Cryptographic Schemes Based on the ASASA Structure: Black-box, White-box, and Public-key

A. Biryukov, C. Bouillaguet, D. Khovratovich,

(talk given by Ivica Nikolic)

University of Luxembourg and University of Lille

8 December 2014

White-box cryptography (WBC)

Motivation for the ASASA construction in public- and secret-key schemes

Weak white-box

WBC centers around *white-box implementation*:

1. Pure software implementation of a cipher (encryption or decryption routine) with *embedded key*;
2. Implementation is assumed available to an adversary.
3. **Weak white-box**: Adversary can not extract the *key* from the implementation.

Weak white-box

WBC centers around *white-box implementation*:

1. Pure software implementation of a cipher (encryption or decryption routine) with *embedded key*;
2. Implementation is assumed available to an adversary.
3. **Weak white-box**: Adversary can not extract the *key* from the implementation.

Example: recovering protected media content, which is decoded in software:

Strong white-box

WBC centers around *white-box implementation*:

1. Pure software implementation of a cipher (encryption or decryption routine) with *embedded key*;
2. Implementation is assumed available to an adversary.
3. **Strong white-box**: adversary can not *invert* the cipher, i.e. can not decrypt given the encryption routine.

Similar to public-key cryptography. Why not using it?

Strong white-box

WBC centers around *white-box implementation*:

1. Pure software implementation of a cipher (encryption or decryption routine) with *embedded key*;
2. Implementation is assumed available to an adversary.
3. **Strong white-box**: adversary can not *invert* the cipher, i.e. can not decrypt given the encryption routine.

Similar to public-key cryptography. Why not using it?

- RSA-2048 encryption speed — 1000 cycles per byte.
- AES-128 encryption speed — 0.7 cycles per byte.

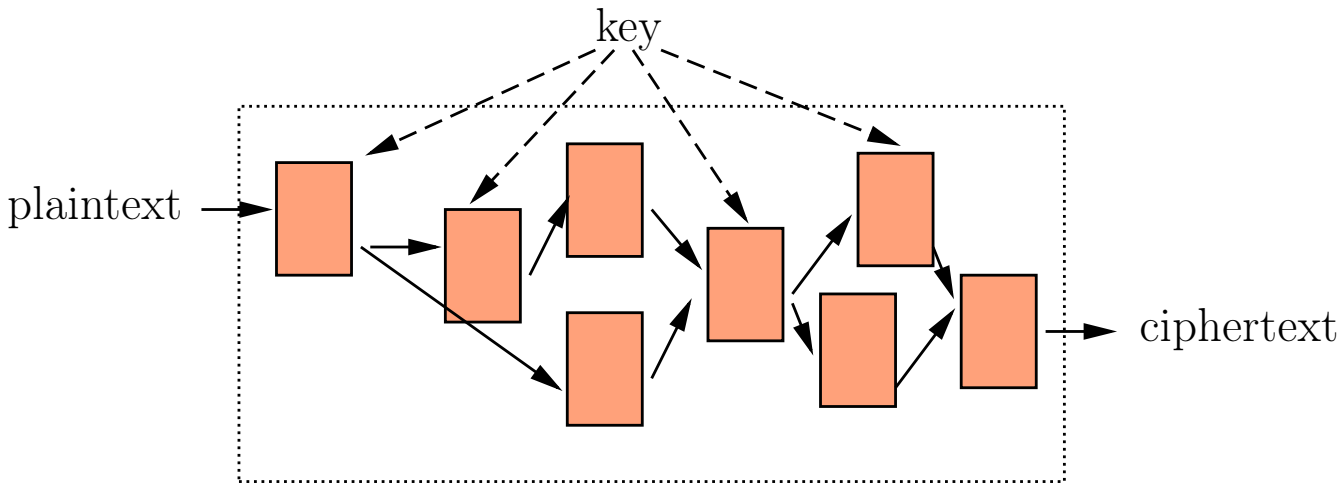
Impractical for large amount of data (HD movies, etc.).

