CS255: Introduction to Cryptography **Winter 2024**

Final Exam

Instructions:

- **Please answer all five questions. You have two hours.**
- The exam is open book, open notes, open laptops, and open Internet (to consult a static online resource such as the course textbook or Wikipedia). You are expected to do the exam on your own. A web search engine, such as Google, is not allowed. **You may not interact, collaborate, or discuss the exam with another person or an AI chat bot during the exam window.**
- To submit your answers please either (i) use the provided LaTeX template, or (ii) use the provided PDF on Gradescope with a tablet and write your answers in the provided spaces, or (iii) use a paper copy of the exam and write your answers in the provided spaces. When done, please upload your solutions to Gradescope (course code `3PBEKW`) or hand the physical copy of the exam to one of the proctors. You will have time to upload your answers to Gradescope after the exam time has elapsed.
- The LaTeX template for the final is available at here. Please do not share this link with others.
- Students are bound by the Stanford honor code. In particular, you are expected to do the exam on your own.

Problem 1. (*Questions from all over*) **(20 points)**

a. Suppose $G : \{0,1\}^s \to \{0,1\}^n$ is a secure PRG. Is $G'(x) = G(x \oplus 1^s)$ a secure PRG? If so, briefly explain why by providing a security reduction. If not, describe an attack.

Your answer:

b. Suppose $F : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ is a secure PRF. Is $F'(k,x) = F(k,x) \oplus F(k,\ x \oplus 1^n)$ a secure PRF? If so, briefly explain why by providing a security reduction. If not, describe an attack.

Your answer:

c. Let (S,V) be a secure MAC with message space $\{0,1\}^n$ for some large n. Define the MAC (S',V') as

$$S'(k,m) := \big(S(k,m)\ ,\ S(k,\ m \oplus 1^n)\big) \quad \text{and}$$

$$V'\big(k,\ m,\ (t_1,t_2)\ \big) := \begin{cases} 1 & \text{if } V(k,m,t_1) = V\big(k,(m \oplus 1^n),t_2\big) = \mathsf{accept} \\ 0 & \text{otherwise} \end{cases}$$

Is this (S',V') a secure MAC? If so, briefly explain why by providing a security reduction. If not, describe an attack.

Your answer:

d. Let $\mathbb{G}$ be a group in which the Computational Diffie-Hellman problem (CDH) is easy. That is, there is an efficient algorithm $\mathcal{A}$ that for all $x, y \in \mathbb{Z}$, given $(g, g^x, g^y) \in \mathbb{G}^3$ outputs g^{xy}. Can this algorithm be used to break the ElGamal encryption system in $\mathbb{G}$? If so, explain how. If not, explain why not.

Your answer:

e. Let $\mathbb{G}$ be a cyclic group of large prime order q with generator $g \in \mathbb{G}$. Explain how to quickly calculate the 7th root of g in $\mathbb{G}$. That is, how do you find an $h \in \mathbb{G}$ such that $h^7 = g$?

Your answer: $h := g^d$ **where** d **is**

Problem 2. (*Stream ciphers*) **(20 points)** Let $G : \mathcal{K} \to \{0,1\}^n$ be a PRG. In class we defined the derived stream cipher (E_G, D_G) as a cipher defined over $(\mathcal{K}, \{0,1\}^n, \{0,1\}^n)$ that operates as $E_G(k, m) := m \oplus G(k)$ and $D_G(k, c) := c \oplus G(k)$. We showed that if G is a secure PRG then (E_G, D_G) is semantically secure.

a. Show that (E_G, D_G) is malleable. In particular, construct an adversary $\mathcal{A}(m, c) \to c'$ that takes as input $m \in \{0,1\}^n$ and $c := E_G(k, m) \in \{0,1\}^n$, and outputs a $c' \in \{0,1\}^n$ s.t. $D_G(k, c') = m \oplus 1^n$.

