Announcements

• HW4 due today
• Optional HW5 up on course webpage, due 3/13
• Midterm coming up in class on 3/15
• Midterm Review Sheet up on course webpage
• There will be a cheat sheet on the midterm. Cheat sheet will be posted on Canvas.
Agenda

• Last time:
  – Block Ciphers (K/L 3.5)
  – Modes of Operation (K/L 3.6)
  – Started MAC (K/L 4.2)

• This time:
  – Security Definition for MAC (K/L 4.2)
  – Constructing MAC from PRF (K/L 4.3)
  – Domain Extension for MACs (K/L 4.4)
Message Authentication Codes

Definition: A message authentication code (MAC) consists of three probabilistic polynomial-time algorithms $(Gen, Mac, Vrfy)$ such that:

1. The key-generation algorithm $Gen$ takes as input the security parameter $1^n$ and outputs a key $k$ with $|k| \geq n$.
2. The tag-generation algorithm $Mac$ takes as input a key $k$ and a message $m \in \{0,1\}^*$, and outputs a tag $t$.
   \[ t \leftarrow Mac_k(m). \]
3. The deterministic verification algorithm $Vrfy$ takes as input a key $k$, a message $m$, and a tag $t$. It outputs a bit $b$ with $b = 1$ meaning valid and $b = 0$ meaning invalid.
   \[ b := Vrfy_k(m, t). \]

It is required that for every $n$, every key $k$ output by $Gen(1^n)$, and every $m \in \{0,1\}^*$, it holds that $Vrfy_k(m, Mac_k(m)) = 1$. 
Security of MACs

The message authentication experiment $MAC_{\text{forge}}_{A, \Pi}(n)$:

1. A key $k$ is generated by running $Gen(1^n)$.
2. The adversary $A$ is given input $1^n$ and oracle access to $Mac_k(\cdot)$. The adversary eventually outputs $(m, t)$. Let $Q$ denote the set of all queries that $A$ asked its oracle.
3. $A$ succeeds if and only if (1) $Vrfy_k(m, t) = 1$ and (2) $m \notin Q$. In that case, the output of the experiment is defined to be 1.
Security of MACs

Definition: A message authentication code $\Pi = (Gen, Mac, Vrfy)$ is existentially unforgeable under an adaptive chosen message attack if for all probabilistic polynomial-time adversaries $A$, there is a negligible function $neg$ such that:

$$\Pr[MAC_{forge_A,\Pi}(n) = 1] \leq neg(n).$$
Strong MACs

The strong message authentication experiment $MAC_{\text{forge}}_{A,\Pi}(n)$:

1. A key $k$ is generated by running $Gen(1^n)$.
2. The adversary $A$ is given input $1^n$ and oracle access to $Mac_k(\cdot)$. The adversary eventually outputs $(m, t)$. Let $Q$ denote the set of all pairs $(m, t)$ that $A$ asked its oracle.
3. $A$ succeeds if and only if (1) $Vrfy_k(m, t) = 1$ and (2) $(m, t) \notin Q$. In that case, the output of the experiment is defined to be 1.
Strong MACs

Definition: A message authentication code \( \Pi = (Gen, Mac, Vrfy) \) is a strong MAC if for all probabilistic polynomial-time adversaries \( A \), there is a negligible function \( neg \) such that:

\[
Pr[MACsforge_{A,\Pi}(n) = 1] \leq neg(n).
\]
Constructing Secure Message Authentication Codes
A Fixed-Length MAC

Let $F$ be a pseudorandom function. Define a fixed-length MAC for messages of length $n$ as follows:

- **Mac**: on input a key $k \in \{0,1\}^n$ and a message $m \in \{0,1\}^n$, output the tag $t := F_k(m)$.

- **Vrfy**: on input a key $k \in \{0,1\}^n$, a message $m \in \{0,1\}^n$, and a tag $t \in \{0,1\}^n$, output $1$ if and only if $t = F_k(m)$. 
Security Analysis

Theorem: If $F$ is a pseudorandom function, then the construction above is a secure fixed-length MAC for messages of length $n$. 

Let $A$ be a ppt adversary trying to break the security of the construction. We construct a distinguisher $D$ that uses $A$ as a subroutine to break the security of the PRF.

Distinguisher $D$:

1. When $A$ queries its oracle with message $m$, output $O(m)$.
2. Eventually, $A$ outputs $(m^*, t^*)$ where $m^*, t^* \in \{0,1\}^n$.
3. If $m^* \in Q$, output 0.
4. If $m^* \notin Q$, query $O(m^*)$ to obtain output $z^*$.
5. If $t^* = z^*$ output 1. Otherwise, output 0.
Security Analysis

Consider the probability $D$ outputs 1 in the case that $O$ is truly random function $f$ vs. $O$ is a pseudorandom function $F_k$.

- When $O$ is pseudorandom, $D$ outputs 1 with probability $\Pr[MAC\text{forge}_{A,\Pi}(n) = 1] = \rho(n)$, where $\rho$ is non-negligible.
- When $O$ is random, $D$ outputs 1 with probability at most $\frac{1}{2^n}$. Why?
Security Analysis

$D$’s distinguishing probability is:

$$\left| \frac{1}{2^n} - \rho(n) \right| = \rho(n) - \frac{1}{2^n}.$$ 

Since, $\frac{1}{2^n}$ is negligible and $\rho(n)$ is non-negligible, $\rho(n) - \frac{1}{2^n}$ is non-negligible.

This is a contradiction to the security of the PRF.
Domain Extension for MACs
Let $F$ be a pseudorandom function, and fix a length function $\ell$. The basic CBC-MAC construction is as follows:

- **Mac**: on input a key $k \in \{0,1\}^n$ and a message $m$ of length $\ell(n) \cdot n$, do the following:
  1. Parse $m$ as $m = m_1, \ldots, m_\ell$ where each $m_i$ is of length $n$.
  2. Set $t_0 := 0^n$. Then, for $i = 1$ to $\ell$:
      - Set $t_i := F_k(t_{i-1} \oplus m_i)$.
  Output $t_\ell$ as the tag.

- **Vrfy**: on input a key $k \in \{0,1\}^n$, a message $m$, and a tag $t$, do: If $m$ is not of length $\ell(n) \cdot n$ then output 0. Otherwise, output 1 if and only if $t = Mac_k(m)$. 
CBC-MAC

FIGURE 4.1: Basic CBC-MAC (for fixed-length messages).