Secure Message Authentication Codes against Related-Key Attack

Rishiraj Bhattacharyya Arnab Roy

March 12, 2013

Outline

1. Background
   - Related-Key Attack
   - Message Authentication Codes
2. Related-Key Security of MAC
   - MAC against RK Adversary
   - RKD class
   - Attack against MAC
3. Related-Key Secure MAC
   - First Step
   - Design at a High Level
   - Construction

Outline

Related-Key Attack

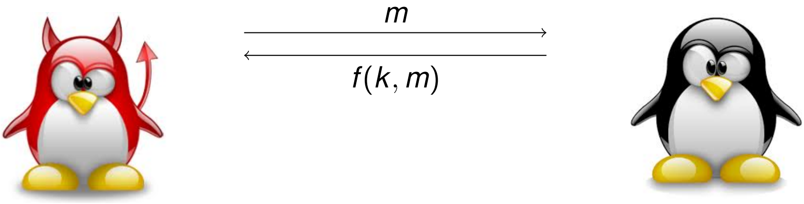
- Adversary can make queries to the primitive with secret key as well as with some function of the secret key

Related-Key Attack

- Adversary can make queries to the primitive with secret key as well as with some function of the secret key

Related-Key Attack

- Adversary can make queries to the primitive with secret key as well as with some function of the secret key



Related-Key Attack

- Adversary can make queries to the primitive with secret key as well as with some function of the secret key

$$\xrightarrow{\quad m \quad} \qquad \xleftarrow[\quad f(k, m) \quad]{}$$

$$\xrightarrow{\quad m, \phi \quad} \qquad \xleftarrow[\quad f(\phi(k), m) \quad]{}$$

Related-Key Attack

- Adversary can make queries to the primitive with secret key as well as with some function of the secret key

$$\xrightarrow{\quad m \quad}$$
$$\xleftarrow[\quad f(k, m) \quad]{}$$
$$\xrightarrow{\quad m, \phi \quad}$$
$$\xleftarrow[\quad f(\phi(k), m) \quad]{}$$

$\boxed{\phi : \mathcal{K} \rightarrow \mathcal{K} \text{ is the RKD function chosen by adversary}}$

- Proposed by Biham in 1993
- Many well known attacks, including the attack on AES
- Formal theoretical model introduced by Bellare and Kohno in 2003
- A series of work in recent past (Bellare Cash 2010, Bellare Cash Miller 2011)
- Related-key attack on HMAC AsiaCrypt 2012.

Outline

Message Authentication Codes

- Message Authentication Codes: $F : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$

Message Authentication Codes

- Message Authentication Codes: $F : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$

Message Authentication Codes

- Message Authentication Codes: $F : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$

$$\xrightarrow{\quad m_1 \quad}$$
$$\xleftarrow[\quad F(k, m_1) \quad]{}$$

Message Authentication Codes

- Message Authentication Codes: $F : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$

$$\xrightarrow{\quad m_1 \quad}$$
$$\xleftarrow[\quad F(k, m_1) \quad]{}$$
$$\xrightarrow{\quad m_q \quad}$$
$$\xleftarrow[\quad F(k, m_q) \quad]{}$$

Message Authentication Codes

- Message Authentication Codes: $F : \mathcal{K} \times \mathcal{D} \rightarrow \mathcal{R}$

$$\xrightarrow{m_1}$$
$$\xleftarrow[F(k, m_1)]{}$$
$$\xrightarrow{m_q}$$
$$\xleftarrow[F(k, m_q)]{}$$

(m^*, σ^*): $m^* \notin \mathcal{Q}$

Outline

Outline

MAC against RK Adversary

- Message Authentication Codes: $F : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$

MAC against RK Adversary

- Message Authentication Codes: $F : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$

MAC against RK Adversary

- Message Authentication Codes: $F : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$

$$\xrightarrow{\quad (m_1, \phi) \quad}$$
$$\xleftarrow[\quad F(\phi(k), m_1) \quad]{}$$

MAC against RK Adversary

- Message Authentication Codes: $F : \mathcal{K} \times \mathcal{D} \rightarrow \mathcal{R}$

$$\xrightarrow{\quad (m_1, \phi) \quad}$$
$$\xleftarrow[\quad F(\phi(k), m_1) \quad]{}$$
$$\xrightarrow{\quad (m_q, \phi) \quad}$$
$$\xleftarrow[\quad F(\phi(k), m_q) \quad]{}$$