Your answer: $\mathcal{A}(m, c) :=$

b. Suppose that for all $k \in \mathcal{K}$, the xor of the n bits of $G(k) \in \{0,1\}^n$ is always zero (that is, $G(k)$ could output 101 but not 001). Is the derived stream cipher (E_G, D_G) semantically secure? If so prove security, otherwise describe a semantic security attacker.

Your answer:

c. Show that (E_G, D_G) is not CPA-secure by describing a CPA adversary that wins the CPA-game with advantage close to 1.

Your answer:

d. Consider the randomized cipher (E', D') defined as

$$E'(k,m) := \left\{ \begin{array}{l} r \xleftarrow{\text{R}} \{0,1\}^n,\ c_1 \leftarrow E_G(k,r),\ c_2 \leftarrow E_G(k,\ m \oplus r) \\ \text{output } c \leftarrow (c_1, c_2) \end{array} \right\}$$

$$D'(k,(c_1,c_2)) := D_G(k,c_1) \oplus D_G(k,c_2).$$

Is (E', D') CPA-secure? If so, explain why. If not, describe an attack.

Your answer:

e. Recall that a cipher provides authenticated encryption (AE) if it is (i) CPA-secure and (ii) has ciphertext integrity. Suppose (E'', D'') is a cipher for which there is an adversary $\mathcal{A}$ as in part (a): given (m, c) as input, the adversary outputs c' that decrypts to $m \oplus 1^n$. Which of the two AE properties fails for (E'', D''): CPA-security or ciphertext integrity?

Your answer:

Problem 3. (*Digital signatures*) **(20 points)**

a. Let (G, S, V) be a signature scheme where algorithm S always outputs a 64-bit signature. Describe an existential forgery attack on the scheme that requires at most 2^{64} invocations of algorithm V.

Your answer:

b. Recall that the Lamport signature scheme is a one-time signature that makes use of a one-way function $f : \mathcal{X} \to \mathcal{Y}$. To sign an n-bit message, the key generation algorithm outputs a public key containing $2n$ elements in $\mathcal{Y}$. A signature on an n-bit message m is a set of pre-images of some n elements in the public key.

Show that the length of a Lamport signature can be reduced by a factor of t at the cost of expanding the public and secret key by a factor of at most 2^t. Make sure to describe your key generation, signing, and verification algorithms. You may assume that t divides n.
Hint: In the Lamport signature scheme we sign one bit of the message at a time. Think of a way to expand the public key that lets us sign t bits of the message a time.

Your answer:

Note: one can shrink the size of Lamport signatures by a factor of t without expanding the public and secret keys. This takes a bit more work and not discussed here.

c. In Lecture 17 we described the Schnorr signature scheme. The scheme operates in a finite cyclic group $\mathbb{G}$ of prime order q with generator $g \in \mathbb{G}$. Recall that the secret key is $\mathsf{sk} := \alpha \in \mathbb{Z}_q$ and the public key is $\mathsf{pk} := g^\alpha \in \mathbb{G}$. The Sign and Verify algorithms use a hash function $H : \mathcal{M} \times \mathbb{G} \to \mathbb{Z}_q$ and work as follows

$$S(\mathsf{sk}, m) := \left\{ \begin{array}{l} r \xleftarrow{\text{R}} \mathbb{Z}_q, \quad c \leftarrow H(m, g^r) \in \mathbb{Z}_q, \quad z \leftarrow c\alpha + r \in \mathbb{Z}_q \\ \text{output } \sigma \leftarrow (c, z) \in \mathbb{Z}_q^2 \end{array} \right\}$$

$$V\big(\mathsf{pk}, m, (c, z)\big) := \big\{\text{output } \mathsf{accept} \text{ if } H(m, g^z/\mathsf{pk}^c) = c\big\}$$

Note that when computing the signature on m, the signer computes $z \leftarrow c\alpha + r$, which becomes part of the signature. Here the random nonce r acts as a "one-time pad" to hide $c\alpha$.

