NOEKEON, The Return

Guido BERTONI[1] Joan DAEMEN[1] Michaël PEETERS[2]
Vincent RIJMEN[3] Gilles VAN ASSCHE[1]

[1]STMicroelectronics

[2]NXP

[3]COSIC and IAIK

January 11, 2010 — ESC Seminar, Remich

NOEKEON

- Block cipher
  - 128-bit blocks
  - 128-bit keys
  - Bit-slice cipher, similar to Serpent
  - Descendent of 3-Way and BaseKing
- Submitted to Nessie in 2000
- Not selected due to related-key distinguishers
- Why dig up again? Unique combination of advantages:
  - Lower bounds on trail weights
  - Lightweight: hardened implementations at low cost
  - Simplicity: interesting object for (crypt)analysis
- See http://gro.noekeon.org/

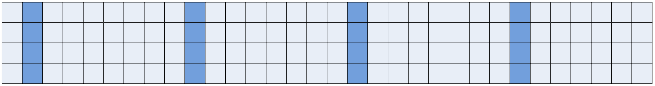
NOEKEON design criteria

- Security
  - Resistance against known types of cryptanalysis
  - Suitability for building hardened implementations
    - (differential) power analysis, electromagnetic analysis,
    - timing attacks, . . .
- Efficiency
  - Both speed-optimized and hardened implementations
  - Software: wide range of platforms
  - Hardware: compact and fast

NOEKEON architecture: maximize symmetry

- Operations:
  - Bit-wise Boolean operations
  - Cyclic shifts
- 16 equal rounds
- 17 equal round keys
- Inverse cipher equal to cipher itself
- Asymmetry provided by round constants only

The NOEKEON state



- Two-dimensional $4 \times \ell$ array
  - 4 *words*
  - ℓ *boxes*
- Additional partitioning of the state: *columns*
  - $\ell/4$ columns
- $\ell = 32$

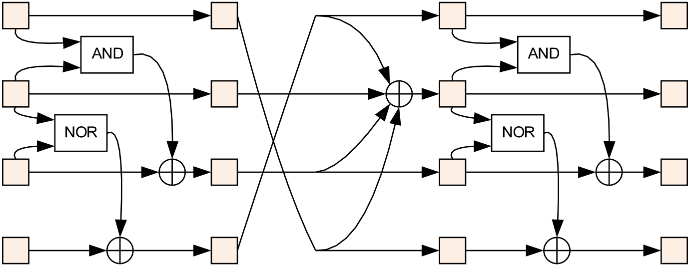
Round transformation

- γ: nonlinear step
  - 4-bit S-box operating on boxes
  - Involution
- θ: combines mixing layer and round key addition
  - Linear 16-bit mixing layer operating on columns
  - Involution
- π: dispersion between columns
  - Rotation of bits within ℓ-bit words
  - Two instances that are each others inverse
- ι: round constant addition for asymmetry

The round and its inverse

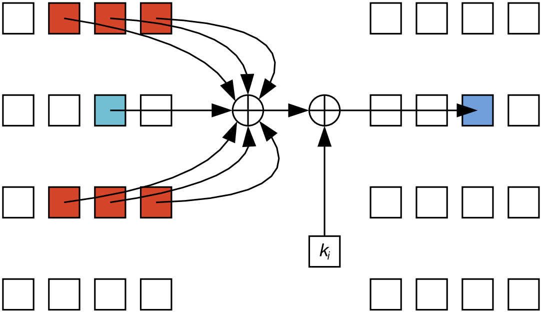
- Round: $\pi_2 \circ \gamma \circ \pi_1 \circ \theta[k]$
- Inverse round:
  - $\theta[k]^{-1} \circ \pi_1^{-1} \circ \gamma^{-1} \circ \pi_2^{-1}$
  - $\theta[k] \circ \pi_2 \circ \gamma \circ \pi_1$
- $\theta[k]$ as final transformation:
  - Regrouping: round of inverse cipher $=$ cipher round
  - round constants prevent involution
- NOEKEON: 16 rounds and a final transformation

γ



- Two identical nonlinear steps with a linear step in between
- Simple algebraic expression

