CSL759: Cryptography and Computer Security

Ragesh Jaiswal

CSE, IIT Delhi

Message Authentication

Message Integrity/Authenticity

Accept/Reject

M, σ

K ⟷ K

Key exchange protocol

- Cryptographic goals:
  - M was sent by Alice and no one else.
  - M was not modified during transit.

Message Integrity/Authenticity

- A message authentication $MA = (T, V)$ is defined by two algorithms T and V:
  - $T_K(.)$ is known as the tag generation algorithm. For any message M, the *tag* of the message is given by $T_K(M)$.
  - $V_K(.,.)$ is known as the verification algorithm that outputs 1 indicating success and 0 indicating failure. So, $V_K(M, \sigma) = 0/1$.
- Any message authentication scheme $MA = (T, V)$ should satisfy the following *consistency* requirement:

$$\forall M, K, V_K\big(M, T_K(M)\big) = 1$$

Message Integrity/Authenticity

Accept/Reject

$M, \sigma = T_K(M)$

If $V_K(M, \sigma) = 1$
accept else reject

- A message authentication $MA = (T, V)$ is defined by two algorithms T and V:
  - $T_K(.)$ is known as the tag generation algorithm. For any message M, the *tag* of the message is given by $T_K(M)$.
  - $V_K(.,.)$ is known as the verification algorithm that outputs 1 indicating success and 0 indicating failure. So, $V_K(M, \sigma) = 0/1$.

Message Authentication Code (MAC)

- One way of designing Message Authentication schemes is to use a function family $F: \{0,1\}^k \times D \rightarrow \{0,1\}^n$ in the following manner:
  - $T_K(M) = F_K(M)$
  - $V_K(M, \sigma) = 1 \; iff \; \sigma = T_K(M)$.
- This means that tag generation is deterministic and stateless and verification is by tag re-recomputation.
- Such Message Authentication Schemes are called *Message Authentication Code* or MAC in short.
- Most authentication schemes are MACs.

Message Integrity/Authenticity

- Let us try to use a block cipher $E: \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$ (our familiar building block) to construct a message authentication scheme.
  - For a message $M = M_1 || \ldots || M_m$ spanning m blocks,
    $T_K(M) = E_K(M_1) \oplus E_K(M_2) \oplus \cdots \oplus E_K(M_m)$
  - $V_K(M, \sigma)$:
    - If $(T_K(M) = \sigma)$, then output 1 else output 0.

- Does this message authentication scheme look secure?

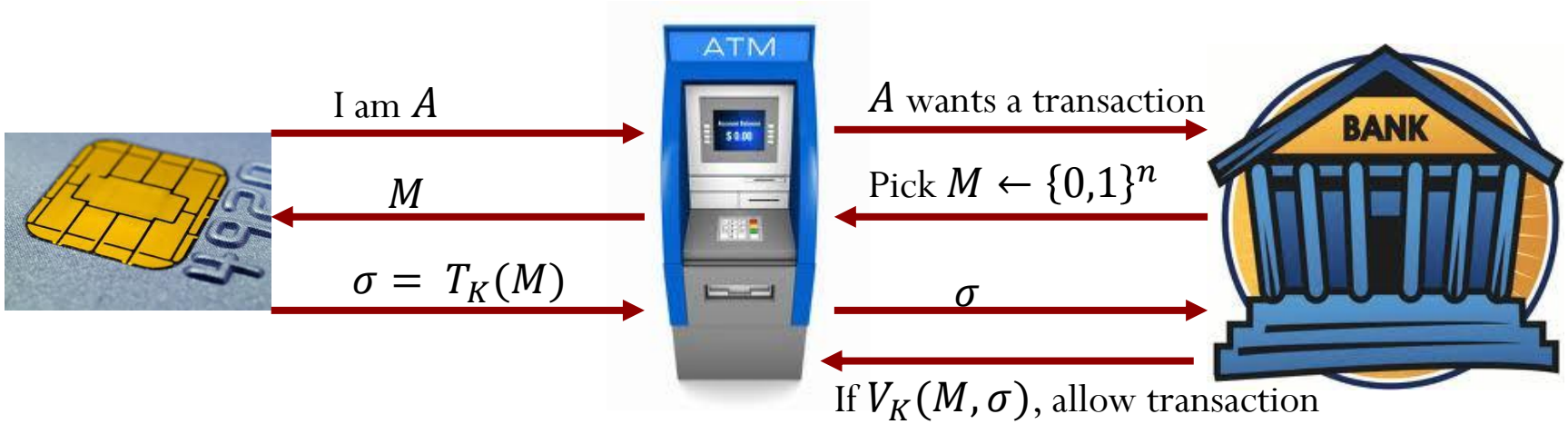
Security

Security

- What could be the attack scenarios:
  1. The adversary gets access to some message-tag pairs $(M_1, \sigma_1), \ldots, (M_l, \sigma_l)$ and is able to compute the secret key K. *(Key recovery under known message attack)*
  2. The adversary gets access to some message-tag pairs $(M_1, \sigma_1), \ldots, (M_l, \sigma_l)$ and is able to compute the correct tag for a *new* message. *(Message forgeability under known message attack)*
  3. The adversary gets access to message-tag pairs for messages of its choice $(M_1, \sigma_1), \ldots, (M_l, \sigma_l)$ and is able to compute the secret key K. *(Key recovery under chosen message attack)*
  4. The adversary gets access to message-tag pairs for messages of its choice $(M_1, \sigma_1), \ldots, (M_l, \sigma_l)$ and is able to compute the correct tag for a *new* message. *(Message forgeability under chosen message attack)*