Suppose that a crypto library implements Schnorr signing incorrectly. Instead of sampling a fresh random r in $\mathbb{Z}_q$ for every signature, they implement r using a counter. The counter r is initialized to a random value in $\mathbb{Z}_q$ when the secret key is first generated. Then, after every issued signature, the counter is incremented by one.

Show that an adversary that observes two *consecutive* signatures $\big(m_1, (c_1, z_1)\big)$ and $\big(m_2, (c_2, z_2)\big)$, can recover the secret signing key α. By consecutive we mean that the signature (c_1, z_1) is generated using some (unknown) value r_1. The signature (c_2, z_2) is generated using the (unknown) value $r_1 + 1$.

Your answer:

d. Briefly explain why your attack from part (c) does not apply if r is sampled uniformly at random in $\mathbb{Z}_q$ for every signature.

Your answer:

Problem 4. (*Weak PRFs*) **(20 points)** Let $F : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$ be a PRF where the input space $\mathcal{X}$ is large (so that $1/|\mathcal{X}|$ is negligible). We say that F is weakly secure if the adversary cannot distinguish F from a truly random function in Funs$[\mathcal{X}, \mathcal{Y}]$ when the adversary only sees the evaluation of $F(k, \cdot)$ at *random* points in the domain $\mathcal{X}$. That is, the adversary is given pairs $\big(x_i, f(x_i)\big)$ for $i = 1, \ldots, q$ where all x_i are chosen at random in $\mathcal{X}$, and is supposed to distinguish the case where f is a truly random function (i.e., $f \xleftarrow{R}$ Funs$[\mathcal{X}, \mathcal{Y}]$) from the case where f is a random PRF instance (i.e., $f(x) := F(k, x)$ for $k \xleftarrow{R} \mathcal{K}$).

a. Write the precise security game defining a weakly secure PRF and define the advantage function for this game. Say that F is weakly secure if no efficient adversary can win the game with non-negligible advantage.

Your answer:

b. Let $F : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$ be a secure PRF (in the standard sense) and define

$$F'(k, x) := \begin{cases} k & \text{if } x = 0 \\ F(k, x) & \text{otherwise} \end{cases}$$

- Is F' a secure PRF in the standard sense? If so explain why, if not give an attack.

Your answer:

- Is F' a weakly secure PRF? If so explain why, if not give an attack.

Your answer:

c. Suppose we use a weakly secure PRF in the standard MAC-based challenge-response protocol. That is, we directly use the weakly secure PRF as the MAC in this protocol. Is the resulting authentication protocol secure against active attacks?
Hint: Give an example weakly secure PRF for which there is an active attack on the resulting challenge-response protocol. Make sure to explain how the attack works.

Your answer:

Note: it is possible to design an authentication protocol secure against active attacks from a weakly secure PRF, but we will leave that as a puzzle for another day.

Problem 5. (*Collision resistance*) **(20 points)** Let $h : \mathcal{X} \to \mathcal{Y}$ be a collision resistant hash function, where $|\mathcal{X}|$ is large.

a. Briefly explain why is it that if we want it to take time 2^{128} to find a collision for h, then we must have $|\mathcal{Y}| > 2^{256}$.

Your answer:

b. Let us show that a one-way function need not be collision resistant. Let $f : \mathcal{X} \to \mathcal{Y}$ be a one-way function. Use f to construct another function $f' : \mathcal{X} \to \mathcal{Y}$ such that f' is one-way, but is not collision resistant. Make sure to explain why your f' is one-way but not collision resistant.

Your answer:

c. Suppose that $h : \mathcal{X} \to \mathcal{X}$ is collision resistant. Is $h'(x) := h(h(x))$ also collision resistant? If so, show that collision for h' gives a collision for h. if not, give an example h that is collision resistant, but h' is not.

Your answer: