Presentation of RMAC
A randomized CBC–MAC beyond the Birthday Paradox Limit,

Éliane JAULMES, Antoine JOUX, Frédéric VALETTE
Crypto Lab
DCSSI

Overview

1. CBC–MAC: definitions and properties
2. Security Arguments
3. Application to the **AES**

Message Authentication Code

- MAC: authentication in secret key settings
- Message $M \longrightarrow \text{MAC}_K(M) = T$
- Sender sends (M, T)
- Receiver verifies $T = \text{MAC}_K(M)$
- Forgery attack on MAC: Find a valid (M, T)

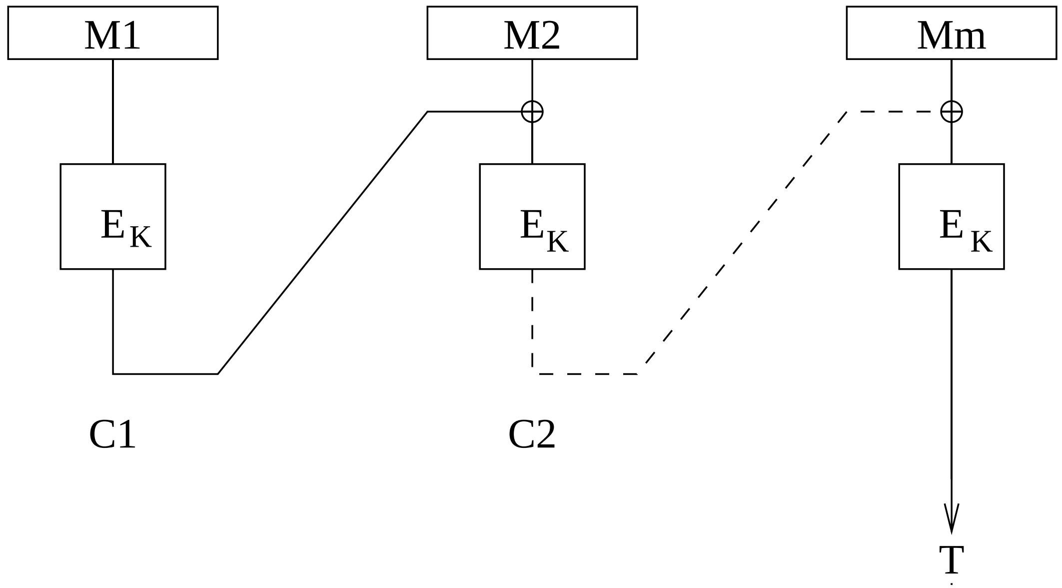
CBC–MAC

- Built from a block cipher E_K
- Message $M = M_1, M_2, \ldots M_m$: m blocks of n bits
- When the size of M is not multiple of $n \rightarrow$ padding
- Principle: encrypt with E_K in **CBC** mode
- MAC = last output of the **CBC** chain

The Padding

- Classical Padding: $M \to M||1||0\ldots0$
- Add ‘1’ and enough ‘0’ to fill the block
- All messages are padded
- From now: message length is a multiple of n

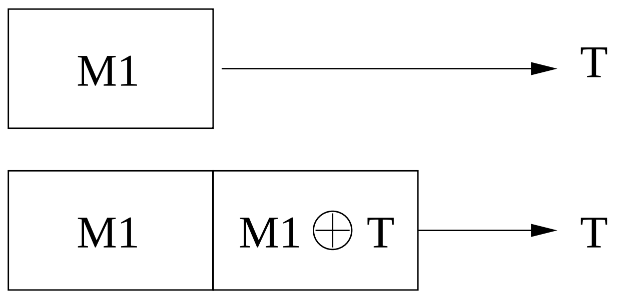
The elementary CBC–MAC



$$
\begin{aligned}
C_0 &= 0^n \\
C_i &= E_K(M_i \oplus C_{i-1}) \text{ for } i \text{ in } 1 \cdots m \\
T &= C_m
\end{aligned}
$$