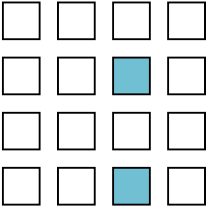
θ



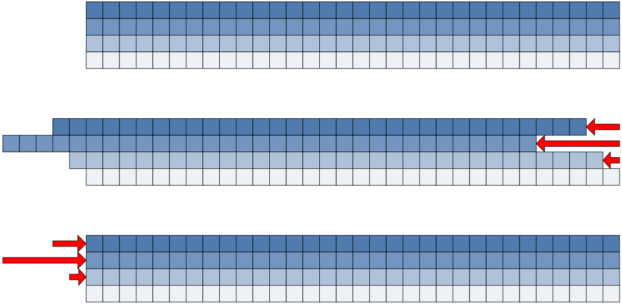
- High average diffusion
- Small number of operations thanks to symmetry

θ cont'd

- Branch number $\mathcal{B}$ only 4 due to symmetry
- Invariant sparse states, e.g.:



π



- π_1 and π_2 are each others inverses

Differential and linear cryptanalysis

- Bounds on 4-round trails (for block length 128)
  - Differential trails: $EDP \leq 2^{-48}$
  - Linear trails: $ELP \leq 2^{-48}$
- 12 rounds: no trails with ELP/EDP above 2^{-144}
- Powerful bounds thanks to
  - High average diffusion in θ and π
  - Invariant sparse states addressed in γ S-box
- Determining bounds:
  - Non-trivial exercise
  - See http://gro.noekeon.org/Noekeon-spec.pdf

Other aspects

- Non-aligned structure:
  - Truncated DC: no clustering of trails along (byte) boundaries
- Lightweight rounds
  - More rounds are required for achieving similar bounds
- Square attacks AKA integral cryptanalysis
  - [Z'aba, Raddum, Henricksen, Dawson, FSE 2008]
  - Best attack: 5 rounds
- Algebraic cryptanalysis
  - interesting subject thanks to simple algebraic equations
  - Vulnerable? To be seen . . .
- Symmetry: round constants
  - Protect against slide attacks
  - Prevent symmetric properties

Efficiency

- Cipher and inverse are equal: re-use of circuit and code
- Hardware: compact and fast
  - number of gates: 1050 XOR, 64 AND, 64 NOR, 128 MUX
  - Gate delay: 7 XOR, 1 AND, 1 MUX
  - Coprocessor architecture: speed/area trade-off
- Software: ideal for embedded, e.g. ARM7:
  - Code size 332 bytes, 44.5 cycles/byte
  - Code size 3688 bytes, 30 cycles/byte
  - RAM usage: everything in registers

Hardened implementations

- Timing: fixed sequence of operations and no table-lookups
- Differential power/electromagnetic analysis: state splitting
  - Solid protection against 1st order DPA
    - Thanks to very limited non-linearity
    - Provably secure based on weak assumptions
  - Software
    - Two shares [Daemen, Peeters, Van Assche, FSE 2000]
    - Roughly doubles execution time, state and code size
  - Hardware (in presence of glitches)
    - Three shares [Nikova, Rijmen, Schläffer, ICISC 2008]
    - number of gates $\times 4$ and slight increase in gate delay

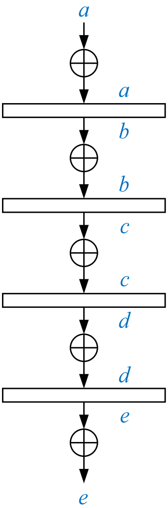
Extension: block length 64

- Addressing low footprint
- Take $\ell = 16$
- Data path:
  - θ and γ: not impacted by the value of ℓ
  - π_1 and π_2: keep same shift offsets
  - ι: new round constants
- Computation of rounds keys from 128-bit working key:
  - Odd-indexed round keys: first part of working key
  - Even-indexed round keys: second part of working key

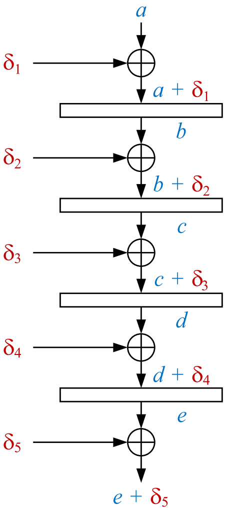
Extension: addition of hermetic key mode

- Originally two modes:
  - Direct: round key = cipher key
  - Indirect: round key = NOEKEON[0](cipher key)
- Related-key distinguisher for indirect mode
  - Non-ideal behaviour
  - [Knudsen, Raddum, NESSIE 2001]
- Addition of hermetic mode:
  - Goal: ideal cipher