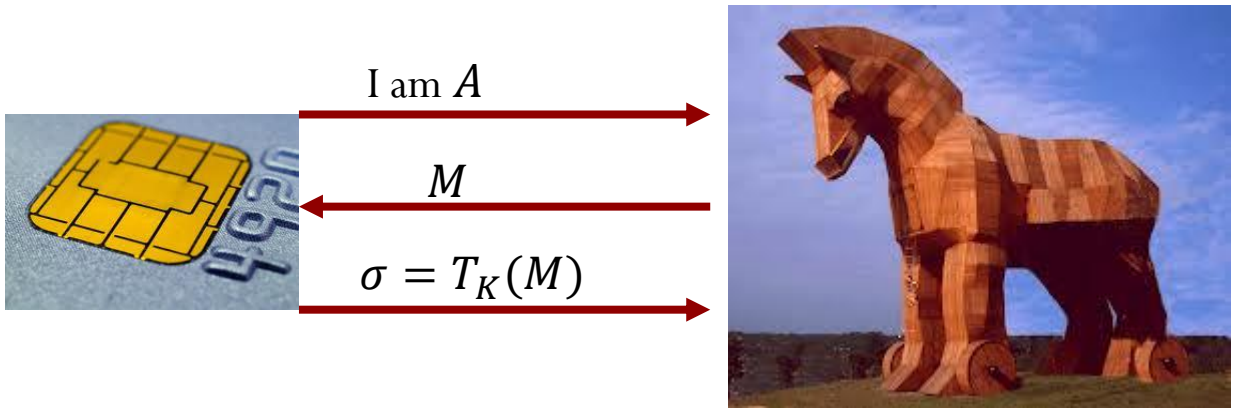
Security

- Known Message Attack (KMA) in real life:
  - A sniffing adversary observes message-tag pairs being exchanges between two parties sharing a secret key.
- Chosen Message Attack (CMA) in real life.
  - Authenticated message forwarding.
  - Trojan horse ATM.



Security

- Known Message Attack (KMA) in real life:
  - A sniffing adversary observes message-tag pairs being exchanges between two parties sharing a secret key.
- Chosen Message Attack (CMA) in real life.
  - Authenticated message forwarding.
  - Trojan horse ATM.



Security

- For any attack model (KMA or CMA), is the message forgeability notion of security stronger than key recovery notion of security?
  - Yes since if you can figure out the secret key, then you can forge a new message.
- CMA is a stronger attack model than KMA.
- So, the strong notion of security that we use for message authentication schemes is the *Unforgeability under Chosen Message Attack (UF-CMA)* notion.
- How do we formally define this security notion?
  - Define a security game/experiment.

Security

- For any attack model (KMA or CMA), is the message forgeability notion of security stronger than key recovery notion of security?
  - Yes since if you can figure out the secret key, then you can forge a new message.
- CMA is a stronger attack model than KMA.
- So, the strong notion of security that we use for message authentication schemes is the *Unforgeability under Chosen Message Attack (UF-CMA)* notion.
- How do we formally define this security notion?
  - Define a security game/experiment.
  - The challenger chooses a secret key and then allows the adversary to obtain tags of messages of its choice.The adversary may send message-tag pair for verification and it succeeds if it is able to produce a correct tag for a fresh message.

Security

- $UFCMA_{A,MA}$
  - Randomly pick the secret key $K \leftarrow \{0,1\}^k$.
  - Let $S \leftarrow \{\}$
  - When adversary makes a tag-generation query M_i, do the following:
    - Let $\sigma_i \leftarrow T_K(M_i)$.
    - $S = S \cup M_i$.
    - Return σ_i to the adversary.
  - When adversary makes a tag-verification query (M_j, σ_j), do the following:
    - If $(M_j \notin S) \wedge (V_K(M_j, \sigma_j) = 1)$, output 1.
    - Else if $(V_K(M_j, \sigma_j) = 1)$, return 1 to adversary
    - Else return 0 to adversary
  - Output 0

Security

- $UFCMA_{A,MA}$
  - Randomly pick the secret key $K \leftarrow \{0,1\}^k$.
  - Let $S \leftarrow \{\}$
  - When adversary makes a tag-generation query M_i, do the following:
    - Let $\sigma_i \leftarrow T_K(M_i)$.
    - $S = S \cup M_i$.
    - Return σ_i to the adversary.
  - When adversary makes a tag-verification query (M_j, σ_j), do the following:
    - If $(M_j \notin S) \wedge (V_K(M_j, \sigma_j) = 1)$, output 1.
    - Else if $(V_K(M_j, \sigma_j) = 1)$, return 1 to adversary
    - Else return 0 to adversary
  - Output 0

- $Adv_{uf-cma}(A, MA) = \Pr[UFCMA_{A,MA} = 1]$

Security

- Let us try to use a block cipher $E: \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$ (our familiar building block) to construct a message authentication scheme.
  - For a message $M = M_1 || \ldots || M_m$ spanning m blocks,
    $T_K(M) = E_K(M_1) \oplus E_K(M_2) \oplus \cdots \oplus E_K(M_m)$
  - $V_K(M, \sigma)$:
    - If $(T_K(M) = \sigma)$, then output 1 else output 0.
- Is the above MAC UF-CMA secure?

