The LOCAL attack: Cryptanalysis of the authenticated encryption scheme ALE

Dmitry Khovratovich and Christian Rechberger

University of Luxembourg and DTU (Denmark)

Presented by Yu Sasaki (NTT, Japan)

15 August 2013

Authenticated encryption

Authenticated encryption

Authenticated encryption — a single-key construction that achieves both confidentiality and data integrity.

Data integrity/authentication means that a decryptable ciphertext must have been produced with a secret key. Hence most ciphertexts must decrypt to $\perp$.

Authenticated encryption

Authenticated encryption — a single-key construction that achieves both confidentiality and data integrity.

Data integrity/authentication means that a decryptable ciphertext must have been produced with a secret key. Hence most ciphertexts must decrypt to $\perp$.

Several types:

- Modes of operation (OCB, EAX, CCM, GCM), which invoke an arbitrary blockcipher;
- Dedicated constructions (Helix/Phelix, Grain128a), which use fixed components.

Both use nonces to achieve confidentiality in the presence of repeated queries or blocks.

Authenticated encryption

Authenticated encryption — a single-key construction that achieves both confidentiality and data integrity.

Data integrity/authentication means that a decryptable ciphertext must have been produced with a secret key. Hence most ciphertexts must decrypt to $\perp$.

Several types:

- Modes of operation (OCB, EAX, CCM, GCM), which invoke an arbitrary blockcipher;
- Dedicated constructions (Helix/Phelix, Grain128a), which use fixed components.

Both use nonces to achieve confidentiality in the presence of repeated queries or blocks.

Furthermore, some input must be authenticated but not encrypted (e.g., routing information). It is called associated data (AD).

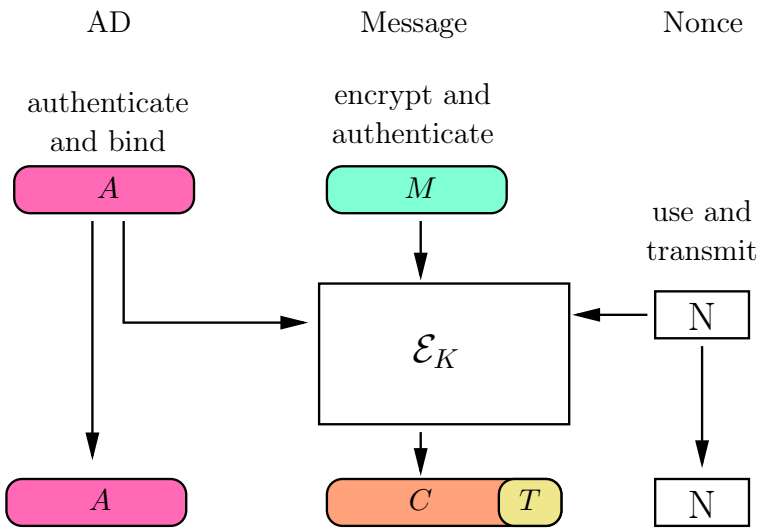
Authenticated encryption with associated data

Encryption:

$$\mathcal{E} : \mathcal{K} \times \mathcal{N} \times \mathcal{A} \times \mathcal{M} \to \mathcal{C}$$

Decryption:

$$\mathcal{D} : \mathcal{K} \times \mathcal{N} \times \mathcal{A} \times \mathcal{C} \to \mathcal{M} \cup \{\perp\}.$$



Confidentiality:

- Ciphertexts indistinguishable from random strings;

Data integrity:

- Most of seemingly valid ciphertexts decrypt to $\perp$.

Breaking AE

Find an attack that violate any security property.