Analysis of the elementary CBC–MAC

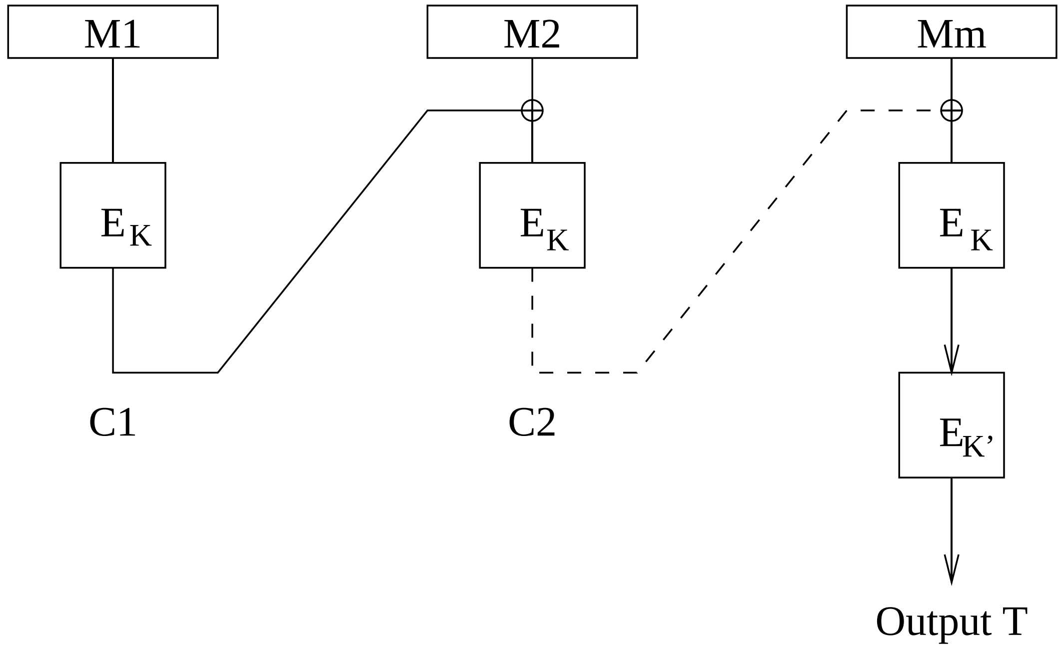
- Proven secure for fixed-length messages [BeKiRo-94]
- Insecure when message length varies



DMAC

- Elementary CBC–MAC insecure
- Need a secure CBC–MAC for messages of different length
- Solution **DMAC**:
  - At the end of **CBC** chain
  - Encrypt result with $E_{K'}$
  - $K' \neq K$

Description of DMAC



$$C_0 = 0^n$$
$$C_i = E_K(M_i \oplus C_{i-1}) \text{ for } i \text{ in } 1 \cdots m$$
$$T = E_{K'}(C_m)$$

Proving the security

- Model: **DMAC** scheme with random permutations
- Adversary: access to an oracle computing the MAC
- Security proven in this setting
- Replace random permutations by block ciphers
- Secure block cipher $\sim$ random permutation

Security of DMAC

- Proven secure in [BlRo00] and [PeRa97]
- In the preceeding model:
  - q questions to the oracle
  - Messages of at most m_{max} blocks

$$\mathbf{Adv}^{\mathbf{Dmac}}(\mathcal{A}) \leq \frac{2q^2 m_{max}^2 + q^2}{2^n}$$

Proof idea (1)

- Perfect MAC = random function
- Distinguish our model of **DMAC** from a random function:
  - Distinguish **DMAC** with random functions from a random function
  - Distinguish random functions from random permutations

Proof idea (2)

E_K and $E_{K'}$ replaced by f_1 and f_2, random functions

- Entries of f_2 all different $\rightarrow$ random outputs
- Collision: M and M' give same **CBC** output
- Probability to distinguish = Collision probability $V_n(M, M')$

Proof idea (3)

- Adversary: q questions of size at most m_{max}
- Collisions: $V_n(M, M') = \dfrac{(m+m')^2}{2^n}$
- Summing for all messages: $\displaystyle\sum_{M,M'} V_n(M, M') \leq \frac{2q^2 m_{max}}{2^n}$
- functions $\rightarrow$ permutations: $\mathbf{Adv}^{\mathbf{prf}/\mathbf{prp}} \leq \dfrac{q^2}{2^{n+1}}$
- Finally: $\mathbf{Adv}^{\mathbf{Dmac}}(\mathcal{A}) \leq \dfrac{2q^2 m_{max}^2 + q^2}{2^n}$