MAC against RK Adversary

- Message Authentication Codes: $F : \mathcal{K} \times \mathcal{D} \rightarrow \mathcal{R}$

$$\xrightarrow{(m_1, \phi)}$$
$$\xleftarrow[F(\phi(k), m_1)]{}$$
$$\xrightarrow{(m_q, \phi)}$$
$$\xleftarrow[F(\phi(k), m_q)]{}$$

$(m^*, \sigma^* = F(k, m^*))$

MAC against RK Adversary

- Message Authentication Codes: $F : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$

$$\xrightarrow{(m_1, \phi)}$$

$$\xleftarrow[F(\phi(k), m_1)]{}$$

$$\xrightarrow{(m_q, \phi)}$$

$$\xleftarrow[F(\phi(k), m_q)]{}$$

$(m^*, \sigma^* = F(k, m^*))$

- $(m^*, \mathrm{id}) \notin \mathcal{Q}$

MAC against RK Adversary

- Message Authentication Codes: $F : \mathcal{K} \times \mathcal{D} \to \mathcal{R}$

$$\xrightarrow{(m_1, \phi)}$$
$$\xleftarrow[F(\phi(k), m_1)]{}$$
$$\xrightarrow{(m_q, \phi)}$$
$$\xleftarrow[F(\phi(k), m_q)]{}$$

$(m^*, \sigma^* = F(k, m^*))$

- $(m^*, \mathrm{id}) \notin \mathcal{Q}$ or $(m^*, \phi) \notin \mathcal{Q}$

Outline

A closer look at the RKD class

- For arbitrary RKD class, it is impossible to get provable security against Related Key Attack. (Bellare Kohno 2003).
- For prf, RKD class should be collision resistant and entropy preserving (Bellare Kohno 2003); trivial attacks using constant RKD functions.

A closer look at the RKD class

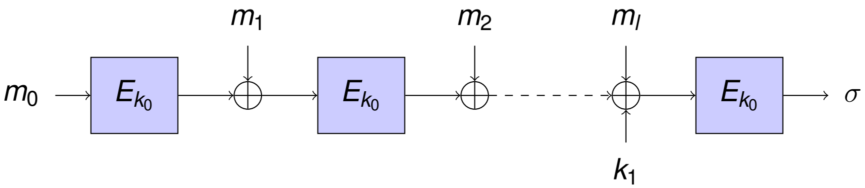
- For arbitrary RKD class, it is impossible to get provable security against Related Key Attack. (Bellare Kohno 2003).
- For prf, RKD class should be collision resistant and entropy preserving (Bellare Kohno 2003); trivial attacks using constant RKD functions.

Theorem

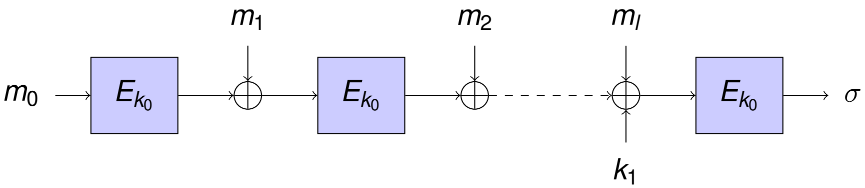
If F is a MAC then F is related-key unforgeable against constant RKD.