Security

- Let us try to use a block cipher $E: \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$ (our familiar building block) to construct a message authentication scheme.
  - For a message $M = M_1 || \ldots || M_m$ spanning m blocks, $T_K(M) = E_K(M_1) \oplus E_K(M_2) \oplus \cdots \oplus E_K(M_m)$
  - $V_K(M, \sigma)$:
    - If $(T_K(M) = \sigma)$, then output 1 else output 0.
- Is the above MAC UF-CMA secure?
  - Adversary A:
    - Make a tag generation quer for $0^n || 1^n$ and get back σ.
    - Make a verification query $(1^n || 0^n, \sigma)$.
  - Another Adversary A'
    - Make a verification query $(0^n || 0^n, 0^n)$.
  - What is $Adv_{uf-cma}(A, MA)$ and $Adv_{uf-cma}(A,' MA)$?

Security

- Let us try to use a block cipher $E: \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$ (our familiar building block) to construct a message authentication scheme.
  - For a message $M = M_1 || \ldots || M_m$ spanning m blocks,
    $$T_K(M) = E_K(M_1) \oplus E_K(M_2) \oplus \cdots \oplus E_K(M_m)$$
  - $V_K(M, \sigma)$:
    - If $(T_K(M) = \sigma)$, then output 1 else output 0.
- Is the above Message Authentication Scheme UF-CMA secure?
  - Adversary A:
    - Make a tag generation quer for $0^n || 1^n$ and get back σ.
    - Make a verification query $(1^n || 0^n, \sigma)$.
  - Another Adversary A'
    - Make a verification query $(0^n || 0^n, 0^n)$.
  - What is $Adv_{uf-cma}(A, MA)$ and $Adv_{uf-cma}(A, 'MA)$?
    - 1.

Replay Attack

Replay Attack

- Suppose we have a UF-CMA secure MAC and we use it to authenticate bank transactions.
- Consider the following scenario:

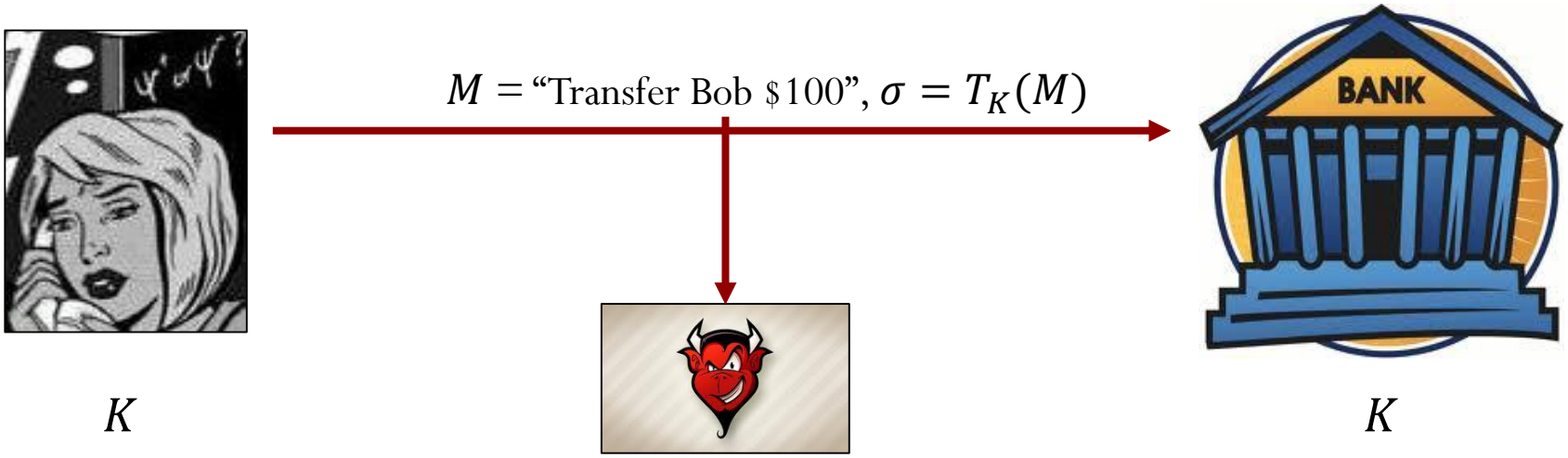
K

$M =$ "Transfer Bob \$100", $\sigma = T_K(M)$

K

Replay Attack

- Suppose we have a UF-CMA secure MAC and we use it to authenticate bank transactions.
- Consider the following scenario:



Replay Attack

- Suppose we have a UF-CMA secure MAC and we use it to authenticate bank transactions.
- Consider the following scenario:

$$M = \text{“Transfer Bob \$100”}, \sigma = T_K(M)$$

K

(M, σ)

(M, σ)

(M, σ)