Tight bound: Attack with birthday paradox

- Output T has size n bits
- Query $\sqrt{2^n} = 2^{n/2}$ messages M^i
- $\Rightarrow$ with high probability 2 messages $M^{(1)}$ and $M^{(2)}$ s.t. $T^{(1)} = T^{(2)}$
- $\forall M'$, $\textbf{DMAC}(M^{(1)}||M') = \textbf{DMAC}(M^{(2)}||M')$

Existing solutions

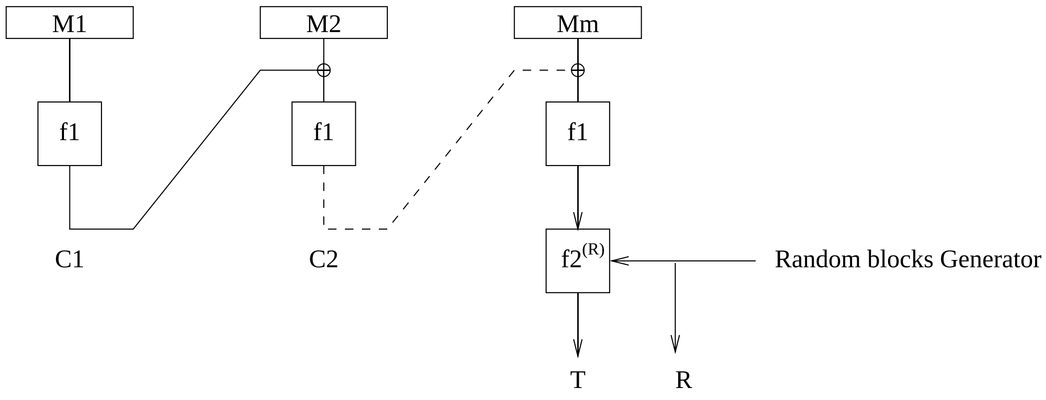
- MACRX [BeGoKr99]: not CBC–MAC based, expands MAC size by a factor 3
- CBC–MAC with counters: maintain counter
- Simple randomized CBC–MAC: L-collisions

⇒ Find a randomized CBC–MAC with proven security and easy to use

Principle of RMAC

- **RMAC** = Randomized CBC–MAC
- Apply a random function at the end of **CBC** computation
- MAC = output + information of the chosen function

Description of our solution: RMAC



$$
\begin{aligned}
C_0 &= 0^n \\
C_i &= f_1(M_i \oplus C_{i-1}) \text{ for } i \text{ in } 1 \cdots m \\
T &= f_2^{(R)}(C_m) \\
MAC &= (T, R)
\end{aligned}
$$

Description of the model

- f_1 random permutation
- $f_2^{(R)}$ random permutations
- Adversary has access to two oracles:
  - MAC generation oracle
  - MAC verification oracle

Security of RMAC

- Optimal construction:
  - Result of MAC on 2 blocks
  - Number of ciphering the same as **DMAC**
- Adversary A asks questions of total size L blocks

$$\mathbf{Adv}^{\mathbf{Rmac}}(\mathcal{A}) \leq \frac{4nL + 4L + 1}{2^n}$$

- Security in 2^n: birthday paradox = exhaustive search

Proof idea (1)

- Perfect randomized MAC = family of random functions
- Distinguish our model of **RMAC** from a family of random functions:
  - Distinguish **RMAC** with f_1, $f_2^{(R)}$ random functions from a family of random functions
  - Distinguish random functions from random permutations

Proof idea (2)

- Entries of $f_2^{(R)}$ all different $\rightarrow$ random outputs
- Different $R \rightarrow$ random outputs
- Collision: M and M' give same **CBC** output **and** $R = R'$
- Probability to distinguish = Collision probability
  - Collisions with the generation oracle
  - Collisions with the verification oracle

Collisions with generations

- Adversary: total length of queries bounded by L
- Collisions within a group a q messages:

$$\mathbf{Pr} \leq \frac{3q \sum_{i=1}^{q} m_i}{2^n}$$

- Size of the groups is less than n with probability $\frac{1}{2^n}$
- Summing for all messages:

$$\mathbf{Pr} \leq \frac{3nL}{2^n}$$

Collisions with verifications

- Adversary: total length of queries bounded by L
- A large group may exists but we may only compare with a reference message
- Collisions with a reference message:

$$\mathbf{Pr} \leq \frac{3 \sum_{i=0}^{q} m_i}{2^n}$$

- Summing for all messages:

$$\mathbf{Pr} \leq \frac{3L}{2^n}$$

Proof idea (3)

