Fundamentals of Cryptography: Midterm

Wednesday Nov 8, 3-6PM

Problem 1 (1pt) ...there exists an positive integer c such that $f(n) = O(n^c)$.

Problem 2 (1pt) ...for any positive integer c, $f(n) = O(n^{-c})$.

Problem 3 (2pt) ...either of the following.

- choose a message and ask the challenger to return a fresh encryption.
- choose a ciphertext $\neq c$ and ask the challenger to return its decryption.

Problem 4 (2pt) (a)(c)

CRHF is seemingly a stronger assumption than OWF. $P \neq NP$ is seemingly a weaker assumption than OWF.

Problem 5 (2pt) (d)(b)(c)(a)

Problem 6 (3pt) In the j-th hybrid H_j, the distinguisher receives a sample from the following distribution

```
Sample $y_1, \ldots, y_j \leftarrow \{0,1\}$, $x^j \leftarrow \{0,1\}^\lambda$;
For $i = j+1, \ldots, \lambda$, compute $y_i \| x^i = g(x^{i-1})$;
Output $y_1 \| y_2 \| \ldots \| y_\lambda \| x^\lambda$.
```

Problem 7 (5pt) F' is not necessarily a PRF.

Suppose $H : \{0,1\}^{\lambda-1} \times \{0,1\}^\lambda \to \{0,1\}^\lambda$ be a secure PRF. Define F as

$$F(k,x) = \begin{cases} k_1 \| H(k_{2:\lambda}, 0)_{2:\lambda} & \text{if } x = 0 \\ H(k_{2:\lambda}, x) & \text{otherwise} \end{cases}$$

That is, the output is mostly the same as $H(k_{2:\lambda}, x)$, the first bit in the output is replaced by k_1 when input $x = 0$.

Then F' is not a PRF, because the first bit of $F'(k,x)$ always equals to 1.

Problem 8 (5pt) Let $F' : \{0,1\}^\lambda \times \lambda^n \to \{0,1\}$ be a secure PRF. Construct F as

$$F(k,x) = \begin{cases} F'(k,0), & \text{if } x = 0 \\ F'(k,i) \oplus F'(k,i-1), & \text{otherwise} \end{cases}$$

Then $\text{psum}(k,x) = F(k',x)$.

Problem 9 (5pt) We start with a simple property of DUF. For any polynomial-size sets $M, \Delta \subseteq \{0,1\}^\lambda$

$$\Pr_k\Big[\exists\text{distinct } m_0, m_1 \in M,\ \exists\delta \in \Delta,\ H(k,m_0) \oplus H(k,m_1) = \delta\Big] \le \text{negl}(\lambda).$$

The property follows simply from the union bound.

Let $x_0^i, x_1^i, x_2^i, x_3^i, x_4^i, y_1^i, y_2^i, y_3^i$ be the input, output and intermediate values associated with the distinguisher's i-th query. Without loss of generality, we can assume inputs (x_0^i, x_1^i) are distinct.

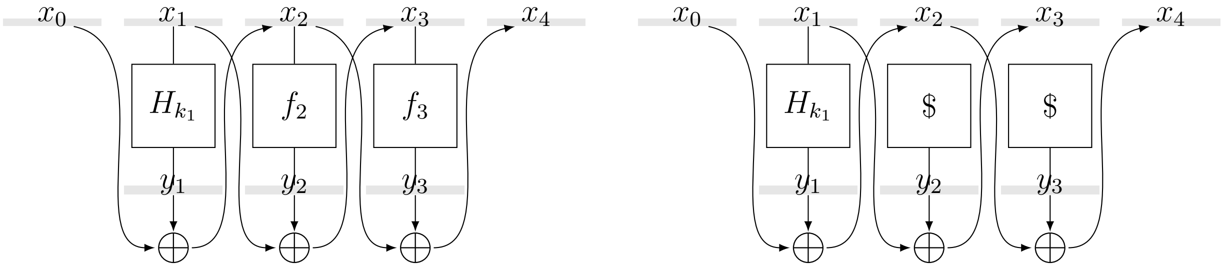
Without loss of generality, we can assume the adversary is deterministic. (A randomized adversary is just an distribution over deterministic adversaries.)

Here is an outline of the hybrid proof.

- Hybrid 0 is the real world.
- Hybrid 1: $F(k_2,\cdot), F(k_3,\cdot)$ are replaced by a truly random function f_2, f_3.

  Hybrid 1 is indistinguishable from Hybrid 0 due to the security of PRF F.
- Hybrid 2: f_2, f_3 are replaced by a randomness generator, which ignores the input and outputs a fresh random string.
- Hybrid 3: The ideal world. (x_3^i, x_4^i) is sampled at random.

  Hybrid 2 is indistinguishable from Hybrid 3 because their distributions are identical.



Hybrid 1

Hybrid 2

It remains to show the indistinguishability between Hybrid 1 and Hybrid 2. To formalize the proof, we specify how the randomness are used in Hybrid 1 and Hybrid 2. There is a tape of random strings $r_1, r_2, \ldots$.