Failure to build weak white-box with AES

Generic approach to white-boxing

How to white-box a cipher:

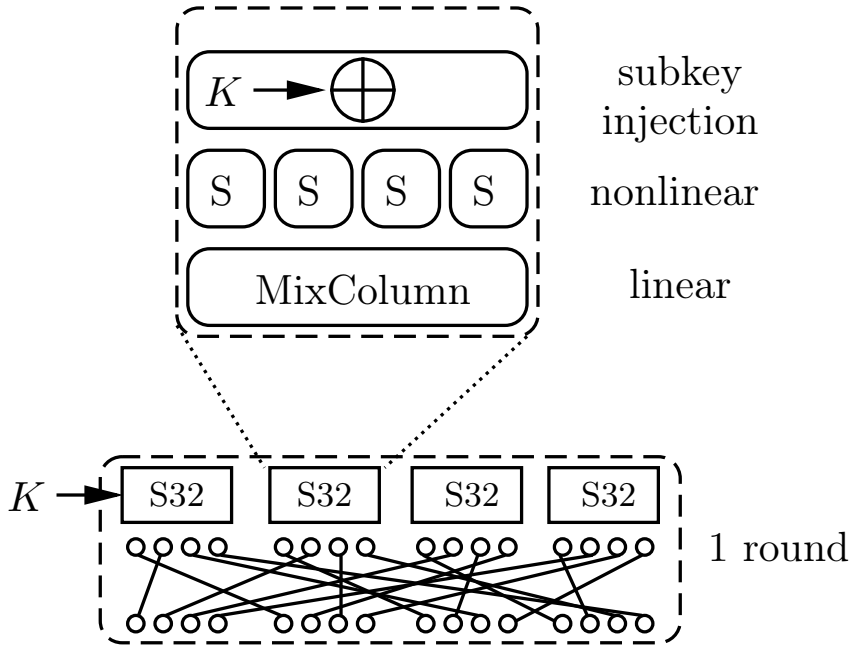
- Replace key-dependent transformations with lookup tables;
- Encode the encryption as a sequence of lookups.



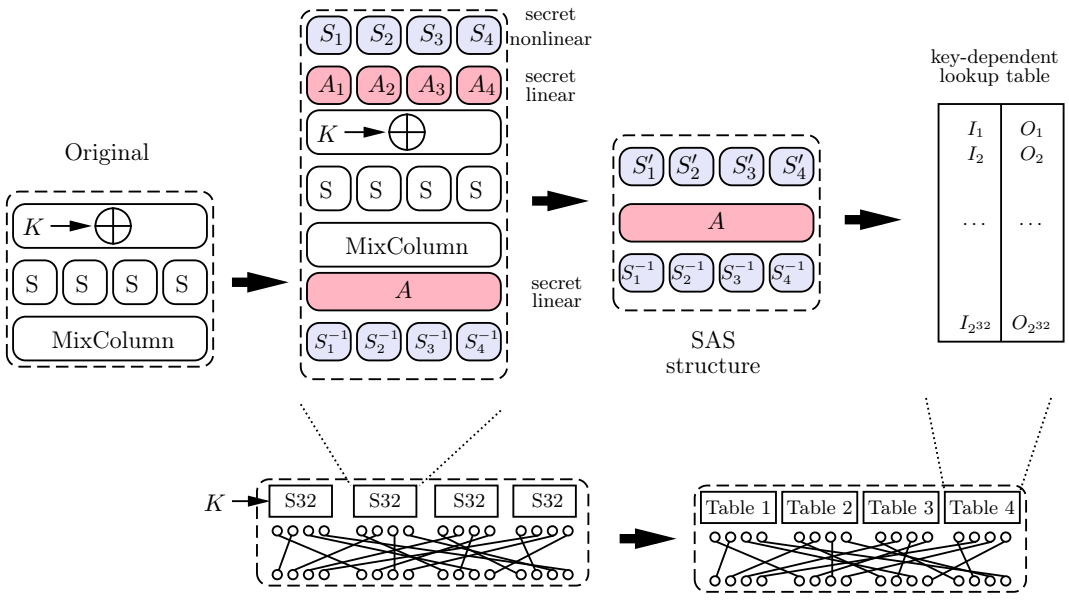
AES-128 (designed in 1997, adopted in 2001): 10-round cipher with 16-byte state.

