CS 388H: Cryptography **Fall 2022**

Take-Home Final Exam

Due: December 9, 2022 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/fa22/static/homework.tex`

You must submit your completed exam via Gradescope (accessible through Canvas).

Collaboration Policy. This is an *individual* assignment. You are not allowed to collaborate with anyone on these problems and you are not permitted to search online for solutions to these problems. If you do consult external sources (these *cannot* include solutions), you must cite them in your submission.

1 Part I: Conceptual Questions

Problem 1: Conceptual Questions [30 points]. You do **not** need to provide any justification for any part of this question, and any justification you write will be ignored. For the multiple choice questions, there could be **multiple** answers or **zero** correct answers. For full credit, you should select **all** correct responses, or indicate that there are **none**.

1. **Symmetric cryptography.** Let S be a set of constant size (e.g., $|S| = 64$). Which (if any) of the following schemes exist?

   (a) A semantically secure symmetric encryption scheme with key space S, message space S, and ciphertext space S.

   (b) A CPA-secure symmetric encryption scheme with key space S, message space S, and a ciphertext space of size $2^{\lambda|S|}$, where λ is a security parameter.

   (c) A secure MAC with key space S, message space S, and tag space S.

   (d) A secure one-time MAC with key space S, message space S, and tag space S.

2. **Padding in CBC.** Recall that when encrypting messages in CBC mode, the messages must first be padded to be a multiple of the block length. Suppose we use a block cipher on an n-bit domain in randomized CBC mode to build a symmetric encryption scheme. Which (if any) of the following padding functions can be applied to obtain a correct *and* CPA-secure cipher? The padding function $\text{pad}\colon \{0,1\}^{\leq n} \to \{0,1\}^{kn}$ takes in the last block of the message and outputs a string that is exactly k blocks long for some positive integer $k \in \mathbb{N}$.

   (a) $\text{pad}(x) = \begin{cases} x & \text{if } |x| = n \\ x \| 1 \| 0^{n-|x|-1} & \text{otherwise.} \end{cases}$

   (b) $\text{pad}(x) = \begin{cases} x \| 1^n & \text{if } |x| = n \\ x \| 1^{n-|x|} & \text{otherwise.} \end{cases}$

(c) $\mathrm{pad}(x) = \begin{cases} x\|1\|0^{n-1} & \text{if } |x| = n \\ x\|1\|0^{n-|x|-1} & \text{otherwise.} \end{cases}$

(d) $\mathrm{pad}(x) = \begin{cases} x\|1\|r_1 & \text{if } |x| = n \\ x\|1\|r_2 & \text{otherwise,} \end{cases}$ where $r_1 \xleftarrow{R} \{0,1\}^{n-1}$ and $r_2 \xleftarrow{R} \{0,1\}^{n-|x|-1}$.

3. **Authenticated encryption.** Suppose $(\mathsf{Encrypt}, \mathsf{Decrypt})$ is an authenticated encryption scheme with key-space $\{0,1\}^\lambda$, message space $\{0,1\}^\lambda$, and ciphertext space $\{0,1\}^n$. In the following, let $k \xleftarrow{R} \{0,1\}^\lambda$ and let $\mathsf{ct} \leftarrow \mathsf{Encrypt}(k, 0^\lambda)$. Which (if any) of the following properties *must* be true?

   (a) ct is computationally indistinguishable from a uniform random string in $\{0,1\}^n$.

   (b) ct is computationally indistinguishable from $\mathsf{Encrypt}(k, 1^\lambda)$.

   (c) ct is computationally indistinguishable from $\mathsf{Encrypt}(k, r)$ where $r \xleftarrow{R} \{0,1\}^\lambda$.

   (d) ct is computationally indistinguishable from $\mathsf{Encrypt}(k, k)$.

4. **The DDH assumption.** Let $\mathbb{G}$ be a group of prime order p and generator g. Suppose the DDH assumption holds in $\mathbb{G}$. In the following, let $n = \mathrm{poly}(\lambda)$ be an arbitrary polynomial in the security parameter λ. Which (if any) of the following problems are hard in $\mathbb{G}$ (under DDH)?

