1. (Expanding PRF output) Suppose that we have a good PRF $E : \{0, 1\}^k \times \{0, 1\}^n \rightarrow \{0, 1\}^n$. We want to build a good PRF $F : \{0, 1\}^k \times \{0, 1\}^n \rightarrow \{0, 1\}^{2n}$. (Note that here the output length of $F$ is twice that of $E$.)
   a) Bob suggests the following construction: $F_K(X) = E_K(X)||E_K(\overline{X})$, where $\overline{X}$ is the complement of $X$. (For example $01 \overline{1} = 100$, and $100 \overline{1} = 0110$.) Give an efficient PRF attack on Bob’s construction. Your attack should use just two queries.
   b) Alice instead suggests the following construction. To compute $F_K(X)$, we first internally compute $Y \leftarrow E_K(X)$, and then output $E_K(Y)||E_K(\overline{Y})$, where $\overline{Y}$ is the complement of $Y$ as defined in part (a). Using the game-based proof framework, prove that Alice construction is a good PRF.

2. (An infamous incorrect use of CBC) The encryption schemes we learn so far (such as CTR mode, or CBC mode) are stateless. However, it’s also common to use stateful encryption, meaning that you maintain a short string $S$, and each time you encrypt, the encryption scheme takes both the message and the state $S$ as input. It then produces a ciphertext $C$, and also updates the state $S$. (The decryption scheme however doesn’t use $S$.)
   For example, here’s how to turn CTR mode into a stateful encryption. The first time you ever encrypt, the state $S$ is the null string $\bot$. The encryption scheme would use the the ordinary CTR mode with IV as $0^n$, and updates $S \leftarrow 0$. Subsequently, whenever you encrypt, the encryption scheme updates $S \leftarrow S + 1 \operatorname{mod} 2^n$, then use ordinary CTR with IV as the $n$-bit encoding of $S$. This stateful CTR mode does achieve left-or-right security.
   Now here’s an incorrect way to use CBC in a stateful way that SSH employed in 2000. The first time you ever encrypt, the state $S$ is the null string $\bot$. The encryption scheme would use the ordinary CBC mode with IV as a random string and update $S$ as the last block of the resulting ciphertext. Subsequently, whenever you encrypt, the encryption scheme uses the ordinary CBC mode with IV as $S$, and update $S$ as the last block of the resulting ciphertext. Give an attack that breaks the left-or-right security of this stateful CBC. Your attack should use just 2 queries.

3. (A practice example of hybrid arguments) Consider the following security definition PRF* on a (keyed) function $F : K \times \{0, 1\}^n \rightarrow \{0, 1\}^n$. Here an adversary $A$ is given two oracles $\text{Fn}$ and $\text{Enc}$, and is then dropped into either a real world or a random world. In each world, the game initially picks a secret key $K \leftarrow \mathcal{K}$ and a random function $f \leftarrow \text{Func}(n, n)$; the oracle $\text{Enc}$ will always return $F_K(X)$ on any query $X \in \{0, 1\}^n$. For each query $\text{Fn}(X)$ with $X \in \{0, 1\}^n$, the game will return $F_K(X)$ in the real world $\text{Real}_K^f$, and return $f(X)$ in the random world $\text{Rand}_K^f$. The adversary has to tell which world it’s in by outputting a bit $b'$. (The convention is that $b' = 1$ means the real world, and $b' = 0$ means the random world.) To prevent trivial wins, the adversary is prohibited from making a query $X$ to one oracle and then repeating the same query $X$ to the other oracle. Define

   $$\text{Adv}_{F}^{PRF^*}(A) = \Pr[\text{Real}_K^f(A) \Rightarrow 1] - \Pr[\text{Rand}_K^f(A) \Rightarrow 1] .$$

   This notion PRF* is an extension of the PRF notion that we learned in class, as the adversary is equipped with an additional oracle $\text{Enc}$. Show that PRF still implies PRF*, meaning that the two notions are in fact equivalent.

---

Homework 3: Deadline Thursday 10/22

Instructor: Viet Tung Hoang