- Adversary: total length of queries bounded by L
- Collisions:

$$\mathbf{Pr} \leq \frac{3nL + 3L + 1}{2^n}$$

- PRF/PRP switching:

$$\mathbf{Adv} \leq \frac{nL + L}{2^n}$$

- Finally:

$$\mathbf{Adv}^{\mathbf{Rmac}}(\mathcal{A}) \leq \frac{4nL + 4L + 1}{2^n}$$

Application a block cipher

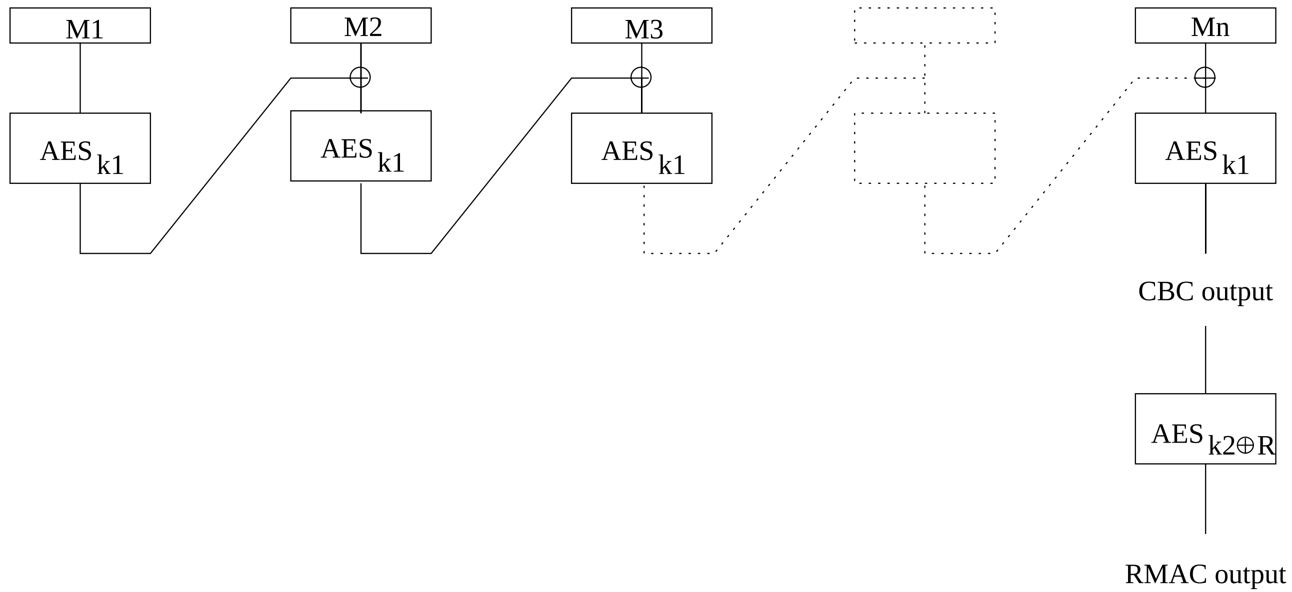
- $f_1 = E_K$
- $f_2^R = E_{R \oplus K'}$
- f_2^R not chosen at random but among a **known** family
- $\Rightarrow$ need to modify the model

New model

- f_2^R chosen in a known family F
- Adversary has access to F through an oracle
- The security becomes:

$$\mathbf{Adv}^{\mathbf{Rmac}}(\mathcal{A}) \leq \frac{5nL + 4L + 2}{2^n}$$

Application to the AES



$$
\begin{aligned}
C_0 &= 0^{128} \\
C_i &= \textbf{AES}_{K_1}(M_i \oplus C_{i-1}) \text{ for } i \text{ in } 1 \cdots m \\
T &= \textbf{AES}_{K_2 \oplus R}(C_m) \\
\textbf{RMAC}(M) &= (T, R)
\end{aligned}
$$

Advantages of this construction

- Computation time identical to **DMAC**:
  - $m+1$ computations
  - 2 keys
- The security is

$$\mathbf{Adv}_{\mathbf{RMAC}_{\mathbf{AES}}} \leq \frac{5 \cdot 128L + 4L + 2}{2^{128}} + \frac{t}{2^{128}} \leq \frac{645L + t}{2^{128}}$$

- Secure for $L \leq 2^{118}$