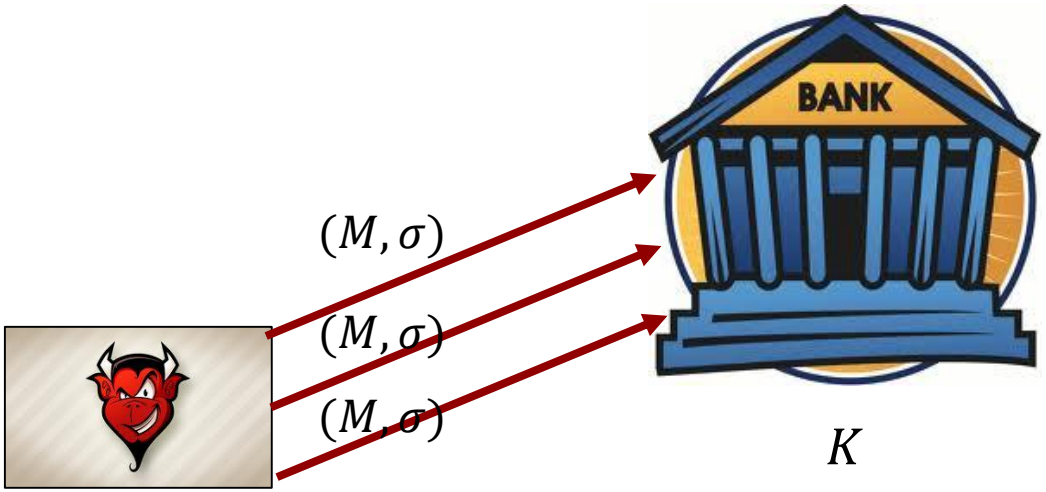
K

Replay Attack

- Suppose we have a UF-CMA secure MAC and we use it to authenticate bank transactions.
- How do we prevent such replay attacks?

$M =$ "Transfer Bob \$100", $\sigma = T_K(M)$

K



K

Replay Attack

- Suppose we have a UF-CMA secure MAC and we use it to authenticate bank transactions.
- How do we prevent such replay attacks?
  - Timestamps: Alice sends $(M||time, E_K(M||time))$. $V_K(.,.)$ also checks the time difference in addition to the tag.
  - Using counters: Alice sends $(M||ctr, E_K(M||ctr))$. The sender and receiver need to maintain a common counter.

$M =$ "Transfer Bob \$100", $\sigma = T_K(M)$

K

(M, σ)

(M, σ)

(M, σ)

K

PRFs make secure MACs

PRF as MAC

- Suppose we have a secure PRF $F: \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$ and suppose we only need to authenticate messages of size n, then consider the following MAC:
  - $T_K(M) = F_K(M)$
  - $V_K(M, \sigma) = 1 \; iff \; \sigma = F_K(M)$.

- Is the above MAC secure in the UF-CMA sense?

PRF as MAC

- Suppose we have a secure PRF $F: \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$ and suppose we only need to authenticate messages of size n, then consider the following MAC:
  - $T_K(M) = F_K(M)$
  - $V_K(M, \sigma) = 1 \; iff \; \sigma = F_K(M)$.

- Is the above MAC secure in the UF-CMA sense?
  - Yes.
  - Intuition: Random function make good MAC and F is close to a random function.

PRF as MAC

- Suppose we have a secure PRF $F: \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$ and suppose we only need to authenticate messages of size n, then consider the MAC associated with F:
  - $T_K(M) = F_K(M)$
  - $V_K(M, \sigma) = 1 \; iff \; \sigma = F_K(M)$.
- Theorem: Consider the function family F above and the associated MAC MA. Let A be a UF-CMA adversary making q_s tag-generation queries and q_v tag-verification queries with $q_v \leq 2^{n-1}$ and having a running time t. There is a PRF adversary B such that:

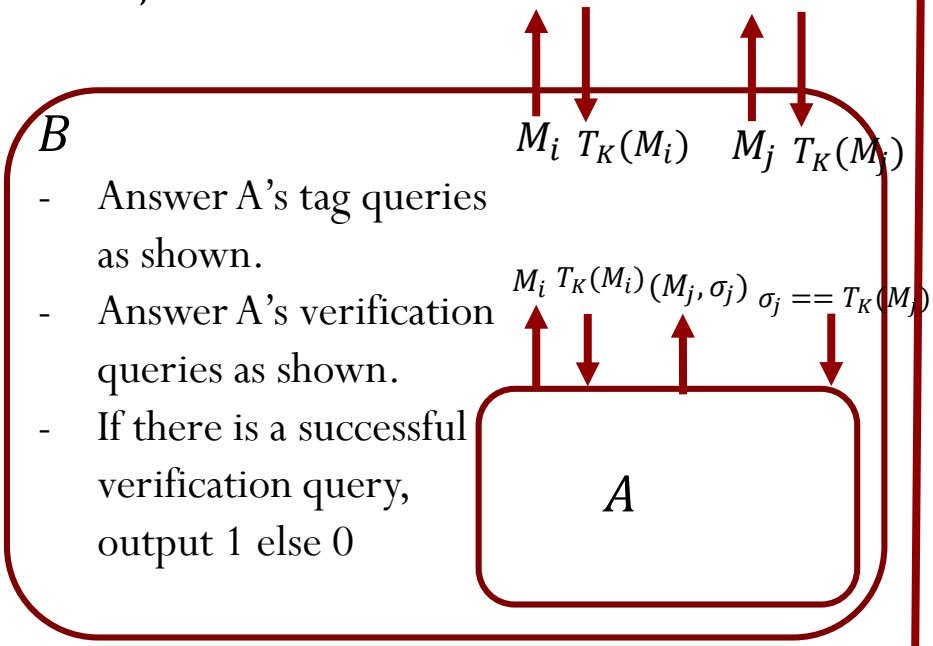
  $Adv_{uf-cma}(A, MA) \leq Adv_{PRF}(B, F) + \frac{2q_v}{2^n}$.