   (a) The CDH problem in $\mathbb{G}$.

   (b) Sample $a, b, c, r \xleftarrow{R} \mathbb{Z}_p$. The problem is to distinguish (g^a, g^b, g^c, g^{abc}) from (g^a, g^b, g^c, g^r).

   (c) Sample $a \xleftarrow{R} \mathbb{Z}_p$. Then, for each $i \in [n]$, sample $b_i, r_i \xleftarrow{R} \mathbb{Z}_p$. The problem is to distinguish $\{(g^a, g^{b_i}, g^{ab_i})\}_{i \in [n]}$ from $\{(g^a, g^{b_i}, g^{r_i})\}_{i \in [n]}$.

   (d) For each $i \in [n]$, sample $a_i, b_i, r_i \xleftarrow{R} \mathbb{Z}_p$. The problem is to distinguish $\{(g^{a_i}, g^{b_i}, g^{a_i b_i})\}_{i \in [n]}$ from $\{(g^{a_i}, g^{b_i}, g^{r_i})\}_{i \in [n]}$.

5. **Identification protocols.** For each of the following identification protocols, indicate whether it is (i) secure against direct attacks; (ii) secure against direct attacks and passive eavesdropping attacks; (iii) secure against direct attacks, passive eavesdropping attacks, and active attacks; or (iv) none of the above. Choose **one** option for each setting.

   (a) Both the client and the server have a shared MAC key k (for a secure MAC). In the identification protocol, the server samples a random challenge $x \xleftarrow{R} \{0,1\}^\lambda$ and sends it to the client. The client responds with a MAC σ on x (using key k). The server accepts if σ is a valid MAC on x (under key k).

   (b) The client has a secret key for a semantically secure public-key encryption scheme while the server has the public key. In the identification protocol, the server samples a random challenge $x \xleftarrow{R} \{0,1\}^\lambda$ and encrypts it under the client's public key. The client decrypts the ciphertext to obtain a message x' and sends x' to the server. The server accepts if $x' = x$.

   (c) The client has a random value $x \xleftarrow{R} \{0,1\}^\lambda$ and the server has $y = f(x)$, where f is a one-way function. In the identification protocol, the client sends a NIZK proof of knowledge of x where $f(x) = y$, and the server accepts if the proof is valid.

(d) Let $\mathbb{G}$ be a group of prime order p with generator g and where the discrete log assumption holds. The client has a secret exponent $x \xleftarrow{R} \mathbb{Z}_p$ and the server has the value $h = g^x$. In the identification protocol, the client sends a NIZK proof (that is sound but *not* a proof of knowledge) that there exists $x \in \mathbb{Z}_p$ such that $h = g^x$. The server accepts if the proof is valid.

6. **Zero-knowledge.** Consider the zero-knowledge proof for graph 3-coloring from lecture that provides negligible soundness error (i.e., λ repetitions of the basic protocol). In the following, let G be a graph and let E be the set of edges in G. Let $E' \subset E$ be a subset containing exactly 3 edges (i.e., $|E'| = 3$). Both the prover and the verifier know E'. For each of the modifications described below, state the properties (if any) that still hold: (i) soundness, (ii) zero-knowledge.

   (a) The prover applies the same random permutation of colors on each repetition of the protocol.

   (b) Instead of sampling $e \xleftarrow{R} E$ in each repetition of the protocol, the verifier now samples $e \xleftarrow{R} E'$.

   (c) Instead of sampling $e \xleftarrow{R} E$ in each repetition of the protocol, the verifier now samples $e \xleftarrow{R} E \setminus E'$.

   (d) Instead of sampling $e \xleftarrow{R} E$ in each repetition of the protocol, the verifier now samples $e \xleftarrow{R} E'$ with probability $99/100$ and $e \xleftarrow{R} E \setminus E'$ with probability $1/100$.

