Cryptographic Hashes and MACs

Question 1 ***Crytographic Hashes*** ()

For each of the given functions H below, determine if it is one-way or not, and if it is collision-resistant or not.

Q1.1 $\mathsf{H}(x) = x^2$

◯ (A) One way

◯ (B) Collision resistant

◯ (C) Both

● (D) Neither

Solution: This function is not collision-resistant. Consider $H(1) = H(-1) = 1$.

This function is not one-way because given $H(x)$, we can calculate $\sqrt{(H(x))} = \sqrt{(x^2)} = x$.

Q1.2 For this part you have access to a SHA-256 hash function. The notation $[x : y]$ refers to a slice of bytes x to $y - 1$.

$\mathsf{H}(x) = \text{SHA-256}(x[0 : \mathsf{len}(x) - 1])$

● (G) One way

◯ (H) Collision resistant

◯ (I) Both

◯ (J) Neither

Solution: This function is not collision-resistant. Consider the values of $x =$ "abc" and $x =$ "abd". As defined by the hash function, we take the first len(x) - 1 bytes and pass that into the SHA-256 hash function. Therefore both vales of x would become SHA-256("ab") and have the same hash value.

The function is one way because SHA-3 is one way and knowing the output of $H(x)$ does not tell us about the input x.

Q1.3 $\mathsf{H}(x) = x^3$

◯ (A) One way

● (B) Collision resistant

◯ (C) Both

◯ (D) Neither

Solution: This function is collision-resistant because the function x^3 is monotonically increasing and no two values of x will have the same output.

This function is not one-way similar to the reasoning in part 1. Given $H(x)$, we can calculate $\sqrt[3]{(H(x))} = \sqrt[3]{(x^3)} = x$.

Question 2 ***MAC Madness*** **(18 min)**

Evan wants to store a list of every CS161 student's firstname and lastname, but he is afraid Mallory will tamper with his 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 that usernames must be unique.

Q2.1 (3 points) $\mathsf{H}(\mathsf{firstname}\|\mathsf{lastname})$

● (A) Mallory can modify a record to be a value of her choosing

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

○ (C) Mallory cannot modify a record without being detected

○ (D) —

○ (E) —

○ (F) —

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.

Q2.2 (3 points) $\mathsf{MAC}(k, \mathsf{firstname}\|\mathsf{lastname})$

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

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

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

○ (I) Mallory cannot modify a record without being detected

○ (J) —

○ (K) —

○ (L) —

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.

Q2.3 (3 points) $\mathsf{MAC}(k, \mathsf{firstname} \| \texttt{"-"} \| \mathsf{lastname})$, where "-" is a hyphen character.

◯ (A) Mallory can modify a record to be a value of her choosing

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

● (C) Mallory cannot modify a record without being detected

◯ (D) —

◯ (E) —

◯ (F) —

Solution: Now, the concatenation includes a separator between first name and last name, so the attack from the previous part is no longer possible. Note that names are alphabetical, so they would never include a dash in them.

Q2.4 (3 points) $\mathsf{MAC}(k, \mathsf{H}(\mathsf{firstname}) \| \mathsf{H}(\mathsf{lastname}))$

◯ (G) Mallory can modify a record to be a value of her choosing

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

● (I) Mallory cannot modify a record without being detected

◯ (J) —

◯ (K) —

◯ (L) —

Solution: Hashes have fixed-length output, so the attack from the previous part (shifting letters between the first and last name) is not possible here either. It will always be unambiguous where the first hash ends and the second hash begins.

Also, since both hashes are used as input to a single MAC, there is no way for an attacker without the key to generate a valid MAC for any different name.

Q2.5 (3 points) $\mathsf{MAC}(k, \mathsf{firstname}) \| \mathsf{MAC}(k, \mathsf{lastname})$

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

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

○ (C) Mallory cannot modify a record without being detected

○ (D) —

○ (E) —

○ (F) —

Solution: Because the first name and last name have separate MACs, Mallory could swap the first name and 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 $\mathsf{MAC}(k, \mathsf{nick}) \| \mathsf{MAC}(k, \mathsf{weaver})$ to $\mathsf{MAC}(k, \mathsf{weaver}) \| \mathsf{MAC}(k, \mathsf{nick})$.

