ICEPOLE: High-speed, Hardware-oriented Authenticated Encryption Scheme

Paweł Morawiecki [1,2] Kris Gaj[4] Ekawat Homsirikamol[4]
Krystian Matusiewicz[7] Josef Pieprzyk[3] Marcin Rogawski[6]
Marian Srebrny[1,2] Marcin Wójcik[5]

Institute of Computer Science, Polish Academy of Sciences, Poland [1]
Section of Informatics, University of Commerce, Kielce, Poland [2]
Queensland University of Technology, Brisbane, Australia [3]
Cryptographic Engineering Research Group, George Mason University, USA [4]
Cryptography and Information Security Group, University of Bristol, United Kingdom [5]
Cadence Design Systems, San Jose, USA [6]
Intel, Gdańsk, Poland [7]

CHES 2014, Busan, South Korea

Outline

1. Motivation
2. Specification of ICEPOLE
3. ICEPOLE security and performance analysis
4. Conclusion

Non-authenticated Encryption

Alice got an encrypted message from Bob...

- Is it really from Bob?
- Has the ciphertext been modified?
- No mechanisms to answer these questions....

Authenticated Encryption (AE) Goals

Authenticated encryption scheme should fulfil two goals:

- confidentiality
- authenticity

Common AE Interface

INPUT:

- key
- plaintext
- associated data (optionally)
- nonce

OUTPUT:

- ciphertext
- authentication tag

Common AE Interface

INPUT:

- key
- plaintext
- associated data (optionally)
- nonce

OUTPUT:

- ciphertext
- authentication tag

Standards

- Encrypt-then-MAC (standardized in ISO/IEC 19772:2009)
- CCM (Counter with Cipher Block Chaining MAC)
- EAX (designed to replace CCM as the NIST standard)
- AES-GCM (arguably most common standard, point of reference in the new competition)
- others (OCB, CWC, ...)

Many Standards and Solutions but...

- Encrypt-then-MAC, EAX (two-pass)
- CCM (two-pass, message length has to be known before encryption starts)
- AES-GCM (polynomial multiplication very expensive in hardware, class of weak keys)
- What if nonce is reused? All security lost? Intermediate level?

CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness

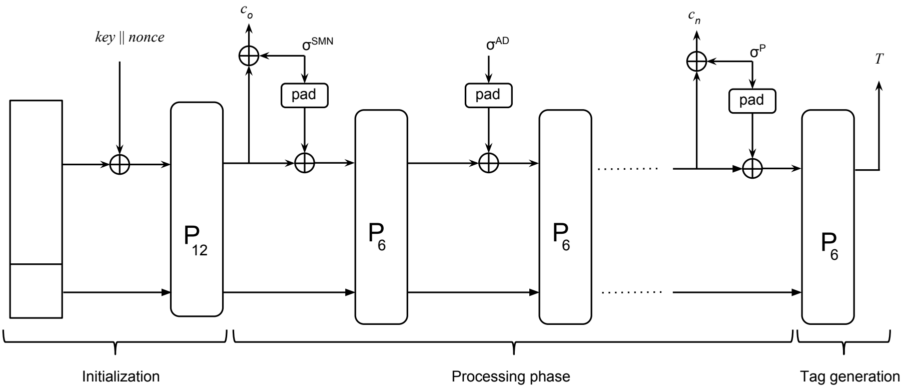
"*CAESAR will identify a portfolio of authenticated ciphers that (1) offer advantages over AES-GCM and (2) are suitable for widespread adoption.*"

- 2014.03.15 - end of 2017
- 1st round - 57 submissions
- http://competitions.cr.yp.to/caesar.html