  Moreover, B makes $(q_s + q_v)$ queries and runs in time $t + \theta(n(q_s + q_v))$.

PRF as MAC

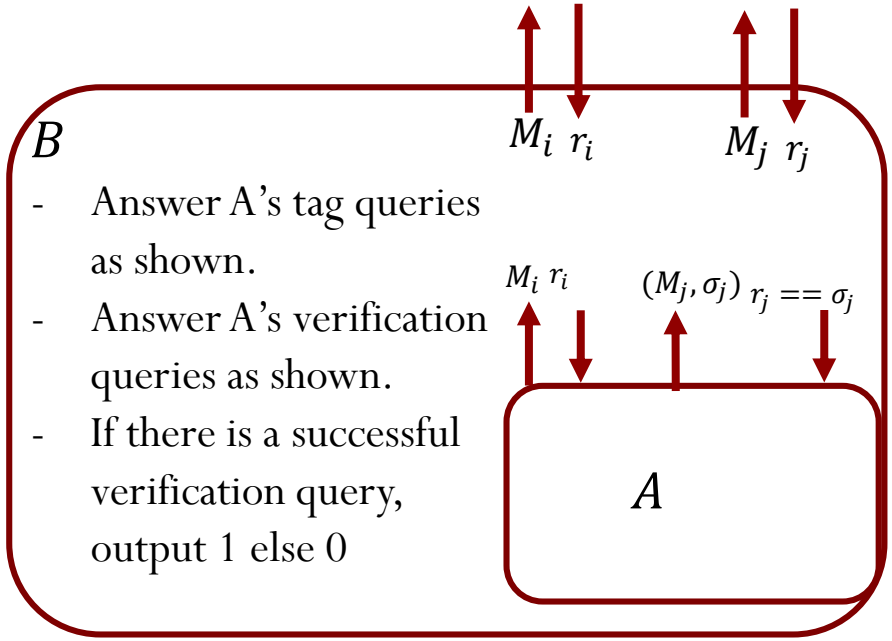
- Theorem: Consider the function family F above and the associated MAC MA. Let A be a UF-CMA adversary making q_s tag-generation queries and q_v tag-verification queries with $q_v \leq 2^{n-1}$ and having a running time t. There is a PRF adversary B such that: $Adv_{uf-cma}(A, MA) \leq Adv_{PRF}(B, F) + \frac{2q_v}{2^n}$.
  Moreover, B makes $(q_s + q_v)$ queries and runs in time $t + \theta(n(q_s + q_v))$.

