DD2448 Foundations of Cryptography (krypto20) Homework

Douglas Wikström, dog@kth.se

May 8, 2020

Abstract

Make sure that you read and understand `Files`→`HW/solution_rules.pdf` at Canvas before you start. This document details the rules for solving and handing in your solutions.

Problem 1 (Pseudo randomness and ciphers). Let $k(n) > 0$ be a fixed polynomial, let $f_n : \{0,1\}^{k(n)} \times \{0,1\}^n \to \{0,1\}$ for $n \in \mathbb{N}$, let S_n be the set of all functions $\{0,1\}^n \to \{0,1\}$, and assume that for every polynomial time algorithm $\mathcal{A}$

$$\left| \Pr_{K \in \{0,1\}^{k(n)}} \left[\mathcal{A}^{f_n(K,\cdot)}(1^n) = 1 \right] - \Pr_{R \in S_n} \left[\mathcal{A}^{R(\cdot)}(1^n) = 1 \right] \right| < \epsilon(n) \enspace ,$$

where $\epsilon(n)$ is a negligible function. We learned in class that four rounds suffice in a Feistel network to construct a pseudo random permutation from a pseudo random function, but the range of our function is only one bit so we cannot use this construction!

Task 1.1 (3T). Consult the literature and see if you can find a modified Feistel network that is suitable for our function. Describe the construction and pros and cons relative the plain Feistel network. Cite your sources.

Task 1.2 (3T). Another approach is to instead construct a pseudo random family of functions $F' = \{f'_n(K,\cdot)\}_{n \in \mathbb{N}, K \in \{0,1\}^{k(n)}}$ such that $f'_n : \{0,1\}^{k(n)} \times \{0,1\}^n \to \{0,1\}^n$ from the one we have. We can then use your construction F' as described in class to construct a Feistel network directly. Define your proposal for such a function family F'.[1]

Task 1.3 (4T). Prove that if there exists a polynomial time algorithm $\mathcal{A}'$ that violates the pseudo randomness of F', then there exists a polynomial time algorithm $\mathcal{A}$ that violates the pseudo randomness of F.

Thus, if F is pseudo random, then your function family F' is pseudo random as well and using it in a four-round Feistel network gives a family of pseudo random permutations.

Problem 2 (Shannon Entropy and Huffman coding). Let X be a random variable over the set $\{a,b,c,d,e,f,g,h,i,j,k\}$ with probability function $\mathsf{P}_X(\cdot)$ defined by $\mathsf{P}_X(a) = 0.03$, $\mathsf{P}_X(b) = 0.07$, $\mathsf{P}_X(c) = 0.17$, $\mathsf{P}_X(d) = 0.21$, $\mathsf{P}_X(e) = 0.09$, $\mathsf{P}_X(f) = 0.06$, $\mathsf{P}_X(g) = 0.07$, $\mathsf{P}_X(h) = 0.04$, $\mathsf{P}_X(i) = 0.04$, $\mathsf{P}_X(j) = 0.05$, and $\mathsf{P}_X(k) = 0.17$.

Task 2.1 (1T). Compute the Shannon entropy $H(X)$ of X.

Task 2.2 (2T). Describe the Huffman code for a memoryless source with distribution defined by $\mathsf{P}_X(\cdot)$ and binary channel (without errors).

Task 2.3 (1T). Compute the expected codeword length of the Huffman code under P_X.

Task 2.4 (1T). If the expected codeword length is not equal to $H(X)$, then try to give an intuitive explanation of why this is the case.

[1] We do not worry about increased key length at this point.

Problem 3 (Replaces invited talk, 4T). Watch Chris Peikert's talk about lattice-based cryptography at YouTube: `https://www.youtube.com/watch?v=FVFw_qb1ZkY`. Lattices are promising in that no algorithms are known which break the associated computational problems on a quantum computer efficiently, but this is not the only reason lattices are important in cryptography.

Write a brief summary of lattice-based cryptography of at most one page. Focus on the pros and cons of this type of computational assumption compared to standard assumptions such as the RSA assumption or various discrete logarithm asssumptions we have seen in class.

Problem 4 (Discrete logarithm problem). Prove that for any[2] group G_q with prime order q the *worst case* discrete logarithm problem is equivalent to the average case discrete logarithm problem defined in class. More precisely, do the following:

Task 4.1 (2T). Define the worst case[3] discrete logarithm problem in G_q.

Task 4.2 (1T). Define the (standard) average case discrete logarithm problem in G_q to allow easy reference.

Task 4.3 (2T). Suppose that $\mathcal{A}$ violates the the average case discrete logarithm assumption. Describe an algorithm $\mathcal{W}$ that solves the worst case discrete logarithm problem using $\mathcal{A}$ as a subroutine.