   Note that $A \setminus B$ denotes the set of elements in A but not in B.

2 Part II: Cryptographic Primitives and Constructions

Instructions. Answer any **two** of the three problems in this section. If you answer more than two problems, only the first two you answer will be graded.

Problem 2: Cryptographic Combiners, Redux! [25 points]. Recall from Homework 2 that a cryptographic combiner is an object that takes multiple candidate instantiations of a cryptographic primitive and outputs a secure instantiation as long as *one* of the inputs is secure. In this problem, we will construct cryptographic combiners for several standard cryptographic primitives.

(a) Let $f_1 : \mathcal{X} \to \mathcal{Y}$ and $f_2 : \mathcal{X} \to \mathcal{Y}$ be efficiently-computable functions. Suppose that *either* f_1 or f_2 is one-way, but you do not know which. Use f_1 and f_2 to construct a one-way function. Prove the one-wayness of your construction (assuming *either* f_1 *or* f_2 is one-way). You are free to choose the domain and range of your one-way function.

(b) Let $(\mathsf{Setup}_1, \mathsf{Encrypt}_1, \mathsf{Decrypt}_1)$ and $(\mathsf{Setup}_2, \mathsf{Encrypt}_2, \mathsf{Decrypt}_2)$ be two candidate public-key encryption schemes on a message space $\{0,1\}^n$. Suppose the ciphertexts in both schemes are ℓ-bits long. Both of these schemes are correct, and exactly one of these two schemes is semantically secure, but you do not know which. Use these two public-key encryption schemes to construct a new public-key encryption scheme that is semantically secure as long as either one of the underlying schemes is secure and whose ciphertexts have length at most 2ℓ. Prove the security of your construction. (You should assume that $\ell \gg n$).

(c) Is your construction from Part (b) a combiner for CCA-security (assuming one of the underlying public-key encryption schemes is CCA-secure)? Give a brief and **informal** explanation (i.e., a proof sketch or an attack sketch).

Problem 3: RSA-FDH Signatures [25 points]. Recall the RSA-FDH signature scheme from class:

- The verification key $\mathsf{vk} = (N, e)$ consists of an RSA modulus $N = pq$ and a public exponent e. The signing key is a secret exponent $\mathsf{sk} = d$, where $ed = 1 \bmod \varphi(N)$. Here the bit-length of the modulus N is determined as a function of the security parameter λ (i.e., $\log N = \mathrm{poly}(\lambda)$).
- The signature on a message $m \in \{0,1\}^\ell$ is $\sigma = H(m)^d \bmod N$, where $H\colon \{0,1\}^\ell \to \mathbb{Z}_N^*$ is a hash function (modeled as a random oracle).
- To verify a signature σ on a message m, the verifier checks that $\sigma^e = H(m) \bmod N$.

We showed in class that if we model H as a random oracle, then for every efficient adversary $\mathcal{A}$ for the signature scheme, there exists an adversary $\mathcal{B}$ for the RSA assumption such that

$$\mathsf{SigAdv}[\mathcal{A}] \le Q \cdot \mathsf{RSAAdv}[\mathcal{B}] + \mathrm{negl}(\lambda),$$

where Q is a bound on the number of random oracle queries algorithm $\mathcal{A}$ makes, $\mathsf{SigAdv}[\mathcal{A}]$ is the advantage of $\mathcal{A}$ in the signature security game and $\mathsf{RSAAdv}[\mathcal{B}]$ is the advantage of $\mathcal{B}$ in breaking the RSA assumption. This means that an adversary that breaks the signature scheme with advantage ε would only break the RSA assumption with advantage roughly ε/Q. In this problem, consider the following variant of RSA-FDH signatures:

- The signing and verification keys are exactly the same as before.
- To sign a message $m \in \{0,1\}^\ell$, the signer samples $b \xleftarrow{\mathrm{R}} \{0,1\}$ and computes $\sigma' \leftarrow H(m \| b)^d \bmod N$ where $H\colon \{0,1\}^{\ell+1} \to \mathbb{Z}_N^*$ is modeled as a random oracle. The signature is $\sigma = (b, \sigma')$.
- To verify a signature $\sigma = (b, \sigma')$, the verifier checks that $(\sigma')^e = H(m \| b)$.