$Real_{B,F}$



$\Pr[Real_{B,F} = 1] = ?$

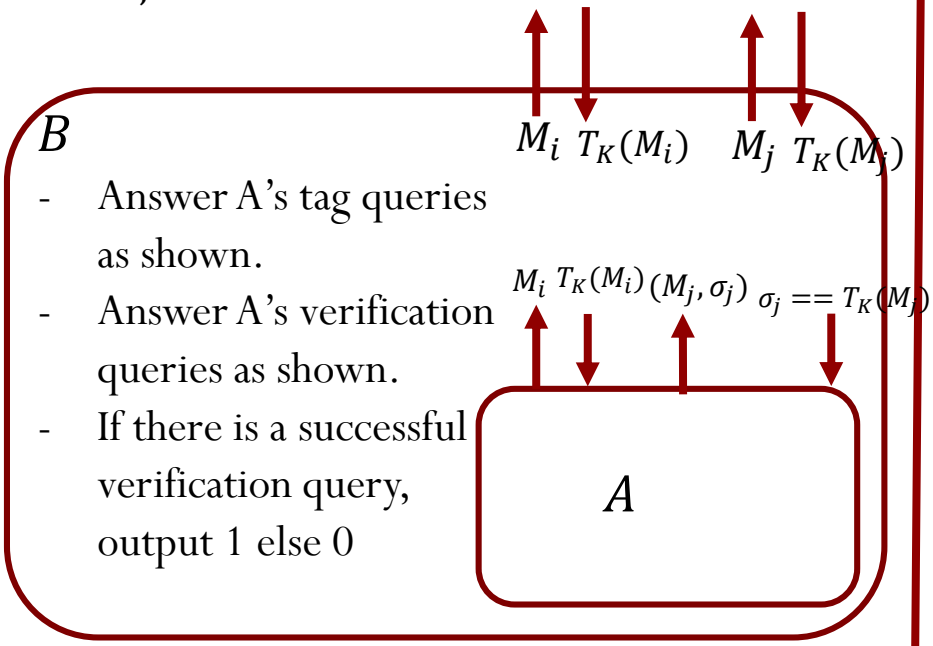
$Random_B$



PRF as MAC

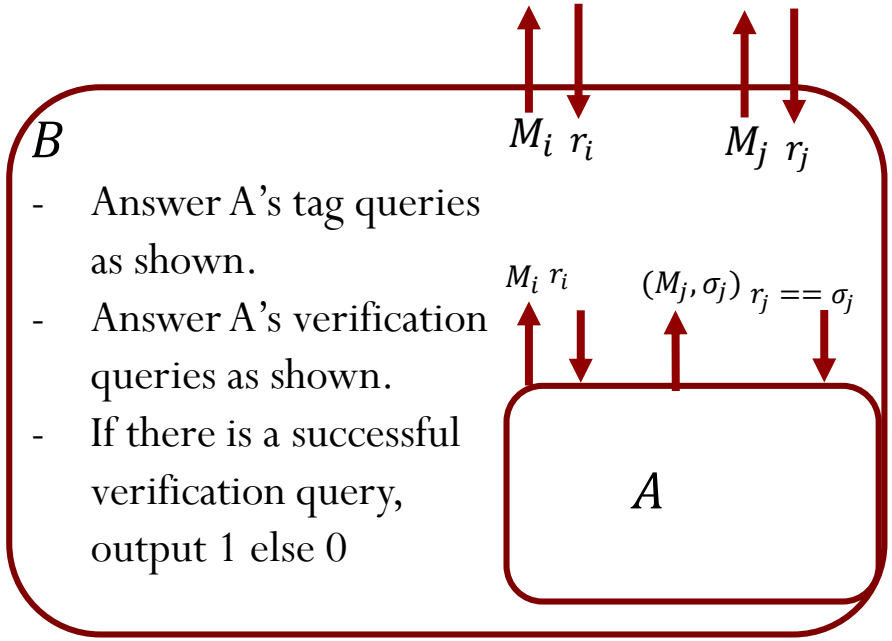
- Theorem: Consider the function family F above and the associated MAC MA. Let A be a UF-CMA adversary making q_s tag-generation queries and q_v tag-verification queries with $q_v \leq 2^{n-1}$ and having a running time t. There is a PRF adversary B such that: $Adv_{uf-cma}(A, MA) \leq Adv_{PRF}(B, F) + \frac{2q_v}{2^n}$.
  Moreover, B makes $(q_s + q_v)$ queries and runs in time $t + \theta(n(q_s + q_v))$.

$Real_{B,F}$



$\Pr[Real_{B,F} = 1] = Adv_{uf-cma}(A, MA)$

$Random_B$

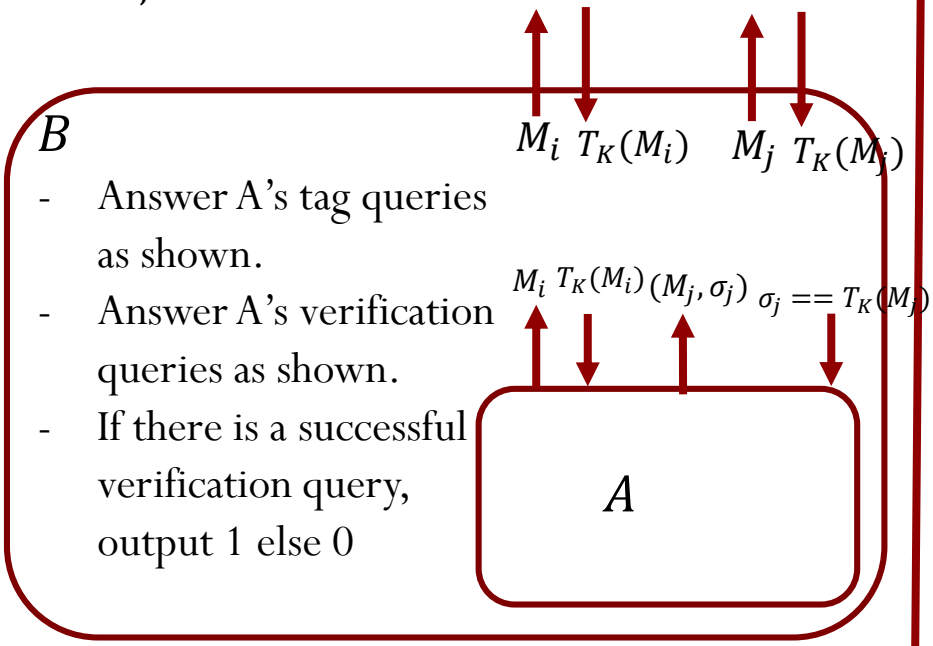


$\Pr[Random_B = 1] = ?$

PRF as MAC

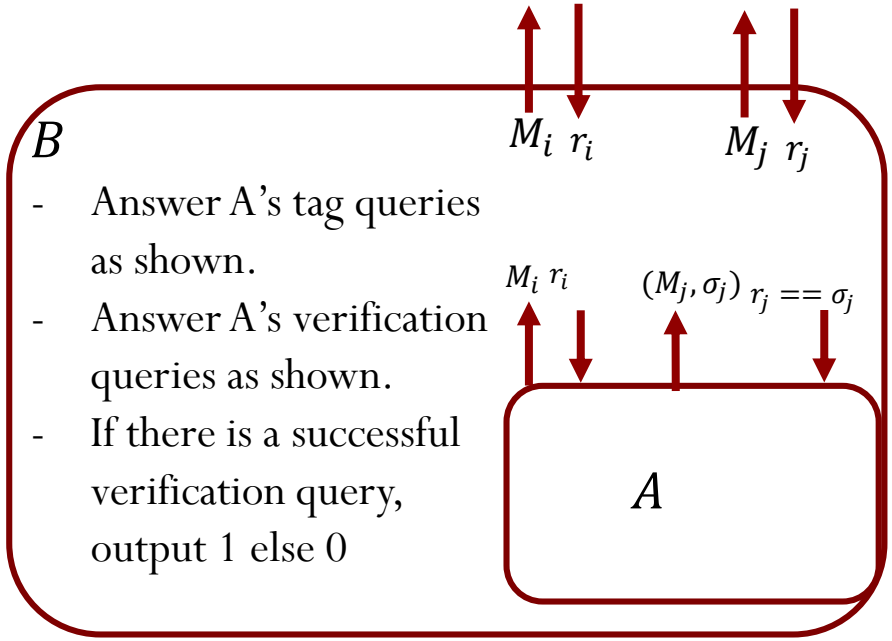
- Theorem: Consider the function family F above and the associated MAC MA. Let A be a UF-CMA adversary making q_s tag-generation queries and q_v tag-verification queries with $q_v \leq 2^{n-1}$ and having a running time t. There is a PRF adversary B such that: $Adv_{uf-cma}(A, MA) \leq Adv_{PRF}(B, F) + \frac{2q_v}{2^n}$.
  Moreover, B makes $(q_s + q_v)$ queries and runs in time $t + \theta(n(q_s + q_v))$.

