CS 276: Homework 4

Due Date: Sunday, Apr 19, 2026 at 8:59pm via Gradescope

Usage of LLMs/Generative AI tools is prohibited. Other online resources (textbooks/lecture notes) are permissible.

1. Recall that an IBE scheme $\mathcal{E}_{id} = (G, K, E, D)$ where:

   - $G(1^n)$: setup; outputs master public key mpk and master secret key msk.
   - $K(msk, id)$: keygen; outputs secret key sk_{id} for identity id.
   - $E(mpk, id, m)$: encrypts m under identity id, outputs ciphertext c.
   - $D(sk_{id}, c)$: decrypts; outputs m or $\perp$.

   and satisfies correctness and IBE-CPA security (as defined in the notes). Consider a weaker version IBE-CPA* without any key queries as follows:

   Game $\mathbf{Exp}^{\textbf{IBE-CPA*}}_{\Pi,\mathcal{A}}(n)$

   (a) $\mathcal{A}$ outputs challenge identity id^*.
   (b) Challenger runs $(mpk, msk) \leftarrow G(1^n)$, sends mpk to $\mathcal{A}$.
   (c) $\mathcal{A}$ outputs messages (m_0, m_1).
   (d) Challenger samples $b \leftarrow \{0,1\}$, computes $c^* \leftarrow E(mpk, id^*, m_b)$, sends c^* to $\mathcal{A}$.
   (e) $\mathcal{A}$ outputs b'; output 1 if $b' = b$, else 0.

   Given a IND-CPA secure public key encryption scheme, construct an IBE scheme that satisfies IBE-CPA* security but not IBE-CPA security.

   Solution Let $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be an IND-CPA secure public-key encryption scheme. We construct an IBE scheme $\mathcal{E}_{id} = (G, K, E, D)$ as follows:

   - $G(1^n)$: Run $(pk, sk) \leftarrow \mathsf{Gen}(1^n)$. Set $mpk := pk$ and $msk := sk$. Output (mpk, msk).
   - $K(msk, id)$: *Ignore* the identity id. Output $sk_{id} := msk$ (i.e. the same secret key for every identity).
   - $E(mpk, id, m)$: *Ignore* id. Output $c \leftarrow \mathsf{Enc}(mpk, m)$.
   - $D(sk_{id}, c)$: Output $\mathsf{Dec}(sk_{id}, c)$ (since $sk_{id} = sk$, this correctly decrypts).

   Correctness. For every identity id and message m, we have $sk_{id} = sk$, so $D(sk_{id}, E(mpk, id, m)) = \mathsf{Dec}(sk, \mathsf{Enc}(pk, m)) = m$ by the correctness of Π.

   IBE-CPA* security. In the IBE-CPA* game the adversary $\mathcal{A}$ makes *no key queries*. It receives $mpk = pk$ and a challenge ciphertext $c^* \leftarrow \mathsf{Enc}(pk, m_b)$. This is exactly the IND-CPA game for Π. Hence any IBE-CPA* adversary yields an IND-CPA adversary with the same advantage, so

   $$\Pr[\mathrm{Exp}^{\text{IBE-CPA*}}_{\mathcal{E}_{id},\mathcal{A}}(n) = 1] - \frac{1}{2} \leq \mathsf{negl}(n).$$

Not IBE-CPA secure. Consider the following IBE-CPA adversary $\mathcal{A}'$:

(a) Choose an arbitrary identity id^* (e.g. $id^* = 0$).

(b) **Key query:** Query an identity $id \neq id^*$ (e.g. $id = 1$) and receive sk_{id}. Since all identities share the same key, $sk_{id} = sk$.

(c) Output challenge identity id^* and two distinct messages (m_0, m_1).

(d) Upon receiving c^*, decrypt: $m' \leftarrow \mathsf{Dec}(sk, c^*)$.

(e) If $m' = m_0$ output $b' = 0$; else output $b' = 1$.

By correctness of Π, $\mathcal{A}'$ always recovers m_b and wins with probability 1. Therefore $\mathcal{E}_{id}$ is *not* IBE-CPA secure. ■

2. Show that the following commitment scheme for bits satisfies both the hiding and binding properties.

Let $G : \{0,1\}^n \to \{0,1\}^{3n}$ be a secure PRG. To commit to a bit $b \in \{0,1\}$, Alice and Bob engage in the following protocol:

- **Commit**: Bob commits to bit $b \in \{0,1\}$ by doing the following:
  (a) Alice chooses a random $r \in \{0,1\}^{3n}$ and sends r to Bob.
  (b) Bob chooses a random $s \in \{0,1\}^n$ and computes:

$$c = \mathrm{com}(s, r, b) := \begin{cases} G(s) & \text{if } b = 0, \\ G(s) \oplus r & \text{if } b = 1. \end{cases}$$

  Bob outputs c as the commitment string and uses s as the opening string.
- **Open**: Bob sends (b, s) to Alice. Alice accepts the opening if $c = \mathrm{com}(s, r, b)$ and rejects otherwise.

Solution We show that the scheme satisfies *computational hiding* and *statistical binding*.