Breaking AE

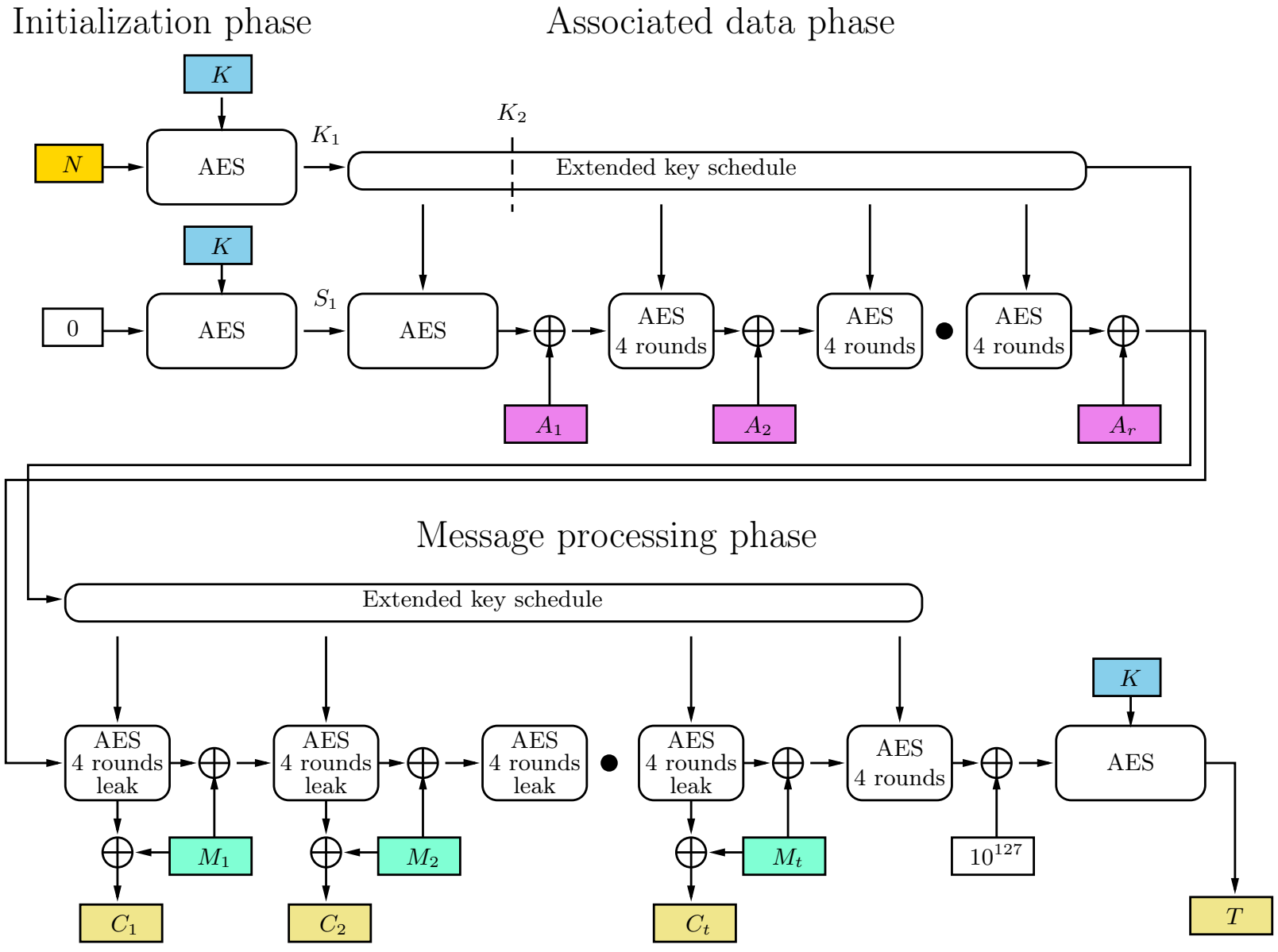
Find an attack that violate any security property.

In our case — forgery attack, i.e. constructing a ciphertext that decrypts to $M \neq \perp$.

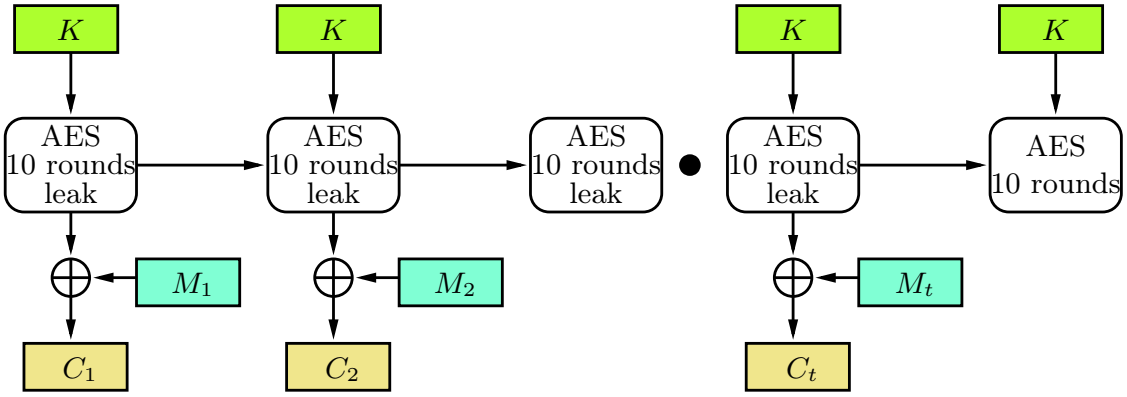
Breaking AE

In our case, (existential) forgery attack means:

- We are given access to the encryption oracle
  [message $\mathcal{M}$] $\times$ [nonce $\mathcal{N}$] $\longrightarrow$ [ciphertext+tag $\mathcal{C}$];
- Note that $\mathcal{C}$ has some redundancy: most seemingly valid ciphertexts are not decryptable.
- Ask $C = E_K(M, N)$ (we ignore associated data);
- Construct C' such that $D_K(C') = M' \neq \perp$.

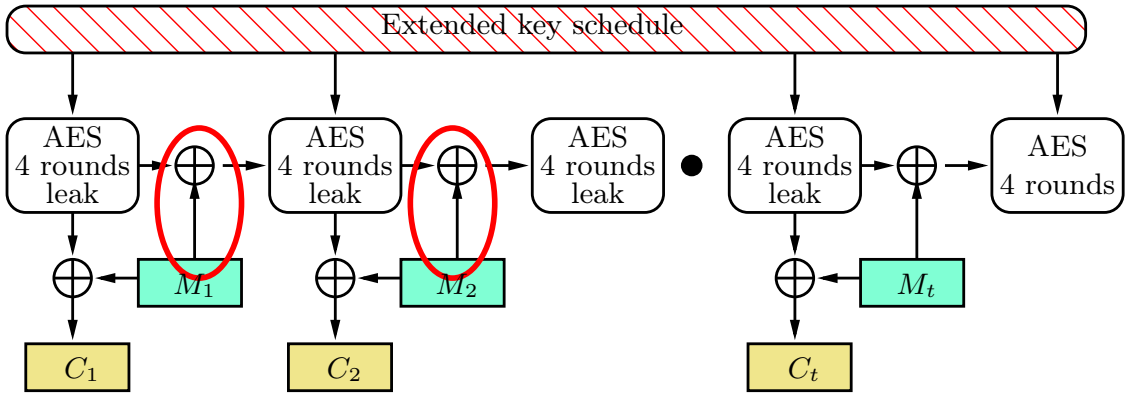

Initialization phase
Associated data phase
K
N
AES
K1
K2
Extended key schedule
K
0
AES
S1
AES
AES 4 rounds
AES 4 rounds
AES 4 rounds
A1
A2
Ar
Message processing phase
Extended key schedule
AES 4 rounds leak
AES 4 rounds leak
AES 4 rounds leak
AES 4 rounds leak
AES 4 rounds
K
AES
M1
M2
Mt
10^127
C1
C2
Ct
T

LEX stream cipher



Two crucial differences: key schedule and message injection

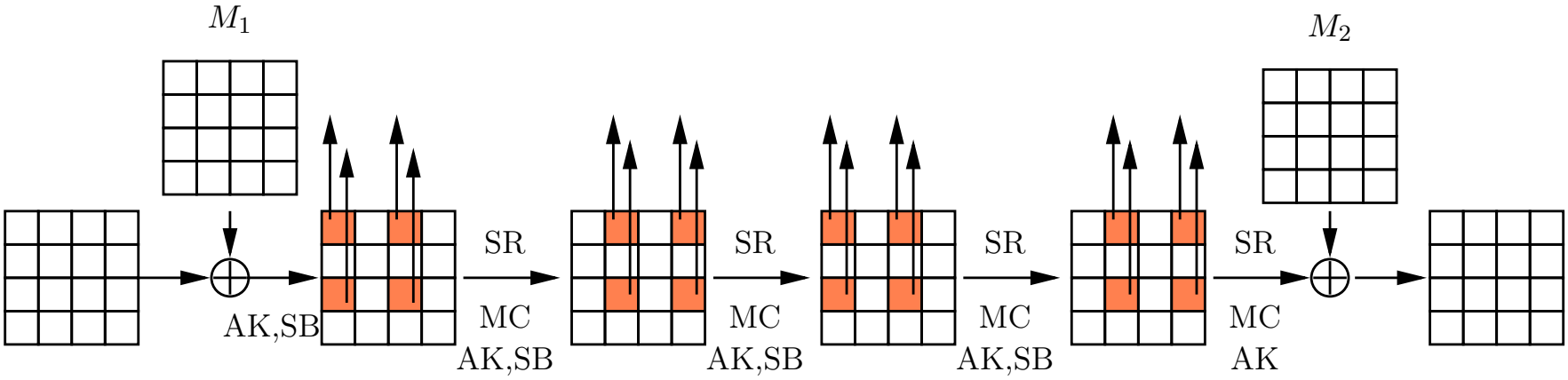
ALE scheme



The latter helps.

