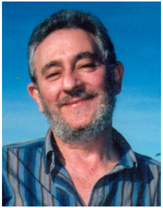
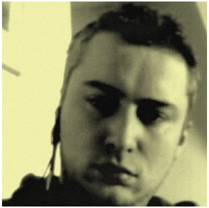
CAESAR candidate ICEPOLE

Pawel Morawiecki[1,2], Kris Gaj[3], Ekawat Homsirikamol[3], Krystian Matusiewicz[4], Josef Pieprzyk[5,6], **Marcin Rogawski**[7], Marian Srebrny[1,2], and Marcin Wojcik[8]

Polish Academy of Sciences, Poland[1]; University of Commerce, Poland[2]; George Mason University, USA[3]; Intel, Gdansk, Poland[4]; Queensland University of Technology, Australia[5]; Macquarie University, Australia[6]; Cadence Design Systems, USA[7]; University of Bristol, United Kingdom[8]

DIAC 2014: Directions in Authenticated Ciphers

Co-authors

Outline

1. Introduction and Motivation
2. Icepole Design
3. Security Analysis
4. HW and SW Performance
5. Summary

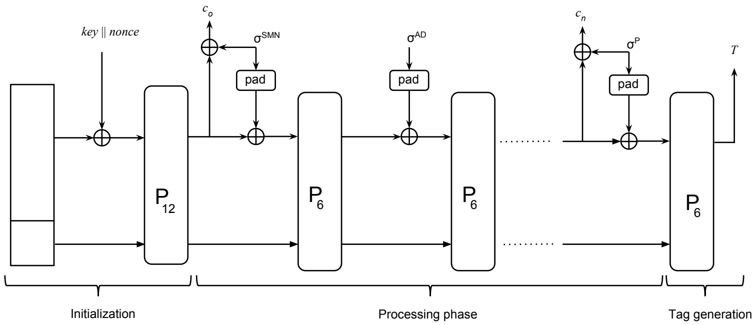
Introduction and Motivation

- Multiple Internet protocols require authenticated encryption: IPSec/TLS/SSL etc.
- High-speed hardware-oriented cipher with authentication, more efficient that AES-GCM
- Existing frameworks/strategies for provably secure cryptographic schemes (e.g.: Sponge Construction etc.)
- CAESAR competition

ICEPOLE 101

- based on 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 SMN
- primary recommendation: ICEPOLE-128: 128-bit key and 128-bit nonce

Encryption and Tag Generation - Overview



ICEPOLE Internal State Organization

- 1280-bit internal state S
- organized into dwo-dimensional array $S[4][5]$
- each element of array is a 64-bit word
- $S[x][y][z]$ refers to the bit z in the row x and the column y
- the mapping between a vector V and the S:
  $V[64(x+4y)+z] = S[x][y][z]$

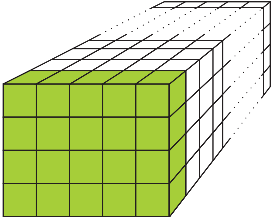
ICEPOLE Round and P6, P12 Permutations

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

ICEPOLE Permutations

- P6 - 6 rounds of ICEPOLE permutation
- P12 - 12 rounds of ICEPOLE permutation

Transformation: μ



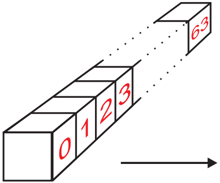
$$\begin{pmatrix} 2 & 1 & 1 & 1 \\ 1 & 1 & 18 & 2 \\ 1 & 2 & 1 & 18 \\ 1 & 18 & 2 & 1 \end{pmatrix} \begin{pmatrix} Z_0 \\ Z_1 \\ Z_2 \\ Z_3 \end{pmatrix} = \begin{pmatrix} 2Z_0 + Z_1 + Z_2 + Z_3 \\ Z_0 + Z_1 + 18Z_2 + 2Z_3 \\ Z_0 + 2Z_1 + Z_2 + 18Z_3 \\ Z_0 + 18Z_1 + 2Z_2 + Z_3 \end{pmatrix}$$