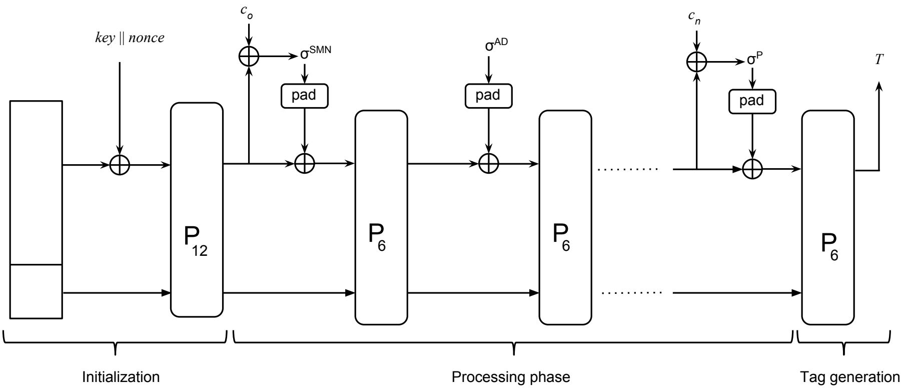
ICEPOLE General Overview

- based on the variant of duplex framework introduced by Bertoni et al. "Duplexing the sponge: (...)" Cryptology ePrint archive 2011/499
- high-speed hardware-oriented ICEPOLE permutation is the heart of our design
- family of authenticated encryption schemes with three parameters: key, nonce and secret message number
- primary recommendation: ICEPOLE-128: 128-bit key and 128-bit nonce

Encryption and Tag Generation - Overview



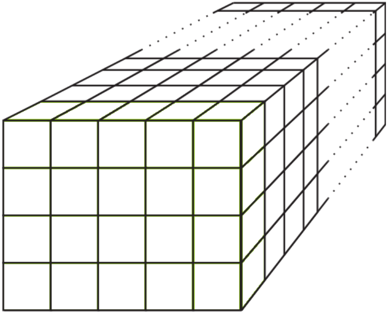
Decryption and Tag Verification



- The same permutations used for encryption and decryption

ICEPOLE Internal State Organization

- 1280-bit internal state S
- can be viewed as two-dimensional array $S[4][5]$, where each element of array is a 64-bit word



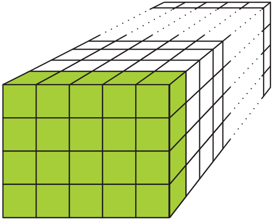
ICEPOLE Round and P_6, P_{12} Permutations

$$R = \kappa \circ \psi \circ \pi \circ \rho \circ \mu$$

ICEPOLE Permutations

- P_6: 6-round permutation, used in Processing Phase
- P_{12}: 12-round permutation, used only in Initialization

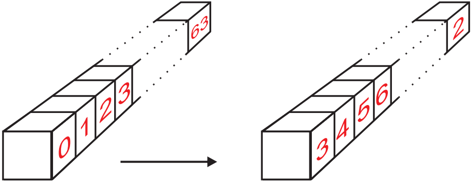
μ Step



$$\begin{pmatrix} 2 & 1 & 1 & 1 \\ 1 & 1 & 18 & 2 \\ 1 & 2 & 1 & 18 \\ 1 & 18 & 2 & 1 \end{pmatrix} \begin{pmatrix} Z_0 \\ Z_1 \\ Z_2 \\ Z_3 \end{pmatrix} = \begin{pmatrix} 2Z_0 + Z_1 + Z_2 + Z_3 \\ Z_0 + Z_1 + 18Z_2 + 2Z_3 \\ Z_0 + 2Z_1 + Z_2 + 18Z_3 \\ Z_0 + 18Z_1 + 2Z_2 + Z_3 \end{pmatrix}$$

- GF(2^5) multiplication modulo $x^5 + x^2 + 1$
- easy to implement (just XOR operations)
- main source of diffusion in the algorithm

