CS442: Introduction to Cryptography Instructor: Aarushi Goel

Homework 4

Due: April 19; 2026 (11:59 PM)

1 Authenticated Encryption

(10 points) Let $(\mathsf{KeyGen}_1, \mathsf{Enc}, \mathsf{Dec})$ be a **CPA-secure encryption scheme** and $(\mathsf{KeyGen}_2, \mathsf{Sign}, \mathsf{Verify})$ be a **many-time strongly unforgeable MAC scheme**. Recall the following **Encrypt-then-MAC** approach for authenticated encryption discussed in Lecture 17:

- $\mathsf{KeyGen} \to k$: Sample $k_1 \xleftarrow{\$} \mathsf{KeyGen}_1$ and $k_2 \xleftarrow{\$} \mathsf{KeyGen}_2$. Output $k = (k_1, k_2)$.
- $\mathsf{AuthEnc}(k, m) \to C$: Parse $k = (k_1, k_2)$. Compute $c = \mathsf{Enc}(k_1, m)$, $\sigma = \mathsf{Sign}(k_2, c)$ and output $C = (c, \sigma)$.
- $\mathsf{AuthDec}(k, C) \to m$: Parse $k = (k_1, k_2)$ and $C = (c, \sigma)$. Compute $m = \mathsf{Dec}(k_1, c)$ and $b \leftarrow \mathsf{Verify}(k_2, c, \sigma)$. If $b = 1$, output m, else output $\perp$.

Explain why the above scheme necessarily requires $(\mathsf{KeyGen}_2, \mathsf{Sign}, \mathsf{Verify})$ to be many-time **strongly** unforgeable. In other words, explain why the scheme may fail to satisfy authenticated-encryption security if $(\mathsf{KeyGen}_2, \mathsf{Sign}, \mathsf{Verify})$ is only many-time unforgeable, but **NOT strongly** unforgeable.

2 Collision-Resistant Hash Functions

Determine whether the following are collision-resistant hash functions. If you believe it is collision-resistant, provide an explanation as discussed in Lecture 19. If you believe it is not collision-resistant, give an example of a collision.

1. **(10 points)** Let $\left\{h_0^k : \{0,1\}^* \to \{0,1\}^n\right\}_{k \in \mathcal{K}}$ and $\left\{h_1^k : \{0,1\}^* \to \{0,1\}^n\right\}_{k \in \mathcal{K}}$ be two families of collision-resistant hash functions. Consider the following function $\left\{H^k : \{0,1\}^* \to \{0,1\}^n\right\}_{k \in \mathcal{K}}$,
$$H^k(x) = h^k_{x_1 \oplus x_2}(x_3 \| \dots \| x_L),$$
where $x = x_1 \| x_2 \| x_3 \| \dots \| x_L$ and each $x_i \in \{0,1\}$.
2. **(10 points)** Let $\left\{h_0^k : \{0,1\}^* \to \{0,1\}^n\right\}_{k \in \mathcal{K}}$ and $\left\{h_1^k : \{0,1\}^* \to \{0,1\}^n\right\}_{k \in \mathcal{K}}$ be two families of collision-resistant hash functions. Consider the following function $\left\{H^k : \{0,1\}^* \to \{0,1\}^n\right\}_{k \in \mathcal{K}}$,
$$H^k(x) = h_0^k(x_1) \| h_1^k(x_2),$$
where $x = x_1 \| x_2$ and $|x_1| = |x_2|$.
3. **(10 points)** Let $\left\{h^k : \{0,1\}^* \to \{0,1\}^n\right\}_{k \in \mathcal{K}}$ be a family of collision-resistant hash functions. Consider the following function $\left\{H^k : \{0,1\}^* \to \{0,1\}^{n-1}\right\}_{k \in \mathcal{K}}$:
$$H^k(x) = y_1 \| y_2 \| \dots \| y_{n-1},$$
where $y_1 \| y_2 \| \dots \| y_{n-1} \| y_n = h^k(x)$.

3 MACs and Hash

(10 points) Let $\left\{h^k : \{0,1\}^{2n} \to \{0,1\}^n\right\}_{k\in\mathcal{K}}$ be a family of collision-resistant hash functions and $\left\{H^k : \{0,1\}^* \to \{0,1\}^n\right\}_{k\in\mathcal{K}}$ be a family of collision-resistant hash functions obtained via a Merkle-Damgård transformation on $\left\{h^k : \{0,1\}^{2n} \to \{0,1\}^n\right\}_{k\in\mathcal{K}}$. Let $k \xleftarrow{\$} \mathcal{K}$, consider the following message authentication code scheme for messages $m \in \{0,1\}^*$.

- $\mathsf{KeyGen} \to K$: Sample a key $K \xleftarrow{\$} \{0,1\}^n$.
- $\mathsf{Sign}(K, m) \to \sigma$: Compute and output $\sigma = H^k(m\|K)$.
- $\mathsf{Verify}(K, m, \sigma) \to b$: Check if $H^k(m\|K)$ equals σ. Output $b = 1$ if equal, and $b = 0$ otherwise.

Determine whether this is a many-time unforgeable message authentication code (MAC) scheme). If you believe it is, provide an explanation. If you believe it it not, show a concrete attack.

4 Key-Exchange

(10 points) Consider an adversary who can not only eavesdrop, but also actively tamper (i.e., modify) messages exchanged between Alice and Bob during the Diffie–Hellman key exchange protocol. Describe an attack strategy that allows the adversary to cause Alice and Bob to output different keys k_A and k_B (i.e., $k_A \neq k_B$), while preventing both parties from detecting any abnormal behavior.