Leakage under scope

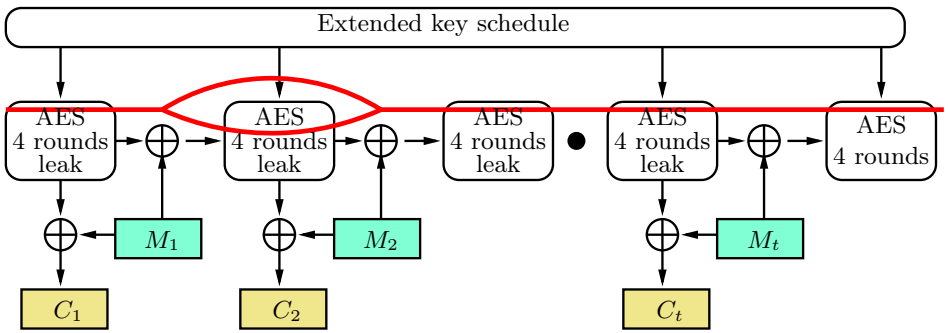
Bytes are extracted just after SubBytes operation:



Hence differential properties of the internal state are partly known.

Main idea I

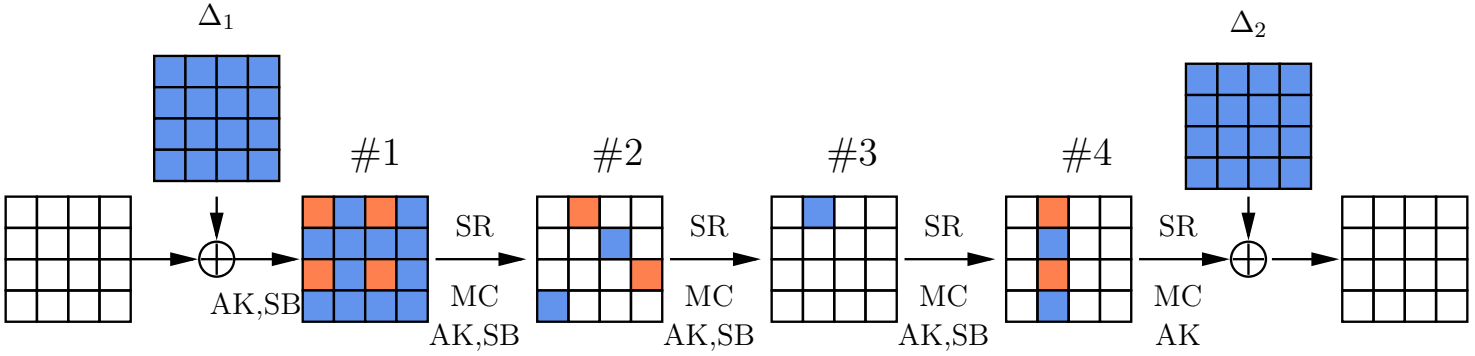
Make a local collision in the state:



Hence the same tag for a fresh ciphertext.