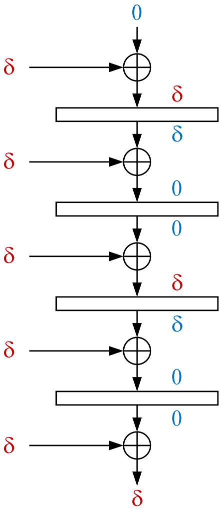
Related-key differential trails



Related-key differential trails

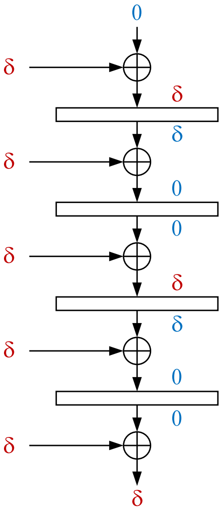


Related-key trails in direct mode of NOEKEON



$K^* = K + \delta$

Related-key trails in indirect mode of NOEKEON



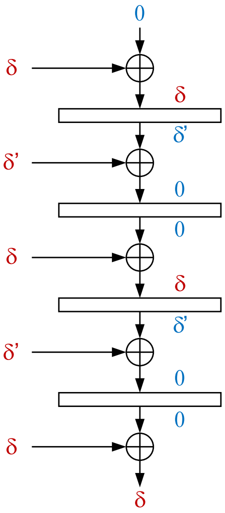
$$K^* = F^{-1}(\,F(\,K\,) + \delta\,)$$

First attempt at the hermetic key schedule

- Working key containing two round keys
  - $k_{2i} = K$
  - $k_{2i+1} = F(K)$ with $F(x) = \text{NOEKEON}[0](x)$
- Now:
  - Simple relation in k_{2i} gives complicated relation in k_{2i+1}
  - Simple relation in k_{2i+1} gives complicated relation in k_{2i}
- But:
  - There exist weak key pairs with overwhelming probability

  ...

Related-key distinguishers for the first attempt



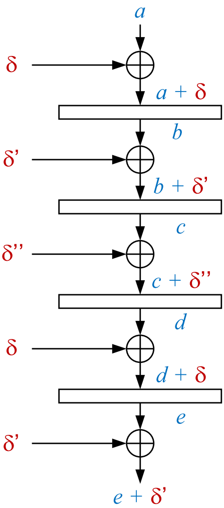
$K + K^* = \delta$

$F(K) + F(K^*) = \delta'$

Second attempt

- Working key containing three round keys
  - $k_{3i} = K$
  - $k_{3i+1} = F(K)$
  - $k_{3i+2} = F(F(K))$
- ...plausible that this is sufficient for our goal

Related-key trail picture



$$K + K^* = \delta$$

$$F(K) + F(K^*) = \delta'$$

$$F(F(K)) + F(F(K^*)) = \delta''$$

Related-key distinguishers: some quantitative arguments

- Numbers:
  - 2^{3n} difference patterns $(\delta, \delta', \delta'')$ exist
  - 2^{2n-1} of them actually occur: one for each K, K^*
  - Say 2^z of all patterns $(\delta, \delta', \delta'')$ are threatening
  - Expected number of threatening pairs K, K^*: $2^{z-(n+1)}$
- Three cases are possible:
  - $0 < z < n$: non-existence
    - Probability no threatening pairs exist: 2^{z-n-1}
  - $n \leq z < 2n$: hard to exploit
    - Expected number of threatening pairs: 2^{z-n-1}
    - Expected number of pairs K, K^* to try: $2^{3n-z} > 2^n$
  - $2n \leq z < 3n$: insufficient protection
    - Expected number of pairs K, K^* to try: $2^{3n-z} < 2^n$

Convenience of the hermetic key schedule

- Re-use of data path: software, hardware and hardened
- Implementation cost: 2 calls to NOEKEON
- Expanded key is three times as long as the cipher key
  - Can be pre-computed and stored

Key mode summary

- Direct mode
  - Goal: secure if no related-key attacks can be mounted
  - Covers all use cases with sound key management
  - Should offer *Pseudorandom Permutation (PRP)* security
- Indirect mode
  - Goal: secure against practical related-key attacks
  - Covers also use cases with lousy key management
- Hermetic mode
  - Goal: absence of structural distinguishers
  - Not inspired by practical use cases, of philosophical interest
  - Should offer *Ideal Cipher* security

Questions?

Thanks for your attention!

Any questions?