One round of AES:

- Four 32-bit blocks:
  - AddRoundKey (simple XOR);
  - SubBytes (bytewise nonlinear);
  - MixColumns (linear).
- ShiftRows (byte permutation).



White-boxing AES round

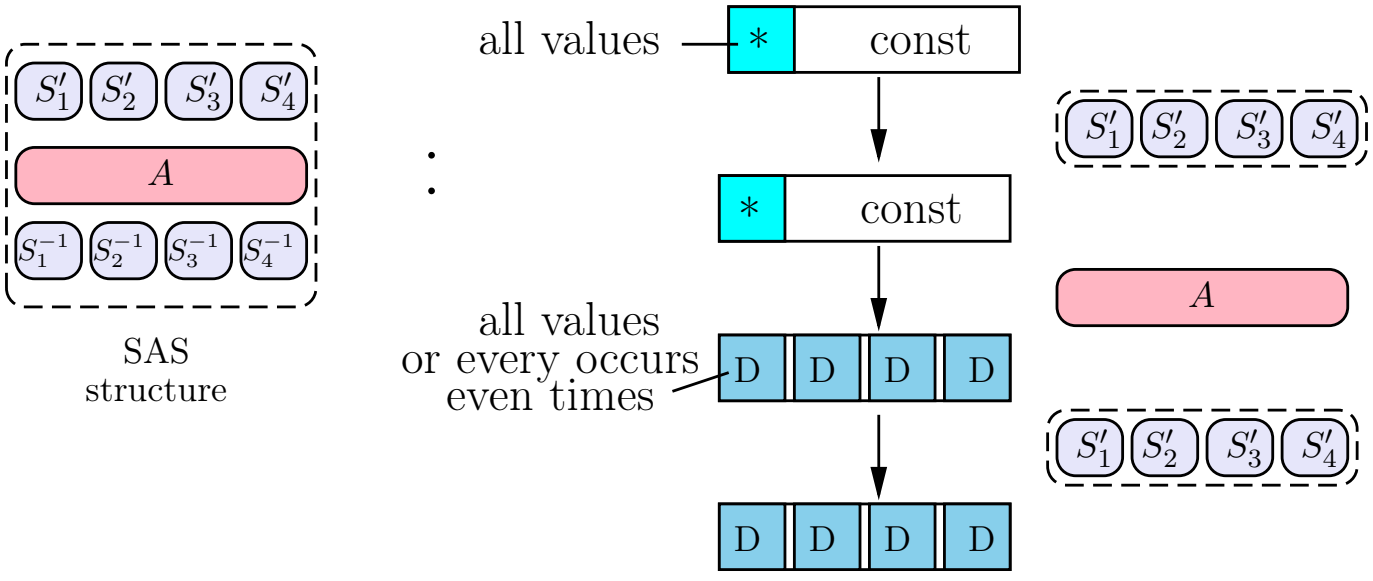


- Wrap the key addition and S-boxes with redundant linear and nonlinear transformations;
- The secret layers collapse to the SAS structure.
- Replace every 32-bit block with a *lookup table*;
- Store everything in memory.

Actual proposal used smaller and weaker tables.

