Midterm Solutions

Question	Parts	Points
1: True or False	10	10
2: User authentication	1	10
3: CPA security	3	10
4: Hashing long messages	4	18
5: Public-key infrastructure	4	10
6: Encryption	3	12
7: Signature scheme	1	6
8: Course survey	2	4
Total:		80

Name: _____

This midterm exam is printed double-sided!

Problem 1. [10 points] **True or False** (10 parts)
Please answer **T** or **F** for the following. *No justification is needed (nor will be considered).*

(a) [1 point] If the lab 1 server is malicious, but lab 1 has been successfully completed, we can guarantee that the new device will fully synchronize all photos with the old device.

Solution: False.

(b) [1 point] Lamport's signature scheme is existential unforgeable against adaptive chosen message attacks.

Solution: False

(c) [1 point] For all secure MACs and for all CPA-secure encryption schemes, the following encryption scheme is CCA secure: encrypt m by first computing a MAC σ on m and then encrypting the pair (m, σ) using the CPA-secure encryption scheme.

Solution: False

(d) [1 point] If honest Alice and honest Bob use Diffie-Hellman key exchange, they will agree on a common shared secret, even if there is a network adversary that can tamper with the messages they exchange.

Solution: False

(e) [1 point] A secure encryption scheme may leak information about the message size.

Solution: True

(f) [1 point] AES-GCM is secure against chosen-plaintext attacks.

Solution: True

(g) [1 point] SHA-256 is a MAC that is existentially unforgeable under chosen-message attacks.

Solution: False

(h) [1 point] If a hash function $H\colon \{0,1\}^* \to \{1,\ldots,N\}$ is collision resistant against attacks that run in time $O(T)$, it must be that $N \geq T^2$.

Solution: True

(i) [1 point] HTTPS/TLS uses "trust on first use" for public-key distribution.

Solution: False

(j) [1 point] If public-key encryption exists then one-way functions exist.

Solution: True

Problem 2. [10 points] **User authentication** (1 part)
Ben Bitdiddle is developing a Twitter-like application. He doesn't want to send passwords over the network, since a network adversary can observe them in transit. Instead, his plan is to use a key-derivation function KDF (effectively just a hash function) to convert the user's password P into a key K, and then use a MAC to authenticate the user's network messages using that key K. Specifically, to post a tweet T in Ben's application, the user's device sends the following message to the server:

$$\mathrm{Post}(U, T, \mathrm{MAC}(\mathrm{KDF}(P), (U, T)))$$

where U is the user that is trying to send this tweet.

What would be the most efficient way for an adversary to gain access to a victim's account (say, posting a message of the adversary's choice)? Assume that the adversary controls the network and can observe the messages sent by the victim when they are posting their tweets, and that the adversary's only power is being able to send and receive network messages.

Solution: Try to brute-force the victim's password by checking whether any given password guess matches the MAC sent over the network by the victim's machine. The time taken for this attack depends on the entropy of the victim's password; for each guess, the adversary needs to compute the KDF and the MAC.

Problem 3. [10 points] **CPA security** (3 parts)
Recall the following variant of the El-Gamal encryption scheme presented in class (for a fixed group G, a fixed generator g and a fixed hash function H): The key generation algorithm generates a key pair $(pk, sk) = (g^a, a)$, and the encryption algorithm encrypts a message m by choosing a random b in $\{1, \ldots, |G|\}$, and outputting $(g^b, H(g^{ab}) \oplus m)$.

(a) [2 points] Under what assumption is this scheme CPA secure (assuming H is a random oracle)?

Solution: CDH assumption.

(b) [4 points] Is it CCA secure? If it is secure then explain why, and if not provide an attack.

Solution: No, an attacker can flip the last bit of the challenge ciphertext, send that to the decryption oracle and thus learn a lot of information about the challenge ciphertext.

(c) [4 points] Consider the variant where we omit H altogether from the encryption scheme, and encrypt m by choosing a random b in $\{1, \ldots, |G|\}$, and outputting $(g^b, g^{ab} \cdot m)$ where we assume that the message m is an elelement in G. Under what assumption is this variant CPA secure? Is it CCA secure? (No need to explain your answer.)

Solution: It is CPA secure under DDH, and is not CCA secure.

