CS 388H: Cryptography **Fall 2026**

Homework 1: Symmetric Cryptography

Due: September 11, 2026 at 11:59pm (Submit on Gradescope) **Instructor:** David Wu

Instructions. You **must** typeset your solution in LaTeX using the provided template:

https://www.cs.utexas.edu/~dwu4/courses/fa26/static/homework.tex

You must submit your problem set via Gradescope (accessible through Canvas).

Collaboration Policy. You may discuss your general *high-level* strategy with other students, but you may not share any written documents or code. You should not search online for solutions to these problems. If you do consult external sources, you must cite them in your submission. You must include the names of all of your collaborators with your submission. Refer to the official course policies for the full details.

Problem 1: Pseudorandom Generators [12 points]. Let $G\colon \{0,1\}^\lambda \to \{0,1\}^n$ be a secure PRG. For each of the following functions G', indicate whether it is a secure PRG or not. If it is secure, give a *formal* proof. If not, describe an explicit attack.

(a) $G'(s) := G(s) \| (G(s) \oplus 1^n)$, where 1^n denotes the all-ones string of length n.

(b) $G'(s_1 \| s_2) := s_1 \| G(s_2)$, where $s_1, s_2 \in \{0,1\}^\lambda$.

Recall that for two bit-strings $s_1, s_2 \in \{0,1\}^*$, we write $s_1 \| s_2$ to denote the *concatenation* of s_1 and s_2. Please refer to this handout for examples of how to formally show whether a construction is secure or not.

Problem 2: One-Way Functions [12 points]. Let $f\colon \{0,1\}^\lambda \to \{0,1\}^n$ be an efficiently-computable function. We say f is a one-way function if for all efficient adversaries $\mathcal{A}$, there exists a negligible function $\mathrm{negl}(\cdot)$ such that

$$\Pr\left[f(y) = f(x) : \begin{array}{c} x \xleftarrow{R} \{0,1\}^\lambda \\ y \leftarrow \mathcal{A}(f(x)) \end{array} \right] = \mathrm{negl}(\lambda).$$

Suppose $G\colon \{0,1\}^\lambda \to \{0,1\}^n$ is a secure PRG. For simplicity, suppose $n \geq 2\lambda$. Show that G is one-way. Namely, show the contrapositive: given an efficient adversary $\mathcal{A}$ that breaks one-wayness of G with non-negligible probability, construct an efficient adversary $\mathcal{B}$ that breaks pseudorandomness of G with non-negligible advantage.

Problem 3: Circular Security and Insecurity [20 points]. In this question, we explore what happens when we use a symmetric encryption scheme to encrypt its own key. Let $(\mathsf{Encrypt}, \mathsf{Decrypt})$ be a semantically-secure cipher with key space $\mathcal{K} = \{0,1\}^\lambda$, message space $\mathcal{M} = \{0,1\}^\lambda$, and ciphertext space $\mathcal{C} = \{0,1\}^n$ (where $n > \lambda$).

(a) Use $(\mathsf{Encrypt}, \mathsf{Decrypt})$ to construct a new encryption scheme $(\mathsf{Encrypt}', \mathsf{Decrypt}')$ with the same key space and message space (but possibly different ciphertext space) that is semantically secure, but is no longer semantically secure if the adversary is given $\mathsf{Encrypt}'(k, k)$. Prove correctness and semantic

security of your new construction. Then, show how an adversary who knows $\mathsf{Encrypt}'(k,k)$ can break semantic security of your new scheme. Compute the advantage of your attack.

Remark: This problem illustrates that encrypting messages that depend on the secret key of an encryption scheme can be problematic.

(b) Use $(\mathsf{Encrypt}, \mathsf{Decrypt})$ to construct an encryption scheme $(\mathsf{Encrypt}', \mathsf{Decrypt}')$ with the same key space and message space (but possibly different ciphertext space) that is semantically secure even given $\mathsf{Encrypt}'(k,k)$; namely, at the beginning of the semantic security game, the adversary is first given $\mathsf{Encrypt}'(k,k)$. Prove correctness and semantic security (given $\mathsf{Encrypt}'(k,k)$) of your construction. We say that such an encryption scheme satisfies "circular security."

Problem 4: Chaining Encryption [20 points]. Let $(\mathsf{Encrypt}, \mathsf{Decrypt})$ be a cipher with key space $\mathcal{K} = \{0,1\}^\lambda$, message space $\mathcal{M} = \{0,1\}^\lambda$, and ciphertext space $\mathcal{C} = \{0,1\}^n$. Consider the following cipher $(\mathsf{Encrypt}', \mathsf{Decrypt}')$ over the same key space $\mathcal{K}' = \mathcal{K}$, message space $\mathcal{M}' = \mathcal{M}$, but a different ciphertext space $\mathcal{C}' = \{0,1\}^{2n}$:

- $\mathsf{Encrypt}'(k,m)$: Sample $k' \xleftarrow{R} \{0,1\}^\lambda$ and compute $c_1 \leftarrow \mathsf{Encrypt}(k,k')$ and $c_2 \leftarrow \mathsf{Encrypt}(k',m)$. Output the ciphertext $c = (c_1, c_2)$.
- $\mathsf{Decrypt}'(k,(c_1,c_2))$: Compute $k' \leftarrow \mathsf{Decrypt}(k,c_1)$ and output $\mathsf{Decrypt}(k',c_2)$.

Show that if $(\mathsf{Encrypt}, \mathsf{Decrypt})$ is semantically secure, then $(\mathsf{Encrypt}', \mathsf{Decrypt}')$ is also semantically secure. **Hint:** Consider using a hybrid argument.

Problem 5: Key Leakage in PRFs [15 points]. Let $F\colon \{0,1\}^\lambda \times \{0,1\}^\lambda \to \{0,1\}$ be a secure PRF. Use F to construct a function $F'\colon \{0,1\}^{\lambda+1} \times \{0,1\}^\lambda \to \{0,1\}$ with the following two properties:

- F' is a secure PRF.
- If the adversary learns the last bit of the key, then F' is no longer secure.

(a) Describe your construction $F'\colon \{0,1\}^{\lambda+1} \times \{0,1\}^\lambda \to \{0,1\}$. **Hint:** Try changing the value of F at a single point.

(b) Show that if F is a secure PRF, then your construction F' is a secure PRF.

(c) Show that your construction F' is insecure against an adversary that learns the last bit of the key (i.e., at the beginning of the PRF security game, the challenger gives the last bit of the PRF key to the adversary). Specifically, give a complete description of your adversary and compute its advantage.

Remark: This problem shows that leaking even a *single* bit of the secret key can break PRF security.

Problem 6: Time Spent [1 point]. How long did you spend on this problem set? This is for calibration purposes, and the response you provide does not affect your score.

Optional Feedback. Please answer the following *optional* questions to help us design future problem sets. You do not need to answer these questions. However, we do encourage you to provide us feedback on how to improve the course experience.

(a) What was your favorite problem on this problem set? Why?

(b) What was your least favorite problem on this problem set? Why?

(c) Do you have any other feedback for this problem set?

(d) Do you have any other feedback on the course so far?