DD2448 Foundations of Cryptography
Lecture 6

Douglas Wikström
KTH Royal Institute of Technology
dog@kth.se

March 27, 2020

Last Lecture: Merkle-Damgård (1/3)

Suppose that we are given a collision resistant hash function

$$f : \{0,1\}^{n+t} \rightarrow \{0,1\}^n \ .$$

How can we construct a collision resistant hash function

$$h : \{0,1\}^* \rightarrow \{0,1\}^n$$

mapping any length inputs?

Last Lecture: Merkle-Damgård (2/3)

Construction.

1. Let $x = (x_1, \ldots, x_k)$ with $|x_i| = t$ and $0 < |x_k| \leq t$.
2. Let x_{k+1} be the total number of bits in x.
3. Pad x_k with zeros until it has length t.
4. $y_0 = 0^n$, $y_i = f(y_{i-1}, x_i)$ for $i = 1, \ldots, k+1$.
5. Output y_{k+1}

Here the total number of bits is bounded by $2^t - 1$, but this can be relaxed.

Merkle-Damgård (3/3)

Suppose A finds collisions in Merkle-Damgård.

- If the number of bits differ in a collision, then we can derive a collision from the last invocation of f.
- If not, then we move backwards until we get a collision. Since both inputs have the same length, we are guaranteed to find a collision.

Standardized Hash Functions

Standardized Hash Functions

Despite that theory says it is impossible, in practice people simply live with **fixed** hash functions and use them as if they are randomly chosen functions.

SHA

- Secure Hash Algorithm (SHA-0,1, and the SHA-2 family) are hash functions standardized by NIST to be used in, e.g., signature schemes and random number generation.
- SHA-0 was **weak** and withdrawn by NIST. SHA-1 was **withdrawn** 2010. SHA-2 family is based on similar ideas but seems safe so far...
- All are **iterated** hash functions, starting from a basic **compression function**.

SHA-3

- NIST ran an open competition for the next hash function, named SHA-3. Several groups of famous researchers submitted proposals.
- Call for SHA-3 explicitly asked for “different” hash functions.
- It might be a good idea to read about SHA-1 for comparison.
- The competition ended October 2, 2012, and the hash function **Keccak was selected as the winner**.
- This was constructed by Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche,

MACs

Message Authentication Code

- Message Authentication Codes (MACs) are used to ensure integrity and authenticity of messages.

Message Authentication Code

- Message Authentication Codes (MACs) are used to ensure integrity and authenticity of messages.
- Scenario:
  1. Alice and Bob share a common key k.
  2. Alice computes an authentication tag $\alpha = \mathsf{MAC}_\mathsf{k}(m)$ and sends (m, α) to Bob.
  3. Bob receives (m', α') from Alice, but before accepting m' as coming from Alice, Bob checks that $\mathsf{MAC}_\mathsf{k}(m') = \alpha'$.

Security of a MAC

Definition. A message authentication code MAC is secure if for a random key k and every polynomial time algorithm A,

$$\Pr[A^{\mathsf{MAC}_{\mathsf{k}}(\cdot)} = (m, \alpha) \wedge \mathsf{MAC}_{\mathsf{k}}(m) = \alpha \wedge \forall i : m \neq m_i]$$

is negligible, where m_i is the ith query to the oracle $\mathsf{MAC}_{\mathsf{k}}(\cdot)$.

Random Oracle As MAC

- Suppose that $H : \{0,1\}^* \rightarrow \{0,1\}^n$ is a random oracle.
- Then we can construct a MAC as $\mathrm{MAC}_\mathrm{k}(m) = H(\mathrm{k}, m)$.

Could we plug in an iterated hash function in place of the random oracle?

HMAC

- Let $H : \{0,1\}^* \rightarrow \{0,1\}^n$ be a "cryptographic hashfunction", e.g., SHA-256.
- $\mathsf{HMAC}_{\mathsf{k}_1,\mathsf{k}_2}(x) = H\big(\mathsf{k}_2 \| H(\mathsf{k}_1 \| x)\big)$
- This is provably secure under the assumption that
  - $H(\mathsf{k}_1 \| \cdot)$ is unknown-key collision resistant, and
  - $H(\mathsf{k}_2 \| \cdot)$ is a secure MAC for fixed-size messages.

CBC-MAC

Let E be a secure block-cipher, and $x = (x_1, \ldots, x_t)$ an input. The MAC-key is simply the block-cipher key.

1. $y_0 = 000\ldots0$
2. For $i = 1, \ldots, t$, $y_i = \mathrm{E_k}(y_{i-1} \oplus x_i)$
3. Return y_t.

Is this secure?

Universal Hashfunction As MAC

Theorem. A t-universal hashfunction f_α for a randomly chosen secret α is an **unconditionally secure** MAC, provided that the number queries is smaller than t.