CS 4501-Cryptography, Homework 5

Response by: Your Name, (computing id)

Total points: 30. Points are noted after each problem.

Directions. For each problem, typeset your solution in the `answer` environment, and if there are sub-problems, mark them clearly. Feel free to use as much space as you want to. Before you submit the PDF, update 1) "Your Name" and id on the above, and 2) the "Acknowledgement" box at the last page properly.

Policies. We re-iterate our policy. It is encouraged to think and discuss the problems before looking for ready-made solutions. You shall acknowledge and/or reference any discussion and published material except for lecture notes and resources of LaTeX. In any case, it is a violation if any of the following cases happens:

- You copied text directly (from any source).
- You used any material or discussion without acknowledgement or citation.
- You are unable to explain your work orally.

This homework reviews MAC with an emphasis on the composition. That is important when we want to authenticate longer messages (which is extremely common and has failed in real-world applications).

Problem 1 (4pt)**.** A MAC $(\mathsf{Gen}, \mathsf{Tag}, \mathsf{Ver})$ is *deterministic* if the Tag algorithm is deterministic (so that for fixed m, k, the output $\mathsf{Tag}_k(m)$ is always the same). Given any deterministic MAC, we may view Tag as a keyed function. In the simple MAC from pseudorandom function [Katz-Lindel, Construction 4.5], Tag is a pseudorandom function. Give a construction of a secure, deterministic MAC in which Tag is not a pseudorandom function.

Answer.

□

Problem 2 (3pt each)**.** Let $f = \{f_k : \{0,1\}^n \to \{0,1\}^n, n = |k|, k \in \{0,1\}^*\}$ be a pseudorandom function. Show that each of the following MACs is insecure even for fixed length messages (that is, ℓ is a fixed number and is known by the verifier). The verification is performed accordingly, i.e., accepts if and only if $\mathsf{Tag}(m) = t$.

Notation: In each case Gen outputs a uniform $k \leftarrow \{0,1\}^n$; we let $\langle i \rangle$ denote an $n/2$-bit encoding of the integer i. Each $m_i \in \{0,1\}^n$ is an n-bit string.

(a) To authenticate a message $m = m_1, m_2$, compute $t := f_k(m_1) \| f_k(m_2)$.

(b) To authenticate a message $m = m_1, m_2$, compute $t := f_k(m_1) \| f_k(f_k(m_2))$.

(c) To authenticate a message $m = m_1, ..., m_\ell$, compute $t := f_k(m_1) \oplus \cdots \oplus f_k(m_\ell)$.

(d) To authenticate a message $m = m_1, ..., m_\ell$, compute $t := f_k(\langle 1 \rangle \| m_1) \oplus \cdots \oplus f_k(\langle \ell \rangle \| m_\ell)$.

Answer.

(a)

(b)

(c)

(d)

□

Problem 3 (4pt each)**.** Read [Katz-Lindel, Section 4.4], in particular "4.4.1 The Basic Construction" and Construction 4.9. Answer the following questions of CBC-MAC.

We consider what happens when the basic CBC-MAC construction is used with messages of different lengths.

(a) Say the sender and receiver do not agree on the message length in advance (and so $\mathsf{Ver}_k((m,t) = 1$ iff $t = \mathsf{Tag}_k(m)$, regardless of the length of m), but the sender is careful to only authenticate messages of 2 blocks. Show that an adversary can forge a valid tag on a message of 4 blocks.

(b) Say the receiver only accepts 3-block messages (so $\mathsf{Ver}_k(m,t) = 1$ only if m has length $3n$ and $t = \mathsf{Tag}_k(m)$), but the sender authenticates messages of any length a multiple of n. Show that an adversary can forge a valid tag on a new message.

Answer.

(a)

(b)

□

Problem 4 (6pt)**.** Given a computable function Com, some polynomial $l(\cdot)$, the definition of perfect binding and hiding are

1. Perfect binding: For all $n \in \mathbb{N}$ and all $v_0, v_1 \in \{0,1\}^n$ such that $v_0 \neq v_1$, $r_0, r_1 \in \{0,1\}^{l(n)}$, it holds that $\mathsf{Com}(v_0, r_0) \neq \mathsf{Com}(v_1, r_1)$.
2. Perfect hiding: For all unbounded distinguisher D, for all $n \in \mathbb{N}$, $v_0, v_1 \in \{0,1\}^n$,

$$\Pr[r \leftarrow \{0,1\}^{l(n)} : D(\mathsf{Com}(v_0, r)) = 1] = \Pr[r \leftarrow \{0,1\}^{l(n)} : D(\mathsf{Com}(v_1, r)) = 1]$$

Prove that there do not exist a commitment scheme Com that satisfies both perfect binding and perfect hiding.

Answer. □

Acknowledgement

Replace this with your collaborators and resources like below (if you did not have any, replace this with *None*).

Problem 1 is inspired by a discussion with Jonathan, who explained Extended Euclidean Algorithm to me.

Problem 5: I used ChatGPT with the prompt "teach me DMT2" and obtained XXX.