Problem 4. [18 points] **Hashing long messages** (4 parts)
Let $H\colon \{0,1\}^* \to \{0,1\}^n$ be a collision-resistant hash function. Let $|x|$ denote the length of the bitstring x and let $\|$ denote string concatenation.

(a) [4 points] Say that Ben Bitdiddle finds a bitstring x. Explain why changing a single bit of x must change at least one bit of $H(x)$.

Solution: Towards a contradiction: Say that $H(x) = H(x')$ where x and x' differ on a single bit. Then (x, x') is a collision for H. Contradiction.

(b) [4 points] Define $H_2\colon \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^n$ as $H_2(x,y) := H(x\|y\|x)$. Write down inputs (a_0, a_1) and (b_0, b_1) that form a collision in H_2.

Solution: Let ϵ denote the empty string. Then $(\epsilon, \text{“00”})$ and $(\text{“0”}, \epsilon)$ form a collision for H_2.

(c) [6 points] Show how to use H to construct a collision-resistant hash function $H_3 \colon \{0,1\}^* \times \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^n$.

Solution: Prepend each of the three inputs with their length using $\lceil \log_2 n \rceil$ bits each. Then concatenate the three values and run them through H.

(d) [4 points] Give an algorithm that finds a collision in the hash function H using $O(2^n)$ invocations of H. Explain why your algorithm succeeds with probability 1.

Problem 5. [10 points] **Public-key infrastructure** (4 parts)
Alice is designing a new type of public-key infrastructure (PKI) for the web. In her design, there is a centralized server that maintains the mapping of domain names to public keys. The web browser fetches the public key for each domain directly from the key server.

(a) [2 points] Say that we used Alice's PKI to replace certificates on the web. Explain one performance problem with Alice's approach.

Solution: The public-key server is on the critical path to viewing a website.

(b) [2 points] Explain how certificate revocation could work in Alice's scheme. Explain why revocation is easier or more challenging to implement in this centralized scheme.

Solution: The server can just change the key for a name at any time. Revocation is easier here.

(c) [3 points] Alice plans to use digital signatures to integrity-protect the communications between the end user and the key server. The key server can afford to use signatures of length at most 100 bytes. What digital-signature scheme should Alice use?

Solution: EC-DSA.

(d) [3 points] Alice is not planning to use encryption to secure the communications between the end user and Alice's key server. Explain why the non-use of encryption can or cannot help an attacker who is trying to cause the end user to recover the incorrect public key for a domain name.

Solution: As long as there is integrity protection (e.g., digital signatures), there is no need for encryption.

Problem 6. [12 points] **Encryption** (3 parts)

(a) [3 points] Say that you are given a secure MAC MAC, a collision-resistant hash function H, and a chosen-plaintext-secure encryption scheme $\mathsf{Enc}_{\mathsf{CPA}}$. All of these primitives operate on messages in $\{0,1\}^*$. Explain how you can use these primitives to construct a chosen-ciphertext-secure symmetric-key encryption scheme Enc using the fewest possible invocations of these primitives.

Solution: Use encrypt-then-MAC.

(b) [5 points] Say that you are given *only* a secure pseudorandom function F. Is it possible to construct a CCA-secure encryption scheme using only F? Explain in 2-3 sentences why this is or is not possible.

Solution: Yes, it is possible. PRFs imply MACs and symmetric-key encryption, which is enough to build authenticated encryption via encrypt-then-MAC.

(c) [4 points] Let Enc be a deterministic encryption scheme. Give an attack that explains why Enc cannot be secure against chosen-plaintext attacks.

Solution: Ask for encryption of 0 and 1, receive c_0 and c_1. Then give the challenger $(0,1)$ as the challenge messages. Receive challenge ciphertext c^* and output 0 if $c_0 = c^*$ and 1 otherwise.

Problem 7. [6 points] **Signature scheme** (1 part)
Show how to use a collisison resistant hash function $h : \{0,1\}^* \to \{0,1\}^{256}$ to convert any signature scheme with message space $\{0,1\}^{256}$ into a signature with message space $\{0,1\}^*$.

Problem 8. [4 points] **Course survey** (2 parts)

(a) [2 points] What changes or improvements would you like to see in the second half of the class?

(b) [2 points] How could we improve the assignments (problem sets and labs) you have done so far?