Problem sheet 3 for Course 02231, 2025

These practice problems have the purpose of helping you understand the material better and learning the skills that are necessary to analyze cryptographic constructions, and sometimes to prepare you for the next class. All answers should be supported by a written justification. To gauge whether a justification is sufficient, ask yourself if your peers would be convinced by it without additional explanations.

We denote vectors as $\mathbf{x} \in \{0,1\}^\lambda$. By $\mathbf{x}[i]$ we denote the ith index of $\mathbf{x}$, where $i \in \{1,\ldots,\lambda\}$. As in the lecture, we write $k \leftarrow K$ if k is sampled from the set K such that it can be each element from K with equal probability $1/|K|$.

Exercise 1. (**Triple the DES!**)

In the lecture you learned about the DES Pseudorandom Permutation (PRP) or block cipher, which has block size $B = 64$ and key length $\lambda = 56$. We denote its algorithms as $\mathsf{DES.KEYGEN}, \mathsf{DES.ENC}, \mathsf{DES.DEC}$. A well-known variation of the DES algorithm is called Triple-DES and works as follows:

$\mathsf{3DES.KEYGEN}()$**:**

1. Compute $k_1, k_2, k_3 \leftarrow \mathsf{DES.KEYGEN}()$.
2. Output $k = (k_1, k_2, k_3)$.

$\mathsf{3DES.ENC}(k, m)$**:**

1. Check that $k = (k_1, k_2, k_3)$.
2. Output $\mathsf{DES.ENC}(k_3, \mathsf{DES.DEC}(k_2, \mathsf{DES.ENC}(k_1, m)))$.

1. Find the missing $\mathsf{3DES.DEC}$ algorithm and argue why the overall scheme is correct.
2. In a so-called brute-force attack, an attacker tries out all possible keys from the keyspace until it finds the one which decrypts a ciphertext. Assume that we are given a DES plaintext/ciphertext pair $m, c = \mathsf{DES.ENC}(k, m)$ for an unknown key k. Our goal is to find k. Further, assume for simplicity that one $\mathsf{DES.ENC}$ and thus $\mathsf{DES.DEC}$-operation takes 1ns of time on a computer of your choice.
   (a) How many days does it take to recover the one (or more) keys which would have led to the given plaintext/ciphertext pair using the computer?
   (b) Assume you have 1024 such machines at your disposal. Can you use these machines to speed up the brute-force attack? How long will the attack take now?
   (c) Compute the same attack runtime for 3DES. You may want to compare the runtime of the attack to the age of the universe, which is estimated at 13.8 billion years.

❓ Exercise 2. (There are more PRFs)

In the class, you learned what a Pseudorandom Function, or PRF for short, is: It is a function $F : \{0,1\}^{\lambda} \times \{0,1\}^{in} \to \{0,1\}^{out}$ such that $L^F_{PRF-Real}$ and $L^F_{PRF-Rand}$ are indistinguishable.

Assume that you have such a PRF F given. Let $\mathbf{r} \in \{0,1\}^{out}$ be any fixed, publicly known string, and define the function $G : \{0,1\}^{\lambda} \times \{0,1\}^{in} \to \{0,1\}^{out}$ as $G(k,x) = F(k,x) \oplus \mathbf{r}$.

Show that G is also a PRF. To prove this, it is instructive to follow the steps below:

1. Write down the *Lookup* functions in the libraries $L^G_{PRF-Real}$ and $L^G_{PRF-Rand}$.
2. Find a way to rewrite $L^G_{PRF-Real}$ into another library $\overline{L}^G_{PRF-Real}$ which uses $L^F_{PRF-Real}$.
3. You can now apply the hybrid technique to switch out $L^F_{PRF-Real}$ with $L^F_{PRF-Rand}$.
4. Based on this, find an argument how to complete indistinguishability of the resulting library with $L^G_{PRF-Rand}$.

❓ Exercise 3. (Birthday bounds in practice)

In the class, you learned about $BirthdayProb(q,N)$ as the probability of sampling two or more identical items after q random queries from a set of size N. In class, we computed an upper-bound on this probability.

1. Write a program (in your preferred programming language) that computes $BirthdayProb(q,N)$ given the exact expression that we used in the class. The program should also compute the upper-bound on $BirthdayProb(q,N)$ that we derived in class. How close is the upper-bound to the real value?
2. Compute the probability that two of your fellow students in this course (yourself included) are born on the same date, assuming uniform distribution of birthdays throughout the year, using your program.
3. In the class, we used $BirthdayProb(q,N)$ to argue that we sometimes can treat PRFs and PRPs as the same! But this assumed that $q \ll N$. Assume you have a PRF $F : \{0,1\}^{32} \times \{0,1\}^{32} \to \{0,1\}^{32}$ as well as a PRP $G : \{0,1\}^{32} \times \{0,1\}^{32} \to \{0,1\}^{32}$. What is the probability of distinguishing F from G for $q = 2, 2^8, 2^{12}$ or 2^{16}?