Q2.6 (3 points) Which of Evan's schemes guarantee confidentiality on his records?

○ (G) All 5 schemes

○ (H) Only the schemes with a MAC

○ (I) Only the schemes with a hash

● (J) None of the schemes

○ (K) —

○ (L) —

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

Question 3 ***Confidentiality and integrity*** ()

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

- Symmetric encryption.
  - Encryption: $\mathsf{Enc}(\mathsf{K}, \mathsf{m})$.
  - Decryption: $\mathsf{Dec}(\mathsf{K}, \mathsf{c})$.
- Cryptographic hash function: $\mathsf{Hash}(\mathsf{m})$.
- MAC: $\mathsf{MAC}(\mathsf{K}, \mathsf{m})$.

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.

Alice sends to Bob
1. $\mathsf{c} = \mathsf{Hash}(\mathsf{Enc}(\mathsf{K}, \mathsf{m}))$
2. $\mathsf{c} = \mathsf{c}_1, \mathsf{c}_2$: where $\mathsf{c}_1 = \mathsf{Enc}(\mathsf{K}, \mathsf{m})$ and $\mathsf{c}_2 = \mathsf{Hash}(\mathsf{Enc}(\mathsf{K}, \mathsf{m}))$
3. $\mathsf{c} = \mathsf{c}_1, \mathsf{c}_2$: where $\mathsf{c}_1 = \mathsf{Enc}(\mathsf{K}, \mathsf{m})$ and $\mathsf{c}_2 = \mathsf{MAC}(\mathsf{K}, \mathsf{m})$
4. $\mathsf{c} = \mathsf{c}_1, \mathsf{c}_2$: where $\mathsf{c}_1 = \mathsf{Enc}(\mathsf{K}, \mathsf{m})$ and $\mathsf{c}_2 = \mathsf{MAC}(\mathsf{K}, \mathsf{Enc}(\mathsf{K}, \mathsf{m}))$

Q3.1 Which ones of them can Bob decrypt?

☐ 1 ☐ 2 ☐ 3 ☐ 4

Solution: Bob cannot decrypt Scheme 1 because he cannot invert Hash.

In sum: 2-4

Q3.2 Consider an eavesdropper Eve, who can see the communication between Alice and Bob.

Which schemes, of those decryptable in (a), also provide *confidentiality* against Eve?

☐ 1 ☐ 2 ☐ 3 ☐ 4

Solution: Scheme 3 does not provide confidentiality because the MAC is sent in plaintext. For the same message, the MAC is the same, thus leaky.

In sum: 2, 4

Q3.3 Consider a man-in-the-middle Mallory, who can eavesdrop and modify the communication between Alice and Bob.

Which schemes, of those decryptable in (a), provide *integrity* against Mallory?
i.e., Bob can detect any tampering with the message?

☐ 1 ☐ 2 ☐ 3 ☐ 4

Solution: Scheme 2 does not provide integrity as Mallory can forge a message by sending Bob $(c', \mathsf{Hash}(c'))$.

In sum: 3, 4

Q3.4 Many of the schemes above are insecure against a *replay attack*.

If Alice and Bob use these schemes to send many messages, and Mallory remembers an encrypted message that Alice sent to Bob, some time later, Mallory can send the exact same encrypted message to Bob, and Bob will believe that Alice sent the message *again*.

How to modify those schemes with confidentiality & integrity to prevent replay attack?

◇ The scheme providing confidentiality & integrity is Scheme ☐.

The modification is:

Solution: Add a non-repeating nonce or timestamp in the MAC.

In sum: 4, we replace message m with $\mathsf{timestamp} \,\|\, \mathsf{m}$.