Outline

Related-Key Attack against popular MACs

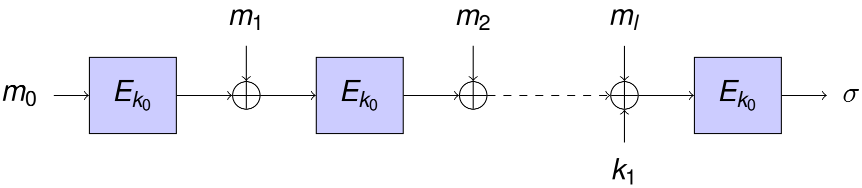


Related-Key Attack against popular MACs



$$\boxed{F(k_0, k_1 \oplus i, M) = F(k_0, k_1, M \oplus i)}$$

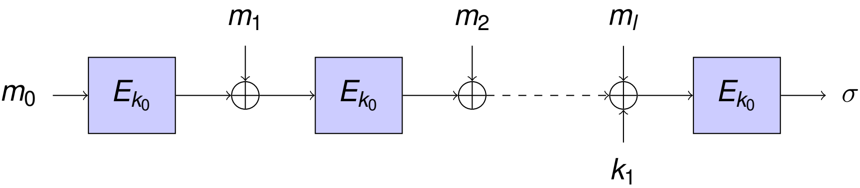
Related-Key Attack against popular MACs



$$\boxed{F(k_0, k_1 \oplus i, M) = F(k_0, k_1, M \oplus i)}$$

$$\boxed{\phi_i(k) = k \oplus i}$$

Related-Key Attack against popular MACs



$$\boxed{F(k_0, k_1 \oplus i, M) = F(k_0, k_1, M \oplus i)}$$

$$\boxed{\phi_i(k) = k \oplus i}$$

$$\boxed{(M \oplus i, \sigma)}$$

Summary of Attacks

- XCBC is not related key secure
- Same attack can be applied to TMAC with little modification
- We also show related-key attacks against ECBC and FCBC

Outline

Outline

Technical Tool: ICTPR Hash Function

- Identity Collision Resistant Hash

Technical Tool: ICTPR Hash Function

- Identity Collision Resistant Hash

Technical Tool: ICTPR Hash Function

- Identity Collision Resistant Hash

$$\xrightarrow{m_1, \phi_1}$$

$$\xleftarrow[H(\phi_1(k), m_1)]{}$$

Technical Tool: ICTPR Hash Function

- Identity Collision Resistant Hash

$$\xrightarrow{\quad m_1,\ \phi_1 \quad}$$
$$\xleftarrow[\quad H(\phi_1(k),\ m_1) \quad]{}$$
$$\xrightarrow{\quad m_q,\ \phi_q \quad}$$
$$\xleftarrow[\quad H(\phi_q(k),\ m_q) \quad]{}$$

Technical Tool: ICTPR Hash Function

- Identity Collision Resistant Hash

$$m_1, \phi_1 \quad \longrightarrow$$

$$\longleftarrow \quad H(\phi_1(k), m_1)$$

$$m_q, \phi_q \quad \longrightarrow$$

$$\longleftarrow \quad H(\phi_q(k), m_q)$$

(m_i, m_j): $H(\phi_i(k), m_i) = H(k, m_j)$, $i < j$

Technical Tool: ICTPR Hash (contd.)

- Target Preimage Resistant Hash

Technical Tool: ICTPR Hash (contd.)

- Target Preimage Resistant Hash

Technical Tool: ICTPR Hash (contd.)

- Target Preimage Resistant Hash

$\boxed{\{z_1, z_2, .., z_t\} \in \mathcal{R} \text{ and } \Phi}$

Technical Tool: ICTPR Hash (contd.)

- Target Preimage Resistant Hash

$\boxed{\{z_1, z_2, .., z_t\} \in \mathcal{R} \text{ and } \Phi}$

$$\xrightarrow{m_1, \phi_1}$$
$$\xleftarrow[H(\phi_1(k), m_1)]{}$$

Technical Tool: ICTPR Hash (contd.)

- Target Preimage Resistant Hash

$\boxed{\{z_1, z_2, .., z_t\} \in \mathcal{R} \text{ and } \Phi}$

$$\xrightarrow{m_1, \phi_1}$$
$$\xleftarrow[H(\phi_1(k), m_1)]{}$$
$$\xrightarrow{m_q, \phi_q}$$
$$\xleftarrow[H(\phi_q(k), m_q)]{}$$

Technical Tool: ICTPR Hash (contd.)