Attacks

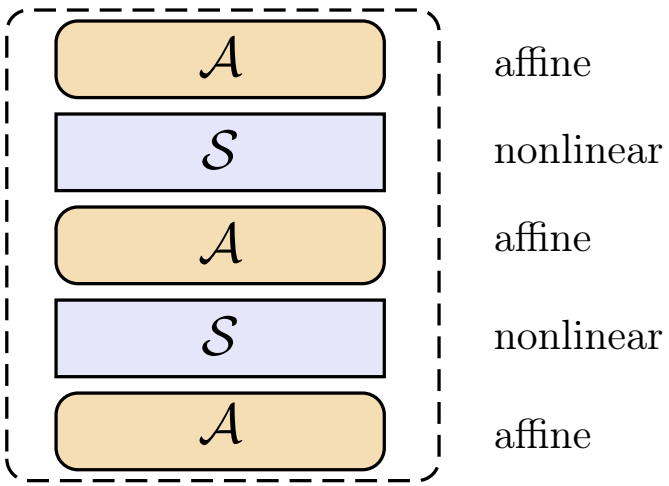
The SAS structure has some exploitable properties:



- For instance, set $(*, C, C, C)$ transforms to (D, D, D, D).
- This led way to various sorts of attacks, including differential and algebraic ones.
- Outer layers can be retrieved.

Constructions as large as SASAS are vulnerable [Biryukov-Shamir'01].

However, the ASASA structure is still unbroken:

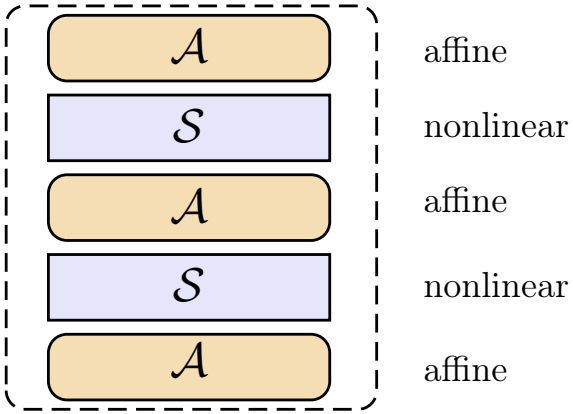


Hints for new designs...

Our contributions

Secret-key ASASA

Symmetric ASASA with secret layers.



- Key-dependent affine layers (random invertible matrices);
- Key-dependent secret 8-bit S-boxes (random permutations);
- We estimate 128-bit security for 128-bit keys and 128-bit blocks;
- Some attacks on other parameters are presented in the paper.

White-box ASASA

Weak white-box security

Weak white-box: adversary can not extract the *key* from the implementation.

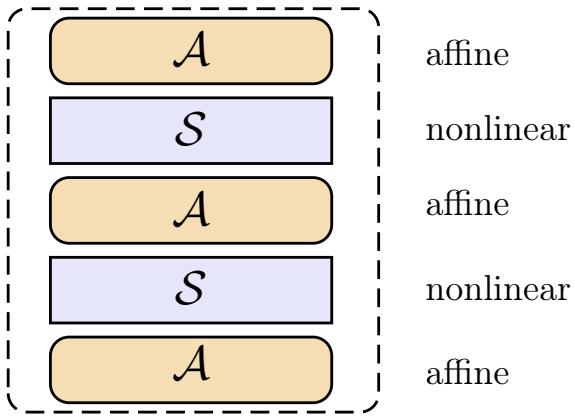
We extend the definition: it should be infeasible to derive a key or *any other compact secret* from the WB implementation.

Adversary can not recover the components $\Longrightarrow$ he has to use/broadcast the implementation "as is".