Task 4.4 (1T). Prove that $\mathcal{W}$ violates the worst case discrete logarithm assumption.

Problem 5. (Sampling) Define the statistical distance between distributions over a finite set Ω with probability functions P_X and P_Y by $\|\mathsf{P}_X - \mathsf{P}_Y\| = \frac{1}{2}\sum_{x\in\Omega} |\mathsf{P}_X(x) - \mathsf{P}_Y(x)|$.

Task 5.1 (5T). Let $q > 2$ be a prime, let X be uniformly distributed over $\{0,1\}^{\lceil \log q\rceil + t}$, define Y by $Y = X \bmod q$, and let Z be uniformly distributed over $\mathbb{Z}_q$. Prove the best bound you can of the form $\|\mathsf{P}_Y - \mathsf{P}_Z\| \leq \beta(t)$.[4]

Hint: Consider a stack of fresh A4 papers, except part of the top sheet has been cut away with a pair of scissors. X amounts to picking a random point on the joint surface of *all* sheets of paper. Y amounts to the vertical projection of that point to the bottom sheet. How far from the uniform distribution is the resulting distribution?

Task 5.2 (4T). Let $X_1, \ldots, X_k \in \mathbb{Z}_q^k$ be uniformly and independently distributed. Prove the best bound you can of the form $\Pr\left[\mathsf{span}(X_1,\ldots,X_k) \neq \mathbb{Z}_q^k\right] \leq \ell(q,k)$. (In other words, what is the probability that the vectors are not linearly independent.)

Task 5.3 (4T). We can clearly combine our two results and use a PRG to sample random vectors which are linearly independent with overwhelming probability when $q \gg k$ using little randomness, but we do not need a fully-blown PRG for this![5]

Consider the function $F(x) = (1, x, x^2, \ldots, x^{k-1})$ and suppose that we define $Y_i = F(S_i)$, where the $S_i \in \mathbb{Z}_q$ are uniformly and independently distributed. Prove the best bound you can of the form $\Pr\left[\mathsf{span}(Y_1,\ldots,Y_k) \neq \mathbb{Z}_q^k\right] \leq f(q,k)$.

[2] Strictly speaking we need to assume that there are efficient algorithms for performing group operations, but this is the case for all the examples we encounter in applications.

[3] This is meant to be a challenge for you.

[4] This situation is quite common in applied cryptography where we need to sample almost uniformly from field, but only have random bit strings to start with.

[5] What we really need is a distribution over some set of vectors which are linearly independent with high probability when the vectors are chosen independently.

Problem 6. (Caveat) In cryptography any claim of security is relative a specific definition and under explicit assumptions. Cryptography is fragile in the sense that small changes can have drastic consequences, but robust in the sense that provably secure constructions do have the claimed properties.

In the following you will construct contrived examples to illustrate this. In each case you start with a construction which satisfies a definition of security covered in class and construct another primitive which satisfies the *same* definition, but still has some additional features that you may have expected to be impossible under the given definition of security.

You are not allowed to make any additional assumptions, i.e., you must in each case provide a construction that only relies on the assumption that the construction you start from satisfies the stated definition of security.

Task 6.1 (2T). *One-wayness does not guarantee secrecy.*
Suppose that $f : \{0,1\}^* \to \{0,1\}^*$ is a one-way function. Construct another one-way function f' such that it is trivial to compute the most significant bit of the input from the output.

Task 6.2 (2T). *CPA security does not prevent modifying encrypted plaintexts.*
Suppose that $\mathsf{CS} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is a CPA secure cryptosystem. Construct another CPA secure cryptosystem $\mathsf{CS}' = (\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$ such that any bit of an encrypted plaintext can be flipped without the secret key.

Task 6.3 (2T). *CMA security does not provide any privacy for signed plaintexts.*
Suppose that $\mathsf{SS} = (\mathsf{Gen}, \mathsf{Sig}, \mathsf{Vf})$ is a CMA secure signature scheme. Construct a CMA secure signature scheme $\mathsf{SS}' = (\mathsf{Gen}', \mathsf{Sig}', \mathsf{Vf}')$ such that the plaintext can be recovered from any valid signature of the plaintext.

Task 6.4 (2T). *Collision resistant hash functions do not necessarily give random-looking outputs.*
Suppose that $H_k : \{0,1\}^* \to \{0,1\}^n$ for any $k \in \{0,1\}^n$ and that $\{H_k\}_{n\in\mathbb{N},k\in\{0,1\}^n}$ is a collision resistant family of hash functions. Construct a collision resistant family of hashfunctions $\{H'_k\}_{n\in\mathbb{N},k\in\{0,1\}^n}$ such that every output $(b_{n-1},\ldots,b_0)$ satisfies $\sum_{i=0}^{n-1} b_i = 0 \bmod 3$.