$Real_{B,F}$



$\Pr[Real_{B,F} = 1] = Adv_{uf-cma}(A, MA)$

$Random_B$

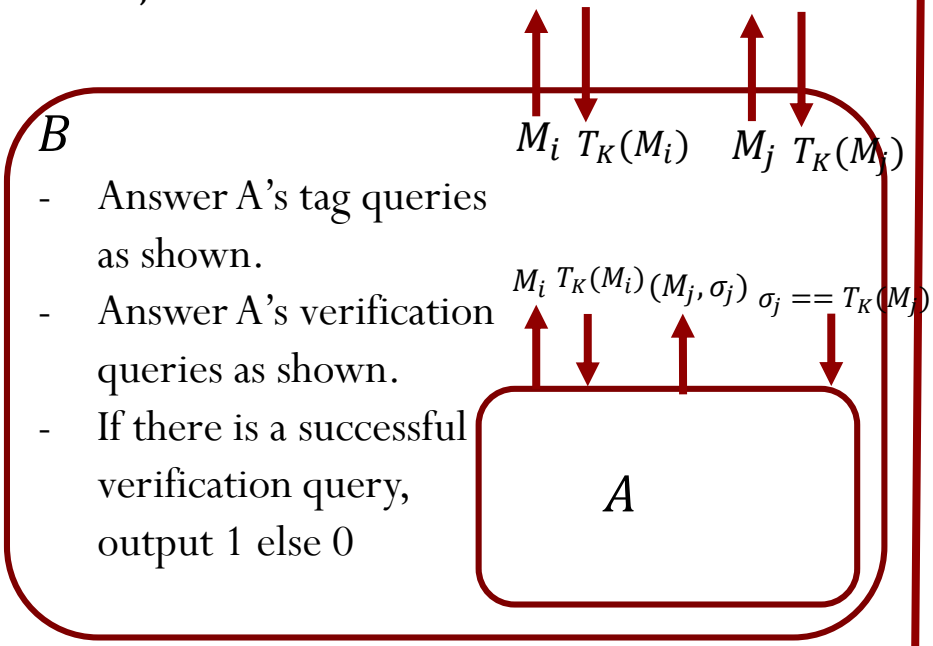


$\Pr[Random_B = 1] = \frac{1}{2^n}$, if $q_v = 1$

PRF as MAC

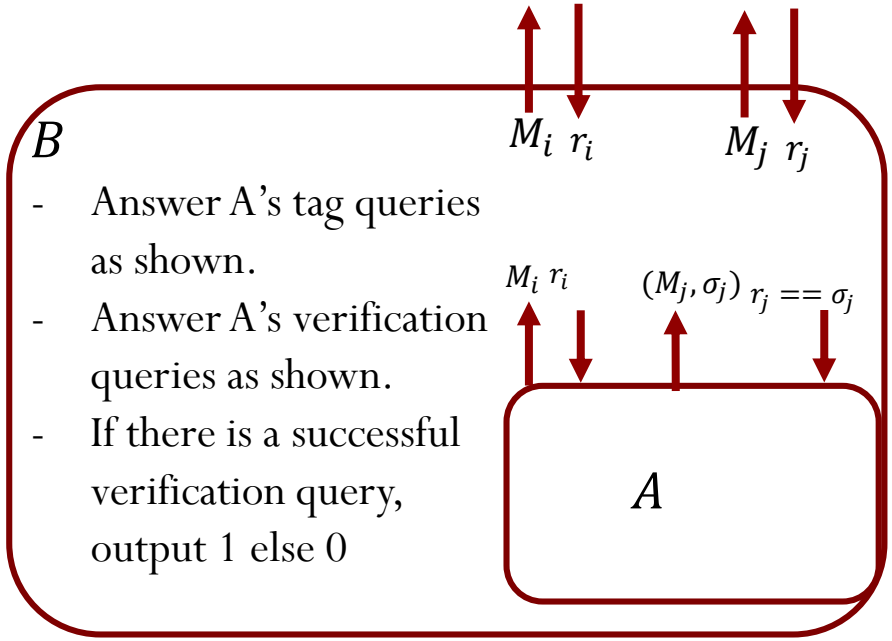
- Theorem: Consider the function family F above and the associated MAC MA. Let A be a UF-CMA adversary making q_s tag-generation queries and q_v tag-verification queries with $q_v \leq 2^{n-1}$ and having a running time t. There is a PRF adversary B such that: $Adv_{uf-cma}(A, MA) \leq Adv_{PRF}(B, F) + \frac{2q_v}{2^n}$.
  Moreover, B makes $(q_s + q_v)$ queries and runs in time $t + \theta(n(q_s + q_v))$.

