short: for small messages

- A single block m of < 128 bits
- (Actually it can be defined for 128 bits by reducing the nonce size)



Saturnin-CTR-Cascade: the main proposal

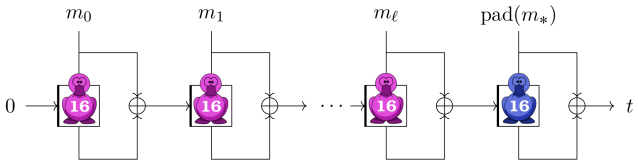


Under a qPRP assumption:

- CTR: IND-qCPA
- Cascade MAC: unforgeable

Saturnin-Hash: hash function proposal

We use a Merkle-Damgård construction with the MMO mode, and 16 Super-rounds.



- Classical birthday bound at $2^{256/2} = 2^{128}$
- **Quantum** birthday bound at $2^{256/3} = 2^{85.3}$
- Quantum collision algorithms are memory-intensive: we make a stronger (conjectural) security claim that depends on the adversary's quantum memory

Performance considerations

Hardware

Block cipher gate count:
118.5 gpb

- AES-256: 283.5
- Skinny-256: 156

Software

Saturnin-Cascade on an ARM Cortex M4:
144 cpb constant-time

- AES-GCM: 143 cpb [Adomnicai & Peyrin, 2020]

- Saturnin-Hash performs fairly well on Rhys Weatherley's microcontroller benchmarks*
- Saturnin-Short is very competitive for short messages

*https://rweather.github.io/lightweight-crypto/index.html

The Faturnin Challenge

We need to know more about the **related-key security** of the 16 Super-round version

- The key-schedule is simpler than the AES
- Classical reduced-round attacks?
- How about quantum attacks?

Saturnin-QCB

The **QCB** mode is a quantum-secure **rate-one** mode similar to ΘCB, based on a tweakable block cipher. We propose to use:

$$K, T, M \mapsto \text{Faturnin}_{K \oplus T}(M)$$

Bhaumik, Bonnetain, Chailloux, Leurent, Naya-Plasencia, Seurin, S., QCB: Efficient quantum-secure authenticated encryption

Conclusion

Post-quantum and lightweight

- We choose a block cipher of 256 bits (the only one in the LWC process)
- We choose well-known modes with quantum security guarantees
- Saturnin also offers a very high classical security

Further work

Although Faturnin (16 super rounds) is not used in the primary proposal, we need to know more about its **related-key security**.

Challenge opens soon!

Thank you!