Q1 *MAC Madness* (18 points)

Evan wants to store a list of every CS161 student's `firstname` and `lastname`, but they are afraid that Mallory will tamper with their list.

Evan is considering adding a cryptographic value to each record to ensure its integrity. For each scheme, determine what Mallory can do without being detected.

Assume MAC is a secure MAC, H is a cryptographic hash, and Mallory does not know Evan's secret key k. Assume that `firstname` and `lastname` are all lowercase and **alphabetic** (no numbers or special characters), and concatenation does not add any delimiter (e.g. a space or tab), so $\texttt{nick}\|\texttt{weaver} = \texttt{nickweaver}$.

Q1.1 (3 points) $\mathsf{H}(\texttt{firstname}\|\texttt{lastname})$

- ● Mallory can modify a record to be a value of her choosing
- ○ Mallory can modify a record to be a specific value (not necessarily of her choosing)
- ○ Mallory cannot modify a record without being detected

Solution: Anybody can hash a value, so Mallory could change a record to be whatever she wants and compute the hash of her new record.

Q1.2 (3 points) $\mathsf{MAC}(k, \texttt{firstname}\|\texttt{lastname})$

Hint: Can you think of two different records that would have the same MAC?

- ○ Mallory can modify a record to be a value of her choosing
- ● Mallory can modify a record to be a specific value (not necessarily of her choosing)
- ○ Mallory cannot modify a record without being detected

Solution: Because the concatenation doesn't have any indicator of where the first name ends and the last name begins, Mallory could shift some letters between the first name and last name. For example, she could change the name Nick Weaver to Ni Ckweaver, Nic Kweaver, Nickw Eaver, etc. Since the MAC would remain unchanged, this edit would be undetectable.

Q1.3 (3 points) $\mathsf{MAC}(k, \texttt{firstname} \| \texttt{"-"} \| \texttt{lastname})$, where "-" is a hyphen character

○ Mallory can modify a record to be a value of her choosing

○ Mallory can modify a record to be a specific value (not necessarily of her choosing)

● Mallory cannot modify a record without being detected

Solution: Since the names are alphabetic, they would never include a dash in them. Hence, the dash serves as a separator between the first name and last name, so the attack from the previous part is no longer possible.

Q1.4 (3 points) $\mathsf{MAC}(k, \mathsf{H}(\texttt{firstname}) \| \mathsf{H}(\texttt{lastname}))$

○ Mallory can modify a record to be a value of her choosing

○ Mallory can modify a record to be a specific value (not necessarily of her choosing)

● Mallory cannot modify a record without being detected

Solution: Because the hashes produce a fixed-length value, concatenating them within the MAC without delimiters does not violate integrity.

Q1.5 (3 points) $\mathsf{MAC}(k, \texttt{firstname}) \| \mathsf{MAC}(k, \texttt{lastname})$

○ Mallory can modify a record to be a value of her choosing

● Mallory can modify a record to be a specific value (not necessarily of her choosing)

○ Mallory cannot modify a record without being detected

Solution: Because the last name and the first name have separate MACs, Mallory could swap the first name and the last name, and swap the two halves of the MAC.

In other words, Mallory could change the name Nick Weaver to Weaver Nick, and change the MAC from $\mathsf{MAC}(k, \texttt{nick}) \| \mathsf{MAC}(k, \texttt{weaver})$ to $\mathsf{MAC}(k, \texttt{weaver}) \| \mathsf{MAC}(k, \texttt{nick})$.

Q1.6 (3 points) Which of Evan's schemes guarantee confidentiality on his records?

○ All 5 schemes

○ Only the schemes with a hash

○ Only the schemes with a MAC

● None of the above

Solution: MACs and hashes do not have any confidentiality guarantees.

Q2 *EU-CMA* **(9 points)**

Alice and Bob want to communicate with confidentiality and integrity. They have:

- Symmetric Encryption:
  - Encryption: $\mathsf{Enc}(K, m)$
  - Decryption: $\mathsf{Dec}(K, m)$
- Cryptographic Hash Function: $\mathsf{Hash}(m)$
- MAC: $\mathsf{MAC}(K, m)$

They share a symmetric key K and know each other's public key.

We assume these cryptographic tools do not interfere with each other when used in combination; *i.e.,* we can safely use the same key for encryption and MAC.

1. $t_i = \mathsf{Hash}(t_{i-1}, m_i), t_0 = \mathsf{Hash}(K)$
2. $t = \mathsf{Hash}(m_i)$
3. $t = \mathsf{MAC}(K, m)$

Q2.1 (3 points) Is scheme 1 EU-CMA secure?

○ Yes (leave the boxes blank)

● No (fill in the boxes)

m : M' t : $\mathsf{Hash}\ (M', t_{i-1})$

Solution:

Q2.2 (3 points) Is scheme 2 EU-CMA secure?

○ Yes (leave the boxes blank)

● No (fill in the boxes)

m : x t : $\mathsf{Hash}(x)$

Solution:

Q2.3 (3 points) Is scheme 3 EU-CMA secure?

● Yes (leave the boxes blank)

○ No (fill in the boxes)

m : 　　　　　　t :

Solution:

Q3 *Why do RSA signatures need a hash?* **(3 points)**

To generate RSA signatures, Alice first creates a standard RSA key pair: (n, e) is the RSA public key and d is the RSA private key, where n is the RSA modulus. For standard RSA signatures, we typically set e to a small prime value such as 3; for this problem, let $e = 3$.

Suppose we used a **simplified** scheme for RSA signatures that skips using a hash function and instead uses message M directly, so the signature S on a message M is $S = M^d \bmod n$. In other words, if Alice wants to send a signed message to Bob, she will send (M, S) to Bob where $S = M^d \bmod n$ is computed using her private signing key d.

Q3.1 (1 point) With this **simplified** RSA scheme, how can Bob verify whether S is a valid signature on message M? In other words, what equation should he check, to confirm whether M was validly signed by Alice?

> **Solution:** $S^3 = M \bmod n$.

Q3.2 (1 point) Mallory learns that Alice and Bob are using the **simplified** signature scheme described above and decides to trick Bob into believing that one of Mallory's messages is from Alice. Explain how Mallory can find an (M, S) pair such that S will be a valid signature on M.

You should assume that Mallory knows Alice's public key n, but not Alice's private key d. The message M does not have to be chosen in advance and can be gibberish.

> **Solution:** Mallory should choose some random value to be S and then compute $S^3 \bmod n$ to find the corresponding M value. This (M, S) pair will satisfy the equation in part (a).
>
> **Alternative solution:** Choose $M = 1$ and $S = 1$. This will satisfy the equation.

Q3.3 (1 point) Is the attack in Q1.2 possible against the **standard** RSA signature scheme (the one that includes the cryptographic hash function)? Why or why not?

> **Solution:** This attack is not possible. A hash function is one way, so the attack in part (b) won't work: we can pick a random S and cube it, but then we'd need to find some message M such that $H(M)$ is equal to this value, and that's not possible since H is one-way.
>
> Comment: This is why the real RSA signature scheme includes a hash function: exactly to prevent the attack you've seen in this question.