$Real_{B,F}$



$\Pr[Real_{B,F} = 1] = Adv_{uf-cma}(A, MA)$

$Random_B$

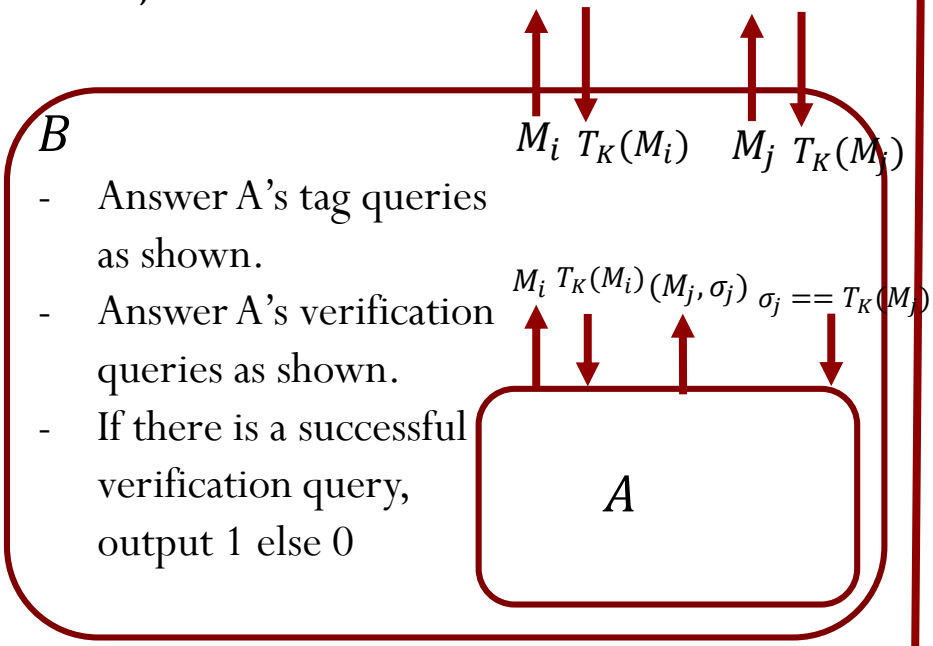


$\Pr[Random_B = 1] = \frac{1}{2^n} + \frac{1}{2^n - 1} + \cdots$

PRF as MAC

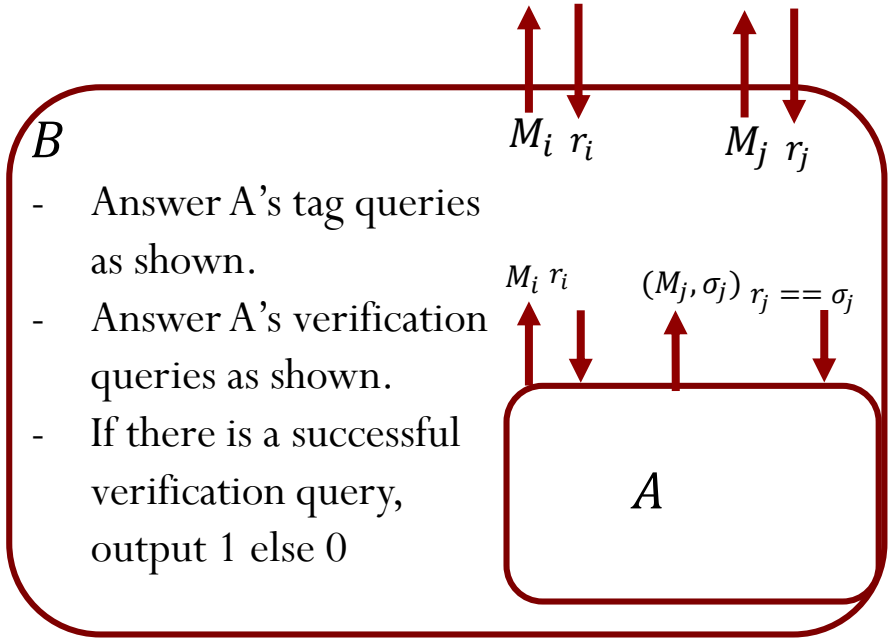
- Theorem: Consider the function family F above and the associated MAC MA. Let A be a UF-CMA adversary making q_s tag-generation queries and q_v tag-verification queries with $q_v \leq 2^{n-1}$ and having a running time t. There is a PRF adversary B such that: $Adv_{uf-cma}(A, MA) \leq Adv_{PRF}(B, F) + \frac{2q_v}{2^n}$.
  Moreover, B makes $(q_s + q_v)$ queries and runs in time $t + \theta(n(q_s + q_v))$.

$Real_{B,F}$



$\Pr[Real_{B,F} = 1] = Adv_{uf-cma}(A, MA)$

$Random_B$



$\Pr[Random_B = 1] = \frac{2q_v}{2^n}$

End