(a) Show that if we model H as a random oracle, then for every efficient adversary $\mathcal{A}$ for the signature scheme that only makes signing queries on *distinct* messages, there exists an efficient adversary $\mathcal{B}$ for the RSA assumption such that

$$\mathsf{SigAdv}[\mathcal{A}] \le 2 \cdot \mathsf{RSAAdv}[\mathcal{B}] + \mathrm{negl}(\lambda). \tag{1}$$

Notably, an adversary that breaks the signature scheme with advantage ε can now break the RSA assumption with advantage roughly $\varepsilon/2$.

(b) In **one sentence**, state how you would modify the above signing algorithm so security holds even against adversaries that are allowed to query for multiple signatures on the same message in the signature security game. You do *not* need to formally prove the security of your modification. It should still be the case that an advantage relation similar to Eq. (1) holds for your modified scheme (i.e., there is no dependence on the number of queries Q).

Problem 4: Commitments from Discrete Log [25 points]. Recall that a commitment scheme allows a user to commit to an input x to obtain a commitment c, and later on, provide an opening π to the value x. The commitment scheme should satisfy (1) *hiding* which says that no efficient adversary can distinguish a commitment to x_0 from a commitment to x_1; and (2) *binding* which says that no efficient adversary can produce two valid openings π_0 and π_1 for a commitment σ to messages $x_0 \ne x_1$, respectively. In this problem, we consider a generalization where the message space is a *vector* of messages. The construction operates as follows:

- Let $\mathbb{G}$ be a group of prime order p with generator g. We will show how to commit to vectors in $\mathbb{Z}_p^n$. The setup algorithm samples $h_1, \ldots, h_n \xleftarrow{\text{R}} \mathbb{G}$ and outputs the common reference string $\sigma = (g, h_1, \ldots, h_n)$.
- To commit to a vector $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{Z}_p^n$, sample a random $r \xleftarrow{\text{R}} \mathbb{Z}_p$ and output the commitment $c = g^r h_1^{x_1} \cdots h_n^{x_n}$. The opening is the tuple $\pi = (r, x_1, \ldots, x_n)$.
- An opening $\pi = (r, x_1, \ldots, x_n)$ is a valid opening to a vector $\mathbf{x} = (x_1, \ldots, x_n)$ for commitment c if $c = g^r h_1^{x_1} \cdots h_n^{x_n}$.

(a) Show that this commitment scheme satisfies hiding. Note that this property holds even against computationally unbounded adversaries.

(b) Show that if the discrete log assumption holds in $\mathbb{G}$, then this commitment scheme is binding.

(c) In the above scheme, verifying the opening requires revealing the entire vector $\mathbf{x}$. Suppose instead we wanted to *locally* open the commitment at a single coordinate x_i without revealing x_j for $j \neq i$. One way to do this is to use a zero-knowledge proof. Namely, to open a commitment c at index i to value x_i, the user provides a zero-knowledge proof of knowledge of exponents $(r, x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$ such that

$$\frac{c}{h_i^{x_i}} = g^r h_1^{x_1} \cdots h_{i-1}^{x_{i-1}} h_{i+1}^{x_{i+1}} \cdots h_n^{x_n}.$$

This relation can be shown via a generalization of Schnorr's protocol. To simplify this problem, we will just focus on the case where $n = 2$. In this case, the statement can be expressed as (g, c, h) and the goal is to prove knowledge of (r, x) such that $c = g^r h^x$. Show how to adapt Schnorr's protocol to obtain a Σ-protocol for this relation. Prove that your protocol satisfies special soundness and write down the HVZK simulator (no need to analyze the distribution for HVZK). You do *not* need to prove or analyze any other properties.

Optional Feedback. If you have any suggestions for improving future iterations of the course, please feel free to share your thoughts here!