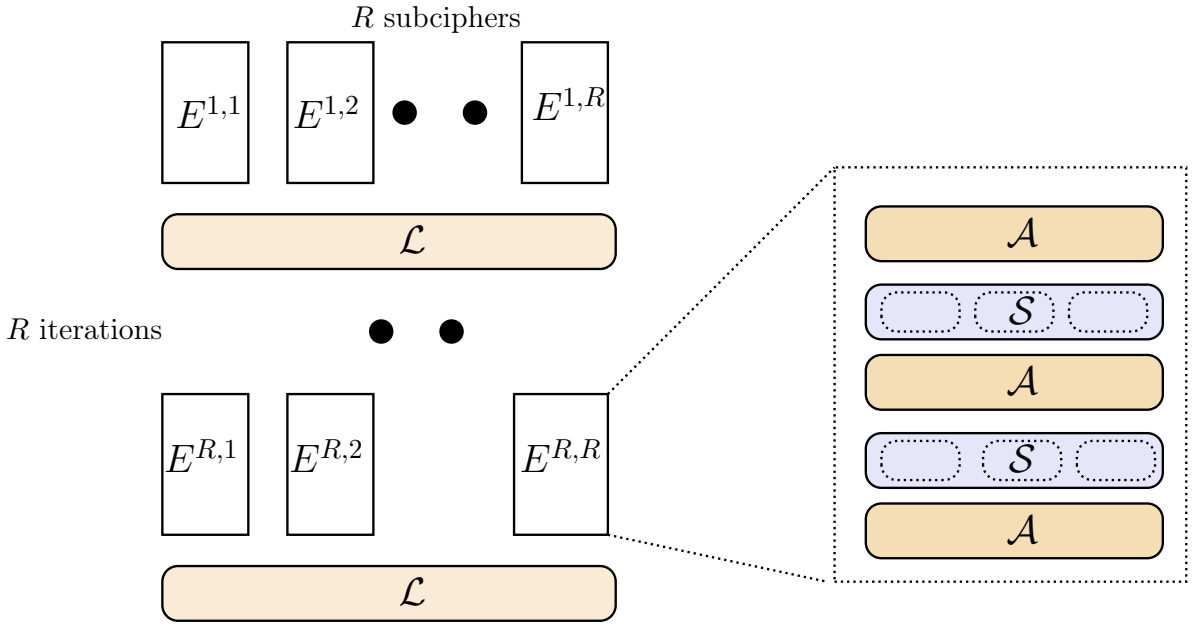
Weak white-box security

Our solution for weak white-box security: an implementation that *can not be compressed*.

Consider small block ($\leq$32 bits) ASASA cipher and encode it as a lookup table.



Incompressible implementation



- Cipher composed of smaller d-bit subciphers ($8 \leq d \leq 28$).
- Parameter d determines the implementation size.
- Subcipher invocations alternate with public permutations ($\mathcal{L}$).
- Total implementation size can be tuned from 2 MB to 20 GB.

Public-key and strong white-box ASASA

Public-key cryptography with polynomials (dates back to 1980s):

$$\mathbf{b} = A_2 \circ \mathbf{S} \circ A_1, \tag{1}$$

where A_1 and A_2 are key-dependent and secret affine transformations, and $\mathbf{S}$ is a public invertible polynomial of degree 2.

- Degree-2 polynomials of 128 boolean variables are compact enough (less than 1 MByte), and there is no generic inversion algorithm.

Public-key cryptography with polynomials (dates back to 1980s):

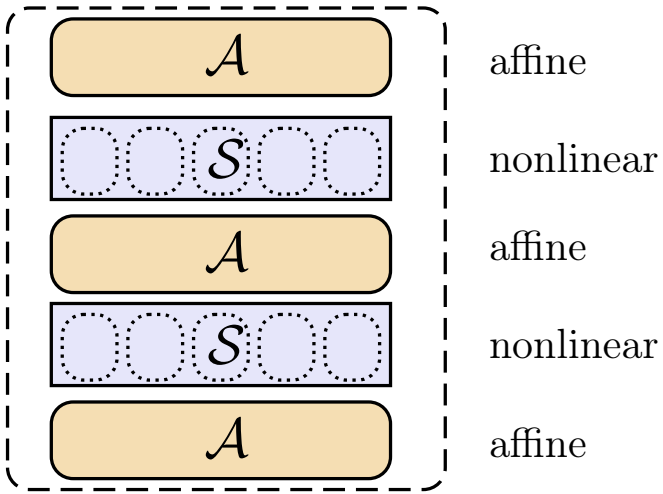
$$\mathbf{b} = A_2 \circ \mathbf{S} \circ A_1, \tag{1}$$

where A_1 and A_2 are key-dependent and secret affine transformations, and $\mathbf{S}$ is a public invertible polynomial of degree 2.

