ECS 127 — Midterm 2 — Spring 2024

Instructions: The exam has this cover page then four more pages. Please write on the front side of pages only. Make your writing **clear** and **dark**—if we can't read it, it's wrong!

A reminder that you may not sit next to any partner or friend (meaning: immediately to the left, right, rear, or diagonal).

Anticipated grading (subject to change): 10 points each for problems 1–9; 40 points for problem 10, based on the number of correct responses in excess of a half.

Name:	
Student ID:	
Signature:	
Seat (eg, D15):	

1. Let $N = 10291 = 41 \cdot 251$ be the product of two primes.
   Compute $7^{20000} \pmod{10291}$.

2. A **trapdoor permutation generator** $\mathcal{F}$ is a probabilistic algorithm that, on input of a security parameter k, outputs a pair $(\_f, \_g) \twoheadleftarrow \mathcal{F}(k)$. What's the meaning of those underscores? What's the difference between $\_f$ and f?

3. Let $N = pq$ be the product of distinct 200-digit primes, and let $e, d \in \mathbb{Z}_N^*$ be inverses of one another in $\mathbb{Z}_{\phi(N)}^*$. Suppose you **sign directly with RSA**, signing $M \in \mathbb{Z}_N^*$ by $\sigma = M^d \pmod{N}$. Give an adversary $\mathcal{A}^{\mathrm{Sign}_{(N,d)}(\cdot)}(N, e)$ that forges $M = 77$.

   Hint: ask for the signatures of two messages, then output your forgery.

4. The **computational Diffie-Hellman assumption** (CDH) says that doing *what* is hard?

5. Suppose you encrypt with a **substitution cipher** $\Sigma = (\mathcal{K}, \mathcal{E}, \mathcal{D})$. Key generator $\mathcal{K}$ outputs the description of a random permutation $\pi \twoheadleftarrow \text{Perm}(8)$ specifies a permutation on bytes. The encryption of an n-byte plaintext $M_1 \cdots M_n$ is $\mathcal{E}_\pi(M_1 \cdots M_n) = \pi(M_1) \cdots \pi(M_n)$. **Now ind-break** Σ: Specify an adversary $\mathcal{A}$ whose ind-advantage $\mathbf{Adv}^{\text{ind}}_{\Sigma}(\mathcal{A}) = \Pr[\pi \twoheadleftarrow \text{Perm}(8) : \mathcal{A}^{\mathcal{E}_\pi(\cdot)} \Rightarrow 1] - \Pr[\pi \twoheadleftarrow \text{Perm}(8) : \mathcal{A}^{\mathcal{E}_\pi(0^{|\cdot|})} \Rightarrow 1]$ is large.

Simple adversary. Don't ask more than two queries.

6. Recall the **Merkle-Damgård construction** for making a cryptographic hash function H from a compression function h. Draw a picture that shows what happens when you hash a 100-byte message $M = M_1 M_2 \cdots M_{100}$. Assume that h that maps 64 bytes to 32 bytes. Assume that **length annotation** (required for Merkle-Damgård) is done by encoding $|M|$ in the last 8 bytes.

7. Define a blockcipher $E \colon \{0,1\}^{128} \times \{0,1\}^{128} \to \{0,1\}^{128}$ (make sure it *is* a blockcipher) that is **perfectly secure** (prp-advantage of 0) if the adversary asks **one** query, but is **highly insecure** (prp-advantage near 1) if the adversary asks **two** queries.

$E_K(X) =$

8. Let's use **Lamport's scheme** (lecture 9F) for a one-time, hash-based signature. Assume a hash function H that returns 32 bytes. Suppose the message you will sign is a one **byte** $M = m_1m_2m_3m_4m_5m_6m_7m_8$. The public key and secret key will be

   $pk =$

   $sk =$

   The signature of $M = 00001111$ will be

   $\sigma =$

9. **Cross out** and **fix** (reword) anything that's particularly problematic in the following. Then **explain** why you made the adjustment you did.

   A **collision-resistant hash function** (also called a *collision-intractable hash function*) is a function $H\colon \{0,1\}^* \to \{0,1\}^n$ with the property that there are no strings M and M' in the domain of H such that $M \neq M'$ yet $H(M) = H(M')$.

   Explanation:

10. **Darken the box** if the statement is **true**. Leave it alone otherwise.

1) ☐ A symmetric encryption scheme Π that is ind\$-secure will be ind-secure.

2) ☐ A symmetric encryption scheme Π that is ind-secure will be ind\$-secure.

3) ☐ A function $F\colon \mathcal{K} \times \{0,1\}^* \to \{0,1\}^{128}$ that is prf-secure will be mac-secure.

4) ☐ We know how to make a practical, provably prp-secure blockcipher.

5) ☐ We know how to make a practical, provably 2^{-128}-AU hash function.

6) ☐ OCB encryption is nonmalleable.

7) ☐ An encryption scheme with a key space smaller than its message space can achieve perfect ind-security.

8) ☐ A MAC can be secure despite being stateless and deterministic.

9) ☐ AEAD encryption $C = \mathcal{E}(K, N, A, M)$ typically produces a ciphertext whose length increases with the length of A, the associated data.

10) ☐ A Carter-Wegman MAC can authenticate a long message with only one blockcipher call.

11) ☐ A prp-secure blockcipher E might have $E_K(K) = K$.

12) ☐ If an encryption scheme's key space is smaller than its message space, it can't achieve perfect privacy.

13) ☐ ChaCha20 is an early AEAD scheme.

14) ☐ There is a message M, quite long, whose CBC MAC is always a string of zeros.

15) ☐ Let $E: \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ be a blockcipher. For any $K \in \mathcal{K}$, the function $X \mapsto E_K(X)$ is permutation, while the function $X \mapsto X \oplus E_K(X)$ is usually not.

16) ☐ A homework showed that, experimentally, RC4 seems highly secure as a PRG.

17) ☐ Adversary $\mathcal{A}$ queries a random function $f \twoheadleftarrow \{0,1\}^{128}$ at 2^{40} different points. The answers returned will probably be distinct (different from one another).

18) ☐ If Alice wants to go on a date with Bob, she should ignore what we did in ECS 127 and just ask him out.

Have a nice life!