- Target Preimage Resistant Hash

$\boxed{\{z_1, z_2, .., z_t\} \in \mathcal{R} \text{ and } \Phi}$

$$\xrightarrow{m_1, \phi_1}$$
$$\xleftarrow[H(\phi_1(k), m_1)]{}$$
$$\xrightarrow{m_q, \phi_q}$$
$$\xleftarrow[H(\phi_q(k), m_q)]{}$$

$m^*\colon H(m^*, k) = z_i$

Outline

- ICTPR hash $H : \mathcal{K}_1 \times \{0,1\}^* \to \mathcal{D}$ over Φ_1
- $F : \mathcal{K}_2 \times \mathcal{D} \to \mathcal{R}$ is weak RK unforgeable MAC over Φ_2 with identity fingerprint $w_1, w_2, .., w_d$

Theorem

With the above mentioned F and H, $G : (\mathcal{K}_1 \times \mathcal{K}_2) \times \{0,1\}^* \to \mathcal{R}$ defined as

$$G(k_1, k_2, m) = F(k_1, H(k_2, m \| F(k_1, w_1) \| F(k_1, w_2) \| \cdots \| F(k_1, w_d)))$$

is related-key unforgeable against chosen message attack, over component induced RKD set $\Phi_1 \times \Phi_2$

Outline

Towards Main Concstruction

- The construction of

  $$G(k_1, k_2, m) = F(k_1, H(k_2, m \| F(k_1, w_1) \| F(k_1, w_2) \| \cdots \| F(k_1, w_d)))$$

  is in the line of previous work.

Towards Main Concstruction

- The construction of

$$G(k_1, k_2, m) = F(k_1, H(k_2, m \| F(k_1, w_1) \| F(k_1, w_2) \| \cdots \| F(k_1, w_d)))$$

  is in the line of previous work.
- Major difference: ICTPR Hash (instead of the unkeyed collision resistant hash function with tailor made range used by Bellare and Cash)

Towards Main Concstruction

- The construction of

$$G(k_1, k_2, m) = F(k_1, H(k_2, m \| F(k_1, w_1) \| F(k_1, w_2) \| \cdots \| F(k_1, w_d)))$$

  is in the line of previous work.
- Major difference: ICTPR Hash (instead of the unkeyed collision resistant hash function with tailor made range used by Bellare and Cash)
- Next we construct ICTPR Hash function from FIL-RK unforgeable function

Towards Main Concstruction

- The construction of

$$G(k_1, k_2, m) = F(k_1, H(k_2, m \| F(k_1, w_1) \| F(k_1, w_2) \| \cdots \| F(k_1, w_d)))$$

  is in the line of previous work.
- Major difference: ICTPR Hash (instead of the unkeyed collision resistant hash function with tailor made range used by Bellare and Cash)
- Next we construct ICTPR Hash function from FIL-RK unforgeable function
- This is done in two steps:

Towards Main Concstruction

- The construction of

  $$G(k_1, k_2, m) = F(k_1, H(k_2, m \| F(k_1, w_1) \| F(k_1, w_2) \| \cdots \| F(k_1, w_d)))$$

  is in the line of previous work.
- Major difference: ICTPR Hash (instead of the unkeyed collision resistant hash function with tailor made range used by Bellare and Cash)
- Next we construct ICTPR Hash function from FIL-RK unforgeable function
- This is done in two steps:
  1. VIL ICTPR Hash from a FIL ICTPR compression function

Towards Main Concstruction

- The construction of

$$G(k_1, k_2, m) = F(k_1, H(k_2, m\|F(k_1, w_1)\|F(k_1, w_2)\| \cdots \|F(k_1, w_d)))$$

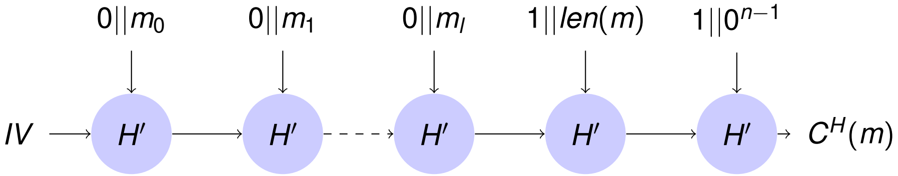
  is in the line of previous work.
- Major difference: ICTPR Hash (instead of the unkeyed collision resistant hash function with tailor made range used by Bellare and Cash)
- Next we construct ICTPR Hash function from FIL-RK unforgeable function
- This is done in two steps:
  1. VIL ICTPR Hash from a FIL ICTPR compression function
  2. FIL ICTPR Hash from FIL RK-MAC