- Degree-2 polynomials of 128 boolean variables are compact enough (less than 1 MByte), and there is no generic inversion algorithm.
- However, virtually all variants of this scheme have been broken because of properties of $\mathbf{S}$: only a few families of invertible polynomials are available (even without trapdoors).

How about adding more layers to get ASASA?

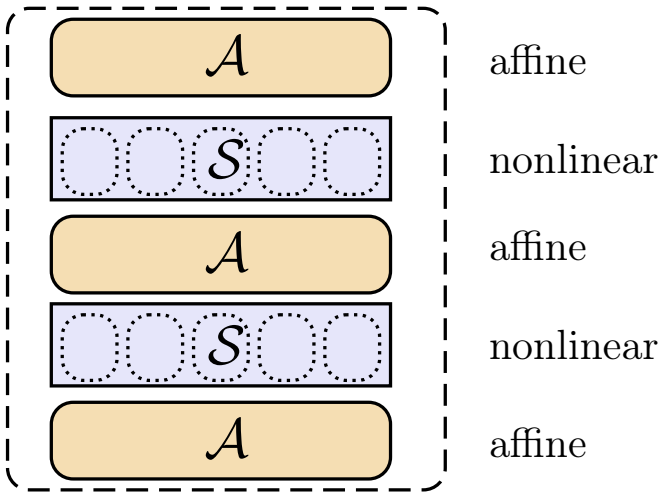
Polynomial-based S-boxes of degree 2?



The encryption function is a set of degree-4 polynomials.

Asymmetric scheme

First attempt: use Daemen's nonlinear function ($y_i = x_i \oplus x_{i+1}x_{i+2} \oplus x_{i+2}$), used in Keccak/SHA-3.



Problems:

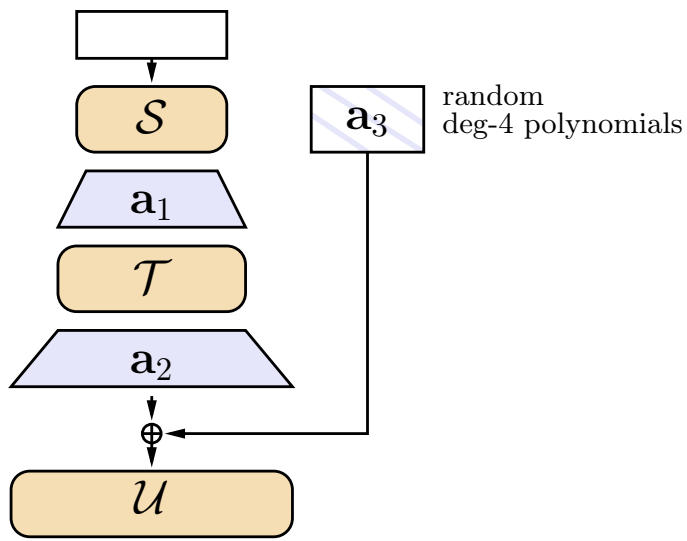
- Broken with Grobner basis attacks in real time;
- Also vulnerable to generic decomposition algorithms [Faugere-Perret'10];

Our solution: random S-boxes and noise

Our solution: ASASA with expanding S-boxes and perturbation (noise):

$$\mathbf{b} = U \circ \mathbf{a}_2 \circ T \circ \mathbf{a}_1 \circ S$$

- Two nonlinear layers ($128 \to 256$ and $256 \to 512$ bits);
- Nonlinear transformations are expanding and more random-looking;
- Perturbation ($\mathbf{a}_3$) added to defeat generic decomposition algorithms;
- 24 MBytes of public key.



Also in the paper

Additional material in the proceedings and on ePrint:

- LPN and algebraic attacks on weakened multivariate schemes;
- Various attacks on secret-key ASASA.

Questions?