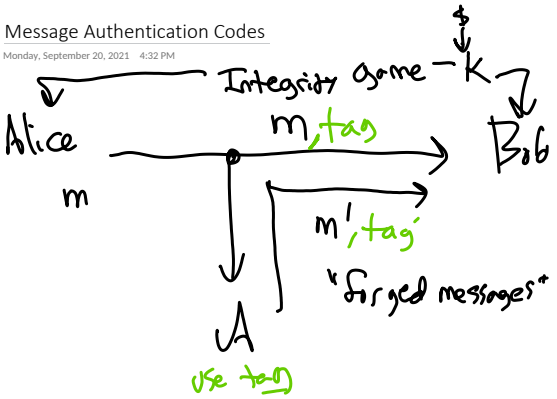
Message Authentication Codes

Monday, September 20, 2021 4:32 PM



"Bob can check if message m was actually sent by Alice (and reject m' sent by A but not sent by Alice)"

MAC syntax: $Gen(1^\lambda) \to k \quad m \in \mathcal{M}$

$Tag(k, m) \to t$

$Check(k, m, t) \to \{0, 1\}$

Simple MAC:

$F: K \times \mathcal{M} \to \{0,1\}^\lambda$

Unforgeability:

(First broken attempt):

$$\forall \underbrace{m_1, \ldots, m_q}_{\text{honest messages}} . \Pr\left[\begin{array}{c} k \leftarrow Gen() \\ \text{for each } m_i: \\ tag_i \leftarrow Tag(k, m_i) \\ \text{"A output a forgery"} \\ (m', t') \leftarrow A(1^\lambda, \{m_i\}, \{tag_i\}) \\ m' \notin \{m_i\} \wedge Check(k, m', t') = 1 \end{array}\right] \leq negl(\lambda)$$

Improved adaptive:

$$\forall A \quad \Pr\left[\begin{array}{c} k \leftarrow Gen() \\ (m', t') \leftarrow A^{Tag(k, \cdot)} \end{array}\right]$$

$m' \notin$ queries made by $\mathcal{A}$]
$\wedge$ Check$(k, m, t) = 1$

Proof sketch: - Can substitute PRF for a real random function.

- With real random function, chance of guessing t' for a new message m' is $1/2^\lambda$

Still to do:

- Long messages.
- mac & encrypt vs mac then encrypt vs encrypt then mac