VIL-ICTPR Hash from ICTPR Compression Functon

$$\boxed{H = pfNI^{H'}(k, m)}$$

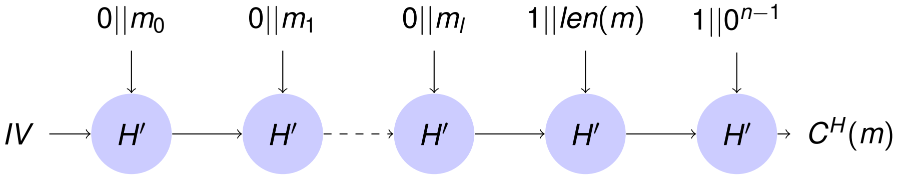
VIL-ICTPR Hash from ICTPR Compression Functon

$$\boxed{H = pfNI^{H'}(k, m)}$$



VIL-ICTPR Hash from ICTPR Compression Functon

$$\boxed{H = pfNI^{H'}(k, m)}$$



Lemma

If $H' : \mathcal{K} \times \{0,1\}^{2n} \to \{0,1\}^n$ is ICTPR then $H : \mathcal{K} \times \{0,1\}^* \to \{0,1\}^n$ is ICTPR.

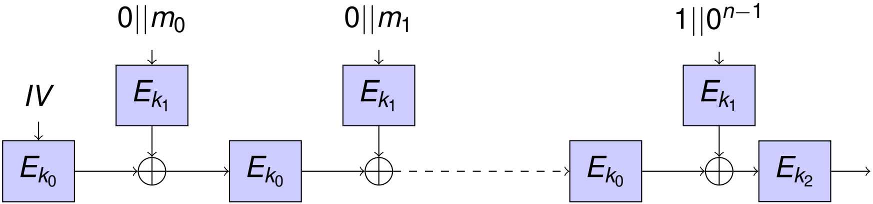
FIL-ICTPR Hash using FIL RK-MAC

- We take $H'_{k1,k_2}(x1, x2) = F(k_1, x_1) \oplus F(k_2, x_2)$ where $F : \mathcal{K} \times \{0,1\}^n \rightarrow \{0,1\}^n$ RK unforgeable.

Lemma

If F is RK unforgeable over RKD set Φ with identity fingerprint $w_1, w_2, .., w_d$ then $H = pfNI^{H'}$ is ICTPR over the RKD set $\Psi : \{0,1\}^\kappa \times \{0,1\}^\kappa \rightarrow \{0,1\}^\kappa$ defined as $((\Phi \setminus \{\mathrm{id}\}) \times \Phi) \cup (\mathrm{id}, \mathrm{id})$

Provable Secure Mode



Modified Enciphered CBC preserves related-key unforgeability.

Constructions using Collision Resistant Hash Function

- $F : \mathcal{K}_2 \times \mathcal{D} \to \mathcal{R}$ is key-homomorphic MAC over Φ with identity fingerprint $w_1, w_2, .., w_d$
- Collision Resistant hash $H : \{0,1\}^* \to \mathcal{D} \setminus \{w_1, w_2, .., w_d\}$

Theorem

$$G(k_1, k_2, m) = F(k_1, H(k_2, m \| F(k_1, w_1) \| F(k_1, w_2) \| \cdots \| F(k_1, w_d))$$

is related-key unforgeable over Φ

Constructions using Collision Resistant Hash Function

- $F : \mathcal{K}_2 \times \mathcal{D} \to \mathcal{R}$ is key-homomorphic MAC over Φ with identity fingerprint $w_1, w_2, .., w_d$
- Collision Resistant hash $H : \{0,1\}^* \to \mathcal{D} \setminus \{w_1, w_2, .., w_d\}$

Theorem

$$G(k_1, k_2, m) = F(k_1, H(k_2, m \| F(k_1, w_1) \| F(k_1, w_2) \| \cdots \| F(k_1, w_d))$$

is related-key unforgeable over Φ

Applications

Two constructions from DDH/CDH assumptions for claw-free class.

Summary

- formal security definition for Related-Key MAC
- MAC is inherently RK unforgeable under constant RKD function
- Mode of operation for RK unforgeable functions
- Finally construction of RK unforgeable MAC from DDH assumption using collision resistant hash function

THANK YOU !