- In Hybrid 2, the randomness generator always pick the next unused random string.
- In Hybrid 1, the random functions f_2, f_3 are sampled on the fly. Whenever $f_i(x)$ is queried and not defined, set $f_i(x)$ as the next unused random string.

If there is no duplicated queries to f_2, f_3, then Hybrid 1 is identical to Hybrid 2.

In Hybrid 2, there is not collision in x_3 with overwhelming probability because $\{x_3^i\}$ are i.i.d. uniform. Also in Hybrid 2, there is not collision in x_2 with overwhelming probability, because $\{x_2^i\}$ are perfectly hidden from the adversary. If the adversary can force a collision in x_2 blindly, it can also attack the security of DUF.

Note that, the probability that is no collision on either $\{x_2^i\}$ or $\{x_3^i\}$ in Hybrid 1 is exactly the same as in Hybrid 2. Let $(r_1, r_2, \ldots)$ be a tape such that the corresponding execution in Hybrid 1 has a collision. Before the first duplicated query, the executions in Hybrid 2 and Hybrid 1 are identical. Thus the same collision also occurs in Hybrid 2 using tape $(r_1, r_2, \ldots)$. Such technique is called *randomness mapping*.

Problem 10 (5pt) Define an encryption scheme Π as

- $\mathsf{Gen}(1^\lambda)$: Compute $k_A \leftarrow \mathsf{Gen}_A(1^\lambda)$, $k_B \leftarrow \mathsf{Gen}_B(1^\lambda)$. Output (k_A, k_B).
- $\mathsf{Enc}((k_A, k_B), m)$: Sample random r of the same length as m. Compute $c_A \leftarrow \mathsf{Enc}_A(k_A, r)$, $c_B \leftarrow \mathsf{Enc}_B(k_B, m \oplus r))$. Output $c = (c_A, c_B)$.
- $\mathsf{Dec}((k_A, k_B), (c_A, c_B)) = \mathsf{Dec}_A(k_A, c_A) \oplus \mathsf{Dec}_B(k_B, c_B)$.

Correctness is straight-forward.

Since the scheme is symmetric, assume without loss of generality that Π_B is CPA-secure and Π_A is not. We show a reduction from an adversary breaking Π to an adversary breaking Π_B.

Let $\mathcal{A}$ be an adversary that wins $\mathrm{PrivK}^{\mathrm{cpa}}_{\mathcal{A},\Pi}$ with non-negligible probability. Consider the following adversary $\mathcal{B}$.

- $\mathcal{B}$ samples key $k_A \leftarrow \mathsf{Gen}_A(1^\lambda)$.
- $\mathcal{A}(1^\lambda)$ is emulated.
- Whenever $\mathcal{A}$ asks for the encryption of m: $\mathcal{B}$ samples random r, compute $c_A \leftarrow \mathsf{Enc}_A(k_A, r)$; asks the challenger to compute c_B as the encryption of $m \oplus r$; returns (c_A, c_B) to $\mathcal{A}$.
- When $\mathcal{A}$ picks two message m_0, m_1: $\mathcal{B}$ samples random r, compute $c_A \leftarrow \mathsf{Enc}_A(k_A, r)$; asks the challenger to compute c_B as the encryption of $m_b \oplus r$; returns (c_A, c_B) to $\mathcal{A}$.
- When $\mathcal{A}$ outputs a guess b', $\mathcal{B}$ outputs the same guess.

The probability $\mathcal{B}$ wins $\mathrm{PrivK}^{\mathrm{cpa}}_{\mathcal{B},\Pi_B}$ is identical to that of $\mathcal{A}$ winning $\mathrm{PrivK}^{\mathrm{cpa}}_{\mathcal{A},\Pi}$.

Problem 11 (5pt) Let $H : \{0,1\}^n \to \{0,1\}$ be a PRF, let $G : \{0,1\}^\lambda \to \{0,1\}^{2n\lambda}$ be a PRG, define F as

$$F((k_{1,0}, k_{1,1}, k_{2,0}, k_{2,1}, \ldots, k_{n,0}, k_{n,1}), x) = \bigoplus_{i=1}^{n} H(k_{i,x_i}, x)$$

where its key $G(k) = (k_{1,0}, k_{1,1}, k_{2,0}, k_{2,1}, \ldots, k_{n,0}, k_{n,1})$ is parsed as $2n$ keys for H.

For a bit-fixing constrain $c \in \{0, 1, ?\}^n$, the constrained key k_c consists of

$$\begin{cases} k_{i,0}, & \text{if } c_i = 0 \\ k_{i,1}, & \text{if } c_i = 1 \\ (k_{i,0}, k_{i,1}), & \text{if } c_i = ? \end{cases}$$

for each $1 \le i \le n$.

The correctness is rather straight-forward.

For privacy, for every (i, b) that $c_i = 1 - b$, $H(k_{i,b}, \cdot)$ can be replaced by a random function from $\{0,1\}^n$ to $\{0,1\}$ since $k_{i,b}$ is hidden from the distinguisher.