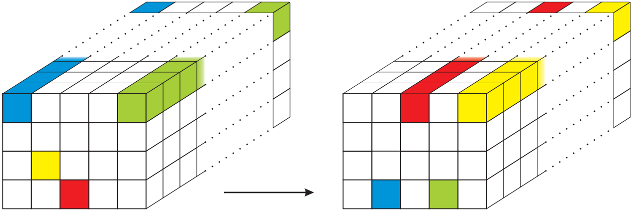
ρ Step



$S[x][y] := S[x][y] \lll \text{offsets}[x][y]$ for all $(0 \leq x \leq 3), (0 \leq y \leq 4)$

- each word has a distinct offset value
- ρ introduced to mix information between ‘slices’ of the state

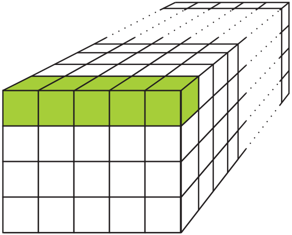
π Step



$$x' := (x + y) \; mod \; 4$$
$$y' := (((x + y) \; mod \; 4) + y + 1) \; mod \; 5$$

- $S[x^{'}][y^{'}] \leftarrow \pi(S[x][y])$
- π reorders the words in the state S
- introduced to provide more mixing between words

ψ Step



for all $(0 \leq k \leq 4)$
$$Z_k = M_k \oplus (\neg M_{k+1} M_{k+2}) \oplus (M_0 M_1 M_2 M_3 M_4) \oplus (\neg M_0 \neg M_1 \neg M_2 \neg M_3 \neg M_4)$$

ICEPOLE S-box

- The S-box maps a 5-bit input vector (M_0, ... M_4) to a 5-bit output vector (Z_0, ... Z_4)
- inspired by the Keccak S-box
- the only non-linear step in ICEPOLE

κ Step

$$S[0][0] := S[0][0] \oplus \mathrm{constant}[\mathrm{numberOfRound}]$$

Round Constants

- each round with a distinct constant
- introduced to break similarities between rounds
- The constants are calculated as the output of a simple 64-bit maximum-cycle Linear Feedback Shift Register (LFSR).

ICEPOLE Security (Parameters)

- ICEPOLE is based on the duplex construction, parameters: r (bitrate) and c (capacity)
- ICEPOLE-128: 128-bit security level ($r = 1026$ bits and $c = 256$ bits)
- ICEPOLE-256: 256-bit security level ($r = 962$ bits and $c = 318$ bits)
- If the underlying permuations are secure, ICEPOLE is secure (security reduction inherited from the duplex construction)

Nonce Requirement

- ICEPOLE requires a nonce
- In case of nonce reuse, some level of intermediate robustness provided by secret message number and associated data (if distinct)
- In case of violating **all** nonce-like mechanisms (nonce reused, secret message number reused, the same associated data), security claims do not hold (recent analysis by Tao Huang, Hongjun Wu, Ivan Tjuawinata)

ICEPOLE Security Analysis

- **Differential cryptanalysis** (with aid of a SAT solver, we provide a bound on differential trail probability — for 12 rounds, probability $\leqslant 2^{-84}$)
- **Linear cryptanalysis** (good linear profile of s-box, propagation of linear masks very similar to differential analysis, expecting similar security margin. Rigorous analysis to be done)
- **Rotational cryptanalysis** (good selection of round constants and pseudo-random initial state prevent this kind of attack)
- **SAT-based cryptanalysis** (experimentally verified, the attack reaches only 3 rounds)
- **Techniques exploiting low algebraic degree** (algebraic degree of a single round is 4, then for 4 rounds a degree is 256, making the attacks infeasible)

