Cryptography II

Question 1 ***Diffie–Hellman key exchange*** **(15 min)**

Recall that in a Diffie-Hellman key exchange, there are values a, b, g and p. Alice computes $g^a \bmod p$ and Bob computes $g^b \bmod p$.

(a) Which of these values (a, b, g, and p) are publicly known and which must be kept private?

(b) Mallory can eavesdrop, intercept, and modify everything sent between Alice and Bob. Alice and Bob perform Diffie-Hellman to agree on a shared symmetric key k. After the exchange, Bob gets the feeling something went wrong and calls Alice. He compares his value of k to Alice's and realizes that they are different. Explain what Mallory has done.

(c) Alice and Bob want to prevent Mallory from tampering with their keys by attaching a hash to each message (ie. Alice sends $(g^a, H(g^a))$ and Bob sends $(g^b, H(g^b))$). Does this successfully stop Mallory?

Question 2 ***Perfect Forward Secrecy*** **(15 min)**

Alice (A) and Bob (B) want to communicate using some shared symmetric key encryption scheme. Consider the following key exchange protocols which can be used by A and B to agree upon a shared key, k.

El Gamal-Based Key Exchange

Message 1	$A \rightarrow B$:	$\{k\}_{PK_B}$
	Key exchanged	
Message 2	$A \leftarrow B$:	$\{secret1\}_k$
Message 3	$A \rightarrow B$:	$\{secret2\}_k$

Diffie-Hellman Key Exchange

Message 1	$A \rightarrow B$:	$g^a \mod p$
Message 2	$A \leftarrow B$:	$g^b \mod p$
	Key exchanged	
	$k = g^{ab} \mod p$	
Message 3	$A \leftarrow B$:	$\{secret1\}_k$
Message 4	$A \rightarrow B$:	$\{secret2\}_k$

Some additional details:

- PK_B is Bob's long-lived public key.
- k, the Diffie-Hellman exponents a and b, and the messages themselves are destroyed once all messages are sent. That is, these values are not stored on Alice and Bob's devices after they are done communicating.

Eavesdropper Eve records all communications between Alice and Bob, but is unable to decrypt them. At some point in the future, Eve is lucky and manages to compromise Bob's computer.

(a) Is the confidentiality of Alice and Bob's prior El Gamal-based communication in jeopardy?

(b) What about Alice and Bob's Diffie-Hellman-based communication?

Question 3 ***Hashing passwords with salts*** **(15 min)**

When storing a password pw, a website generates a random string salt, and saves:

$$(\mathsf{salt}, \mathsf{Hash}(\mathsf{pw} \parallel \mathsf{salt}))$$

in the database, where Hash is a cryptographic hash function.

(a) If a user tries to log in with password pw' (which may or may not be the same as pw), how does the site check if the user has the correct password?

(b) Why use a hash function Hash rather than just store pw directly?

(c) Suppose the site doesn't use a salt and just stores $\mathsf{Hash}(\mathsf{pw})$. What attack becomes easier?

(d) Suppose the site has two candidate hash functions Hash_1 and Hash_2. Their properties are shown in the table below.

Function	One-Way	Collision Resistant
Hash_1	Yes	No
Hash_2	Yes	Yes

Which of them suffice for password hashing?