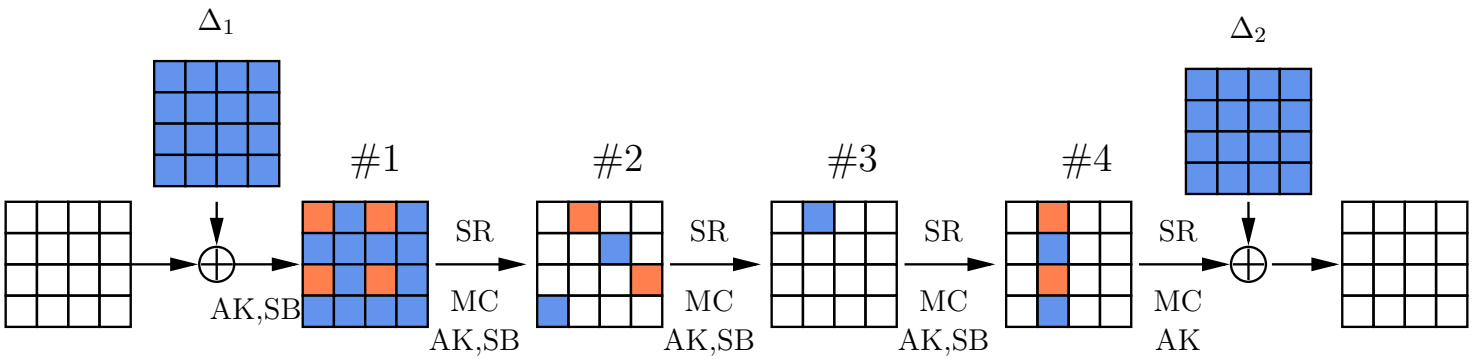
Main idea II

We know the extracted bytes and how a difference would go through it

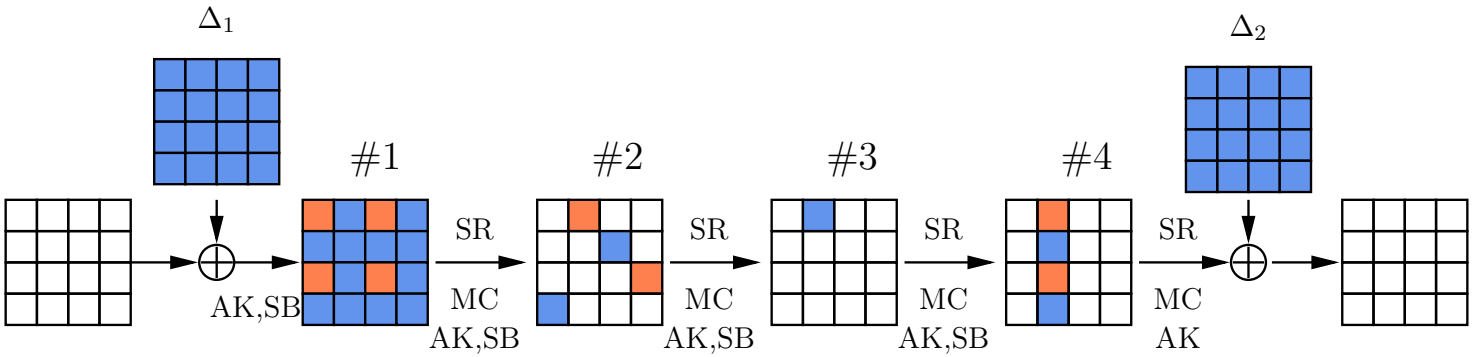


Unknowns

25 total active S-boxes, only 17 unknown:



Start in the middle, assume highest differential probability 2^{-6} everywhere:

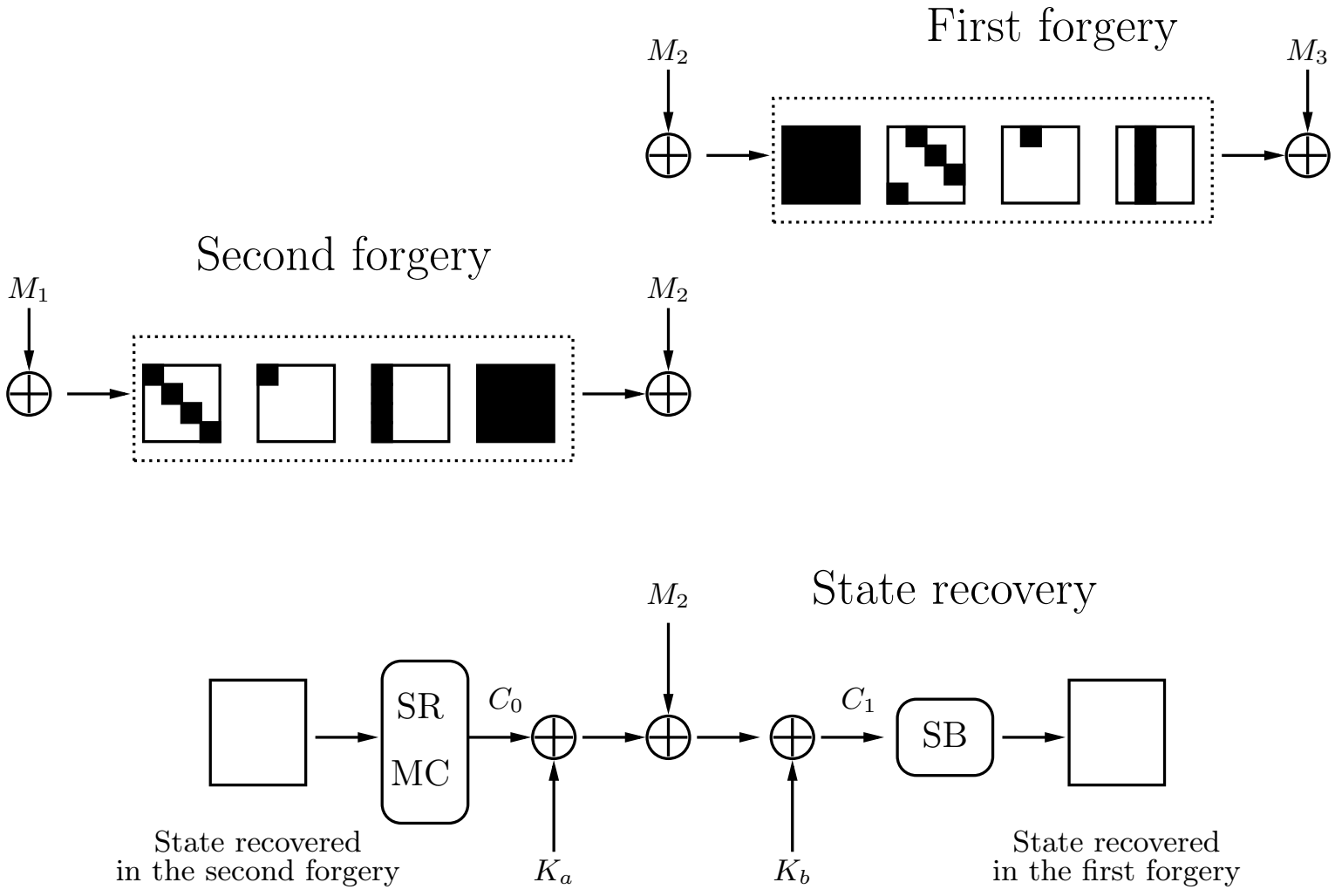


Given output differences, construct a colliding ciphertext (hence a forgery) with probability 2^{-102}.

Complexity

- The designers put the upper bound 2^{40} on the data encrypted on a single key.
- Thus we use other trails if we want to stick to the same message.
- Total 2^{119} attempts before the first forgery, if only one message is known (data/complexity tradeoff).

Two consecutive forgeries for the same message yield state recovery:



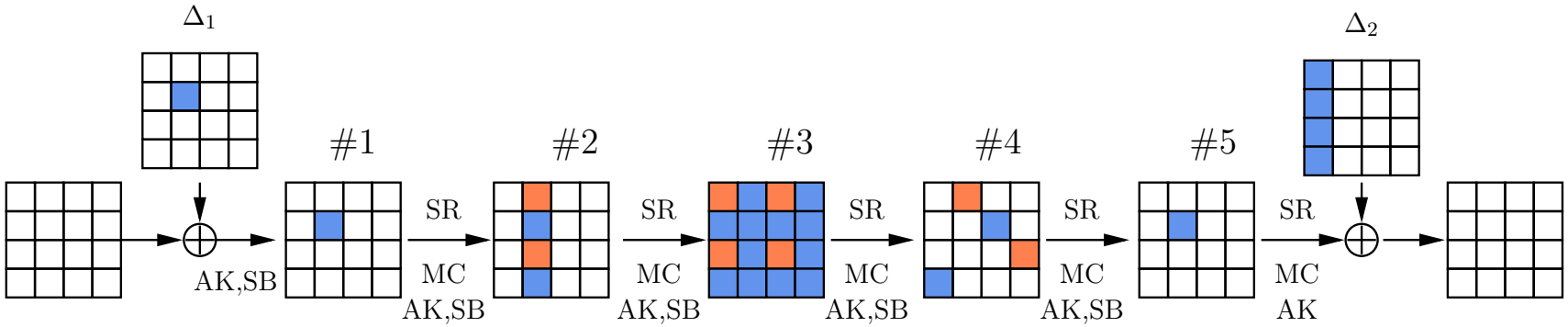
Results

Data	Verification attempts	Memory	Security claim
Forgery			
2^{102}	2^{102}	negl.	not violated
2^{40}	2^{110}	negl.	violated
1	2^{119}	negl.	violated
1	1	negl.	violated, success rate 2^{-119}
State recovery			
1	2^{120}	negl.	violated

Can we prevent the attack by just adding one more round?

5-round attack

Not really. 5-round trail



High data complexity (2^{80}), but still a gain over brute force.

Questions?