FPGA Implementation Results

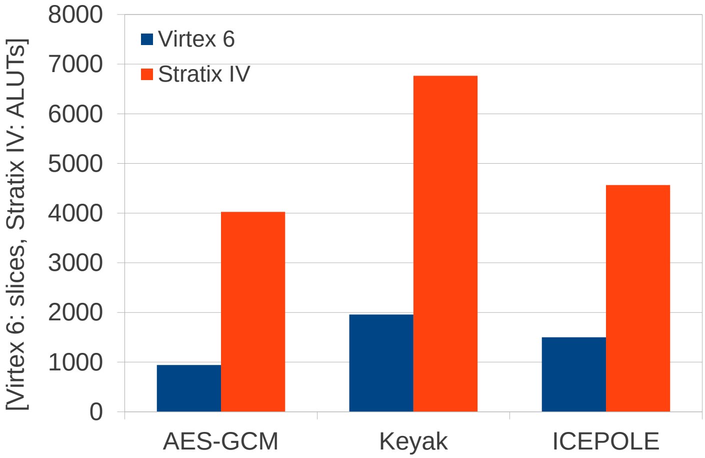
Xilinx Virtex-6

- Throughput: 41364 Mbps
- Area: 1501 Slices
- Throughput/Area: 27.56 Mbps/Slice

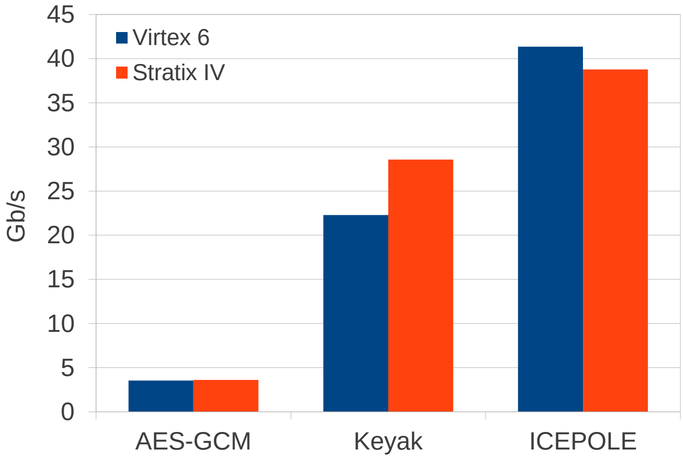
Altera Stratix-IV

- Throughput: 38779 Mbps
- Area: 4564 ALUTs
- Throughput/Area: 8.50 Mbps/ALUT

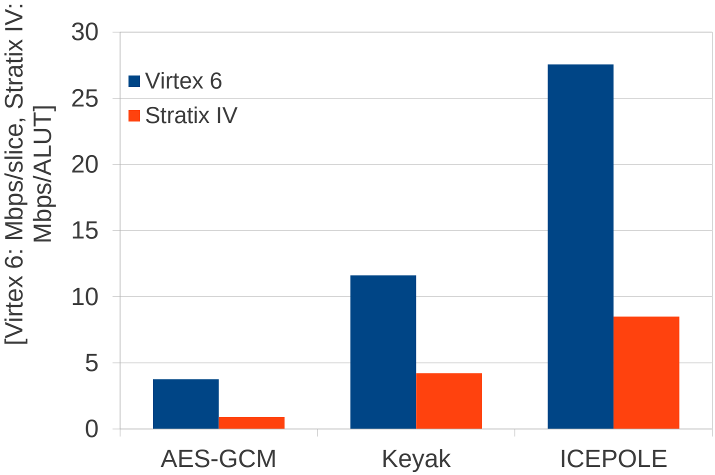
FPGA Implementation - Area



FPGA Implementation - Throughput



FPGA Implementation - Throughput/Area



Software Implementation

- straightforward C implementation compiled for speed (no beyond-C optmization used)
- 9 cycles per byte on Intel Ivy Bridge (i5-3320M)
- 8 cycles per byte on Haswell (Intel Xeon E3 1275)

Conclusion

- monkeyDuplex construction + very efficient permutation = ICEPOLE
- highly efficient in modern FPGAs
- very-high speed in modern FPGAs
- good software performance

Questions

Thank you!

Questions?

Questions?