- GF(2^5) multiplication modulo $x^5 + x^2 + 1$

ICEPOLE Round

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

Transformation: ρ



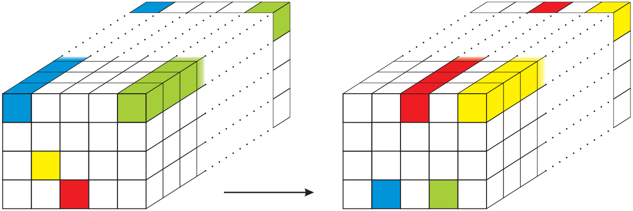
$$S[x][y] := S[x][y] \lll \text{offsets}[x][y] \qquad \text{for all } (0 \le x \le 3), (0 \le y \le 4)$$

$$\begin{array}{llll}
\text{offsets}[0][0] := 0 & \text{offsets}[0][1] := 36 & \text{offsets}[0][2] := 3 & \text{offsets}[0][3] := 41 \\
\text{offsets}[0][4] := 18 & \text{offsets}[1][0] := 1 & \text{offsets}[1][1] := 44 & \text{offsets}[1][2] := 10 \\
\text{offsets}[1][3] := 45 & \text{offsets}[1][4] := 2 & \text{offsets}[2][0] := 62 & \text{offsets}[2][1] := 6 \\
\text{offsets}[2][2] := 43 & \text{offsets}[2][3] := 15 & \text{offsets}[2][4] := 61 & \text{offsets}[3][0] := 28 \\
\text{offsets}[3][1] := 55 & \text{offsets}[3][2] := 25 & \text{offsets}[3][3] := 21 & \text{offsets}[3][4] := 56
\end{array}$$

ICEPOLE Round

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

Transformation: π



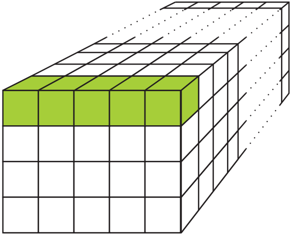
$$x' := (x + y)\ mod\ 4$$
$$y' := (((x + y)\ mod\ 4) + y + 1)\ mod\ 5$$

- π reorders the words in the state S
- $S[x^{'}][y^{'}] \leftarrow \pi(S[x][y])$

ICEPOLE Round

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

Transformation ψ



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 $(M_0, \ldots\ M_4)$ to a 5-bit output vector $(Z_0, \ldots\ Z_4)$

ICEPOLE Round

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

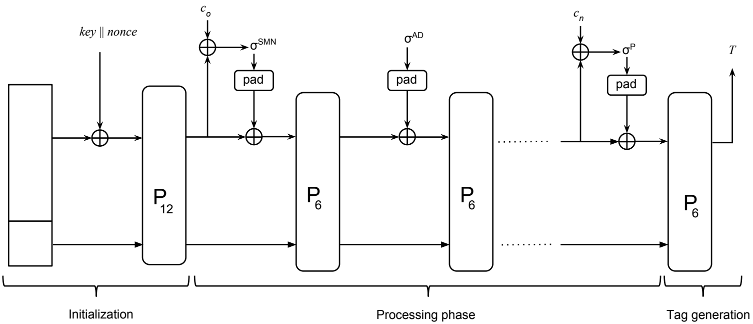
Transformation: κ

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

ICEPOLE Constants

- The constant values are taken as the output of a simple 64-bit maximum-cycle Linear Feedback Shift Register (LFSR).
- The polynomial representation of LFSR is $x^{64} + x^{63} + x^{61} + x^{60} + 1$.
- The LFSR seed 0123456789ABCDEF
- each cycle generates a subsequent constant.