Computational Hiding. We prove that if G is a secure PRG, then no PPT adversary can distinguish a commitment to $b = 0$ from a commitment to $b = 1$ with non-negligible advantage.

Suppose for contradiction that there exists a PPT adversary $\mathcal{A}$ (playing Alice's role) that can distinguish commitments to different bits with non-negligible advantage $\varepsilon(n)$. We construct a PRG distinguisher $\mathcal{B}$ that, given a string $\mathbf{r} \in \{0,1\}^{3n}$ (which is either $G(\mathbf{s})$ for a random $\mathbf{s} \xleftarrow{\$} \{0,1\}^n$, or a truly random string $\mathbf{r} \xleftarrow{\$} \{0,1\}^{3n}$), distinguishes the two cases:

(a) $\mathcal{B}$ receives $\mathbf{r}$ from the PRG challenger.

(b) $\mathcal{B}$ starts $\mathcal{A}$ on input 1^n. $\mathcal{A}$ (acting as Alice) outputs a random string $\mathbf{r}' \xleftarrow{\$} \{0,1\}^{3n}$.

(c) $\mathcal{B}$ samples a random bit $i \xleftarrow{\$} \{0,1\}$ and computes the commitment $\mathbf{c} = \mathbf{r} \oplus i \cdot \mathbf{r}'$ (i.e. $\mathbf{c} = \mathbf{r}$ if $i = 0$, and $\mathbf{c} = \mathbf{r} \oplus \mathbf{r}'$ if $i = 1$). $\mathcal{B}$ sends $\mathbf{c}$ to $\mathcal{A}$.

(d) $\mathcal{A}$ outputs a guess i' for the committed bit.

(e) If $i' = i$, $\mathcal{B}$ outputs $d' = \mathsf{PR}$ (pseudorandom); otherwise $d' = \mathsf{R}$ (random).

When $d = \mathsf{PR}$, $\mathcal{B}$ perfectly simulates the real commitment game so $\mathcal{A}$'s advantage carries over; when $d = \mathsf{R}$, the commitment $\mathbf{c}$ is uniform regardless of i (since $\mathbf{r}$ is uniform), so $\mathcal{A}$'s advantage is zero. Hence $\mathcal{B}$'s PRG distinguishing advantage equals $\mathcal{A}$'s hiding advantage $\varepsilon(n)$, which must be negligible.

Statistical Binding. We show that even a computationally unbounded cheating Bob can open a commitment to two different bits with only negligible probability.

For Bob to break binding, he must find seeds $s, s' \in \{0,1\}^n$ such that $\mathrm{com}(s, r, 0) = \mathrm{com}(s', r, 1)$, i.e.,

$$G(s) = G(s') \oplus r \quad \Longleftrightarrow \quad r = G(s) \oplus G(s').$$

For any fixed pair (s, s'), there is exactly one value of r that satisfies this equation. Since there are at most $2^n \cdot 2^n = 2^{2n}$ pairs (s, s'), the total number of "bad" values of r (for which any collision exists) is at most 2^{2n}.

Since Alice chooses r uniformly at random from $\{0,1\}^{3n}$ (which has 2^{3n} elements), by a union bound:

$$\Pr_{r \xleftarrow{\$} \{0,1\}^{3n}} \left[\exists\, s, s' \text{ s.t. } G(s) \oplus G(s') = r\right] \;\leq\; \frac{2^{2n}}{2^{3n}} = 2^{-n},$$

which is negligible in n. Therefore, with overwhelming probability over Alice's choice of r, no decommitment to two different bits is possible, even for an unbounded adversary. The scheme is statistically binding. ■

3. Recall the ZK proof for graph isomorphism: the interaction is $\mathcal{P}(x, w) \leftrightarrow \mathcal{V}(x)$ where x represents graphs $G_0 = (V, E_0)$ and $G_1 = (V, E_1)$ and w represents a permutation on V such that $w(G_0) = G_1$.

   (a) $\mathcal{P}$ samples a random permutation $\sigma : V \to V$ and sends the graph $H = \sigma(G_1)$ to $\mathcal{V}$.

   (b) $\mathcal{V}$ samples a random bit b and sends it to $\mathcal{P}$.

   (c) If $b = 1$, then $\mathcal{P}$ defines a permutation τ to be σ. If $b = 0$, then instead $\tau = \sigma \circ w$. $\mathcal{P}$ then sends τ to $\mathcal{V}$.

   (d) $\mathcal{V}$ verifies that $\tau(G_b) = H$ and accepts if so.

   Given two instances of this problem (G_0, G_1) and (G_2, G_3) such that $G_0 \cong G_1$ and $G_2 \cong G_3$ but the prover only has access to a witness for one of the two instances, construct a ZK proof for the statement "$G_0 \cong G_1$ OR $G_2 \cong G_3$" and show that it satisfies completeness, soundness and zero-knowledge.
   Note that the proof must not leak which of the two instances the prover has a witness for. You cannot assume that all the graphs have the same vertex set (only that the two graphs in each instance have the same vertex set).

