Message Authentication Code

Scribes: Igor Aronov

Recap from previous lecture.

Security of MAC:
- Key Recovery (KR)
- Existential forgery against adaptive chosen message (CMA).

If our message $m$ is $l$ bits long, and we have $PRF_k(m)$ construction then this PRF-MAC construction is CMA secure.

This construction will produce following MAC: $(m_1, m_2, \ldots, m_n, \mu_1, \mu_2, \ldots, \mu_n)$, but this is not CMA secure.

**Theorem**: CBC-MAC is CMA secure, if $F_k$ is a PRF, then CBC-MAC is CMA secure.

**Proof**: We’ll prove the contrapositive $\neg$ CBC-MAC CMA $\Rightarrow \neg$ $F_k$-PRF.

If exist an adversary $A$ who breaks CMA security of CBC-MAC$m$ then we can construct $B$ which breaks PRF property of $F_k$. 

\[
\text{Adv}^{\text{PRF}}(B) = \Pr(\text{Exp}^{\text{PRF}}_{\text{P}}(B) = 1) - \Pr(\text{Exp}^{\text{PRF}}_{\text{O}}(B) = 1) = \\
= \Pr(d = 1| F_k) - \Pr(d = 1| RF) = \\
= \text{Adv}_{\text{CBC-MAC}}^{\text{CMA}}(A) - \Pr(d = 1| RF) = \\
\text{(I)}
\]

Note that, if the Ith block input for qth query collides with jth block for pth query, then
\[
M_{i-1}^{(q)} \oplus m_{i}^{(q)} = \mu_{i-1}^{(p)} \oplus m_{j}^{(p)} = \\
\mu_{i-1}^{(q)} \oplus m_{j}^{(q)} = \mu_{i-1}^{(p)} \oplus m_{i}^{(q)} = >
\]

q query \(m_{1}^{(q)}, m_{2}^{(q)}, \ldots, m_{i}^{(q)}, \ldots, m_{n}^{(q)}\)

swapping \(m_{i}^{(q)}\) and \(m_{j}^{(p)}\) created forgery

p query \(m_{1}^{(p)}, m_{2}^{(p)}, \ldots, m_{j}^{(p)}, \ldots, m_{n}^{(p)}\)

Continue with (I)
\[
\text{Adv}^{\text{PRF}}(B) = \text{Adv}_{\text{CBC-MAC}}^{\text{CMA}}(A) - \frac{Qn(Qn-1)}{2} \cdot 2^{L} = \\
\text{Adv}_{\text{CBC-MAC}}^{\text{CMA}}(A) = \text{Adv}^{\text{PRF}}(B) + \frac{Qn(Qn-1)}{2} \cdot 2^{L}
\]

**Proposition:** CBC-MAC works with fixed length messages.

**Proof:**
We replace $F_k$ in CBC-MAC construction with $F_k' = F_{F_k(l)}$.

**Fix:** $k' = F_k(l)$

We now attempt to construct MAC based on hash function.

- $MAC_k(m) = H(m \oplus k)$. This construction is not secure, for example we can create forgery in the following manner: If output of our hash function is 128 bits long, then we create two messages, where in first message first 128 bits are all 0 and then its followed by arbitrary number of 0, and second message contains 1’s as first 128 bits and has same number of trailing 0 as the first message.

- $MAC_k(m) = H(k.m)$, here we just concatenate key $k$ and message. This construction is not secure as well. For example, we could append to some arbitrary message $m'$ to $H(k.m)$, and will create valid forgery for $MAC(m.m')$

- $MAC_k(m) = H(m.k)$, here we concatenate message with key $k$, but this is still not secure if we are able to create collision on function $H$.

- NMAC (Nested MAC construction)

$$NMAC_{k_1,k_2}(m) = h_{k_1}(H_{k_2}(m))$$
**Theorem:** NMAC is CMA secure, if \( H^* \) is collision resistant and \( h^* \) is itself CMA secure MAC.

**Proof:** If NMAC is not CMA secure then \( h^* \) is not CMA secure.

\[
\text{Adv}_{h^*}^{\text{CMA}}(B) = \Pr(d = 1) = 1 - \Pr(d = 0) = \\
= 1 - \left[ \Pr(A \text{ fails}) + \Pr(\text{collision on } H^*) \right] = \\
= 1 - [1 - \Pr(A \text{ wins}) + \Pr(\text{collision on } H^*)] = \\
= \text{Adv}_{\text{NMAC}}^{\text{CMA}}(A) = \text{Adv}_{h^*}^{\text{CMA}}(B) + \Pr(H^* \text{ has collision)} = \]

if \( H^* \) is collision resistant then \( \Pr(H^* \text{ has collision}) \) is negligible, and if \( h^* \) is CMA then \( \text{Adv}_{h^*}^{\text{CMA}}(B) \) is also negligible.