Decryption and Tag Generation



ICEPOLE Security (Parameters)

- ICEPOLE is based on the duplex construction - parameters: r (bitrate) and c (capacity)
- ICEPOLE-128: r=1026 bits and c=256 bits (up to 2^{126} blocks)
- ICEPOLE-256: r=962 bits and c=318 bits (up to 2^{62} blocks)
- Security level proven, unless permuation is unsecure

SKEW'11: Bertoni et al. in "On the security of the keyed sponge construction" proved that if the data complexity is limited to 2^a r-bit blocks, the keyed mode withstands generic attacks with time complexity up to 2^{c-a} calls of the underlying permutation. If $a < c/2$, this results in an increase of the security strength from $c/2$ to $c - a$.

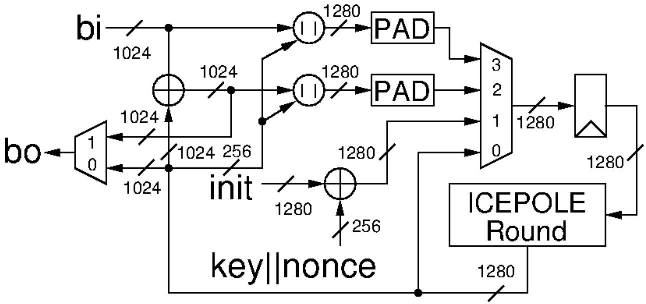
Nonce Requirement

- 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)

Basic Iterative Architecture



Source:

Morawiecki et al. "ICEPOLE: High-speed, Hardware-oriented Authenticated Encryption" at CHES'14

FPGA Implementation Results

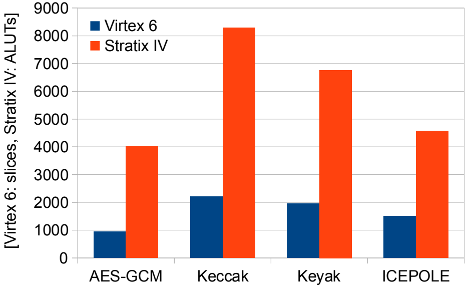
Xilinx Virtex-6

- 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

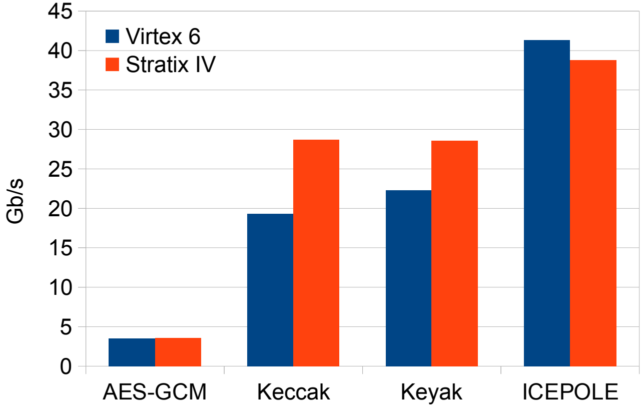
FPGA Implementation - Area



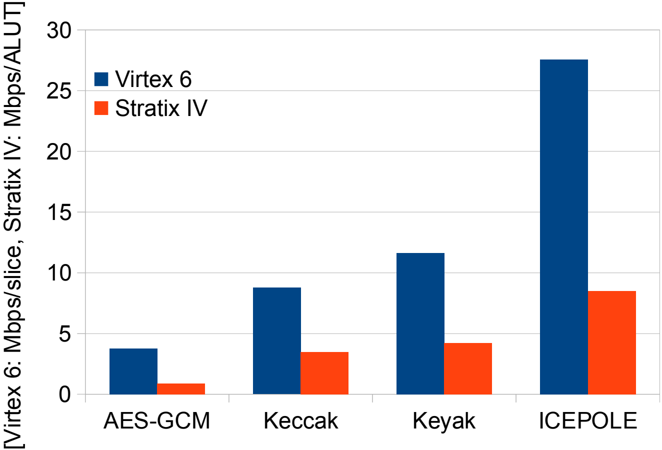
Source:

- Keyak and Keccak (multi-purpose mode) from anonymous submission to anonymous conference :)
  Thanks for sharing!

FPGA Implementation - Throughput



FPGA Implementation - Throughput/Area



Software Implementation

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

Conclusions

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

Questions

Thank you!

Questions?

Questions?