Solution WLOG assume the prover knows a witness w for the first instance, i.e. $w(G_0) = G_1$, but does *not* know a witness for (G_2, G_3). Let V denote the vertex set of (G_0, G_1) and V' the vertex set of (G_2, G_3).

Protocol.

(a) $\mathcal{P}$ samples a random permutation $\sigma_1 : V \to V$ and sets $H_1 = \sigma_1(G_1)$.

$\mathcal{P}$ *also* pre-selects a random bit $b_2 \xleftarrow{\$} \{0,1\}$ for the second instance. $\mathcal{P}$ samples a random permutation $\sigma_2 : V' \to V'$ and sets $H_2 = \sigma_2(G_{2+b_2})$ (so that $\mathcal{P}$ already knows a permutation mapping G_{2+b_2} to H_2, namely σ_2).

$\mathcal{P}$ sends (H_1, H_2) to $\mathcal{V}$.

(b) $\mathcal{V}$ samples a random bit $b \xleftarrow{\$} \{0,1\}$ and sends b to $\mathcal{P}$.

(c) $\mathcal{P}$ sets $b_1 = b \oplus b_2$.

Instance 1 (honest): If $b_1 = 1$, set $\tau_1 = \sigma_1$. If $b_1 = 0$, set $\tau_1 = \sigma_1 \circ w$.

Instance 2 (cheating): Set $\tau_2 = \sigma_2$ (this works because $H_2 = \sigma_2(G_{2+b_2})$ by construction).

$\mathcal{P}$ sends (b_2, τ_1, τ_2) to $\mathcal{V}$.

(d) $\mathcal{V}$ computes $b_1 = b \oplus b_2$ and accepts iff $\tau_1(G_{b_1}) = H_1$ and $\tau_2(G_{2+b_2}) = H_2$.

Completeness. Instance 1: by the same argument as the standard GI protocol, $\tau_1(G_{b_1}) = H_1$ for both values of b_1. Instance 2: $\tau_2(G_{2+b_2}) = \sigma_2(G_{2+b_2}) = H_2$ by construction. So the verifier always accepts.

Soundness. Suppose $G_0 \ncong G_1$ *and* $G_2 \ncong G_3$ (i.e. the statement is false). After sending (H_1, H_2), the prover can commit to H_1 being isomorphic to exactly one of G_0, G_1 (say $G_{b_1^*}$), and H_2 isomorphic to exactly one of G_2, G_3 (say $G_{2+b_2^*}$), since the graphs in each pair are non-isomorphic. The verifier checks $\tau_1(G_{b_1}) = H_1$ and $\tau_2(G_{2+b_2}) = H_2$ where $b_1 = b \oplus b_2$. The prover can only succeed if both $b_1 = b_1^*$ and $b_2 = b_2^*$, which requires $b = b_1^* \oplus b_2^*$. Since b is random and unknown to $\mathcal{P}$ when committing, $\Pr[\text{accept}] = \frac{1}{2}$. (Repeat n times in parallel for negligible soundness error.)

Zero-Knowledge. We construct a simulator $\mathcal{S}$ that, given only (G_0, G_1, G_2, G_3) and black-box access to $\mathcal{V}^*$, produces an indistinguishable transcript.

$\mathcal{S}$ works as follows:

(a) Pre-select random bits $b_1', b_2' \xleftarrow{\$} \{0,1\}$.

(b) Sample random permutations σ_1, σ_2 and set $H_1 = \sigma_1(G_{b_1'})$ and $H_2 = \sigma_2(G_{2+b_2'})$. (The simulator "cheats" on *both* instances by choosing which graph each H_i is isomorphic to.)

(c) Send (H_1, H_2) to $\mathcal{V}^*$ and receive challenge b.

(d) If $b = b_1' \oplus b_2'$: output the transcript with $b_2 = b_2'$, $\tau_1 = \sigma_1$, $\tau_2 = \sigma_2$. Both checks pass by construction.

(e) If $b \neq b_1' \oplus b_2'$: **rewind** $\mathcal{V}^*$ to step 1 and repeat with fresh b_1', b_2'.

Each attempt succeeds with probability $\frac{1}{2}$ (when $b = b'_1 \oplus b'_2$), so the expected number of rewinds is 2.

Indistinguishability: Conditioned on success, $H_1 = \sigma_1(G_{b'_1})$ with $b'_1 = b \oplus b'_2$ and $H_2 = \sigma_2(G_{2+b'_2})$. Since σ_1, σ_2 are uniformly random permutations, H_1 and H_2 are each uniformly random graphs (on their respective vertex sets) that are isomorphic to the correct graphs—exactly as in the real protocol. The permutations $\tau_1 = \sigma_1$ and $\tau_2 = \sigma_2$ are the unique isomorphisms mapping G_{b_1} to H_1 and G_{2+b_2} to H_2, matching the real distribution. Furthermore, b_2 is uniform in $\{0, 1\}$, as in the real protocol. Hence the simulated transcript is identically distributed to the real one.

Note that the protocol and simulation are symmetric in which instance the prover knows the witness for: the prover always cheats on one instance (by choosing its bit) and honestly answers the other. Since b_2 is uniformly random, the transcript reveals nothing about which instance the prover holds a witness for. ■