

**Question 1 Public Key Encryption**

0

The El Gamal encryption scheme is reproduced below:

- **Key Generation:** public key =  $(g, h, p)$ , where  $h = g^k \pmod{p}$ , private key =  $k$
- **Encryption:**  $c = (c_1, c_2) = (g^r \pmod{p}, m \times h^r \pmod{p})$ , where  $r$  is randomly sampled from  $\{1, \dots, p-1\}$ .
- **Decryption:**  $m = c_1^{-k} \times c_2 \pmod{p}$

Look at each scenario below and select the appropriate options.

Q1.1 With El Gamal, it is not a problem if the adversary can learn the value of  $g$  somehow.

- (A) True
- (B) False
- (C) —
- (D) —
- (E) —
- (F) —

**Solution:**  $g$  is part of the public key, so it is fine for it to be known to the public (including the adversary).

Q1.2 With El Gamal, it is not a problem if the value  $r$  used during encryption is accidentally revealed after the encryption is complete.

- (G) True
- (H) False
- (I) —
- (J) —
- (K) —
- (L) —

**Solution:** If the adversary learns  $r$ , they can compute  $c_2 h^{-r} \pmod{p}$ , and that will reveal the message  $m$ .

Now imagine that Alice (A) and Bob (B) want to communicate over an insecure network and they know each other's public key. Consider the following message exchange:

A: Hey Bob, it's Alice. How many dollars do I owe you?

B: 10000

The message is encrypted with Alice's public key using ElGamal encryption.

Alice decrypted this successfully, but suddenly remembered that she only owed Bob \$100.

Q1.1 Assume Bob would not lie. How did an attacker tamper with the message?

**Solution:** The attacker multiplied  $c_2$  by 100, or multiplied  $c_1 \cdot c'_1, c_2 \cdot c'_2$  where  $c'$  is a valid encryption of 100, or they encrypted an entirely new message.

Q1.2 What could Bob have additionally sent that would've stopped this attack?

**Solution:** Bob could attach a signature to his original message.

**Question 2** *Why do RSA signatures need a hash?*

()

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  $d$ .

Q2.1 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:**  $S^3 = M \bmod n$ .

Q2.2 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.

Q2.3 Is the attack in Q3.2 possible against the **standard** RSA signature scheme (the one that includes the cryptographic hash function)? Why or why not?

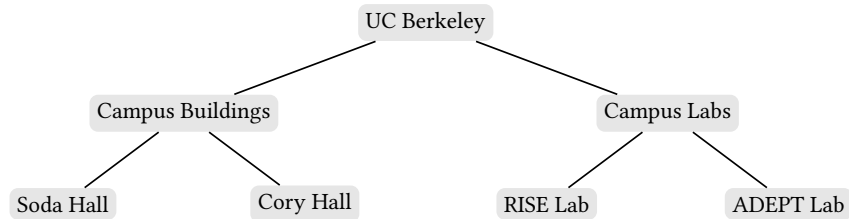
**Solution:** This attack is not possible. A hash function is one-way, so the attack in part (b) won't work: we can pick a random  $S$  and cube it, but then we'd need to find some message  $M$  such that  $H(M)$  is equal to this value, and that's not possible since  $H$  is one-way.

Comment: This is why the real RSA signature scheme includes a hash function: exactly to prevent the attack you've seen in this question.

### Question 3 *RISELab Shenanigans*

()

Certificate authorities of UC Berkeley are organized in a hierarchy as follows:



Alice is a student in RISELab at UC Berkeley and wants to obtain a certificate for her public key. Assume that only RISELab is allowed to issue certificates to Alice.

Q3.1 (2 min) Which of the following values are included in the certificate issued to Alice? Select all that apply.

- (A) Alice's public key
- (B) Alice's private key
- (C) A signature on Alice's *public* key, signed by RISELab's private key
- (D) A signature on Alice's *private* key, signed by RISELab's private key
- (E) None of the above
- (F) —

**Solution:** This follows from the definition of certificates: they include a user's public key, and a signature on the enclosed public key, signed by the issuer (which we state in the prologue is RISELab).

Q3.2 (2 min) Assume that the only public key you trust is UC Berkeley's public key. Which certificates do you need to verify in order to be sure that you have Alice's public key? Select all that apply.

- (G) Certificate for Alice
- (H) Certificate for Soda Hall
- (I) Certificate for RISELab
- (J) Certificate for Campus Labs
- (K) None of the above
- (L) —

**Solution:** To validate Alice's public key, we can follow our way up to our root of trust (which is UC Berkeley's public key). As such, we need certificates for Alice, RISELab, and Campus Labs.

Q3.3 (4 min) RISELab issues a certificate to Alice that expires in 1 hour. Which of the following statements are true about using such a short expiration date? Select all that apply.

- (A) It mitigates attacks where Alice's private key is stolen
- (B) It mitigates attacks where RISELab's private key is stolen
- (C) It mitigates attacks where Campus Labs' private key is stolen
- (D) It forces Alice to renew the certificate more often
- (E) None of the above
- (F) —

**Solution:** Short expiration times only mitigate the situation where Alice's private key is stolen. If RISELab's private key is compromised, the attacker can issue certificates with any expiration date, and it is up to the parent CA to revoke RISELab's certificate, not RISELab itself. The same argument applies to Campus Labs' private key.

The following subparts are independent from the previous subparts.

Passwords on the RISELab website are six-digit codes, where each digit is one of 0–9 (repeat digits are allowed). An attacker steals the password database, which includes Alice's hashed password, and wants to learn Alice's password.

For each password storage scheme, *in the worst case*, how much time would it take for the attacker to brute-force Alice's password?

Assumptions:

- The attacker tries passwords one at a time.
- $H$  is a hash function that takes 1 second to compute.
- The time required for all other operations is negligible.

Q3.4 (2 min) Passwords are stored as  $H(\text{pwd})$ .

- (G)  $10^6 \cdot 2 \cdot 8$  seconds     
  (I)  $10^6 \cdot 2^8$  seconds     
  (K)  $2^8$  seconds  
 (H)  $6 \cdot 10 \cdot 2^8$  seconds     
  (J)  $10^6$  seconds     
  (L) —

**Solution:** Since the password is six-digits, and there are 10 possibilities for each digit, the attacker must try  $10^6$  possible passwords in the worst case.

Q3.5 (2 min) Passwords are stored as  $(\text{salt}, H(\text{salt}||\text{pwd}))$ , where salt is an 8-bit random string.

- (A)  $10^6 \cdot 2 \cdot 8$  seconds     
  (C)  $10^6 \cdot 2^8$  seconds     
  (E)  $2^8$  seconds  
 (B)  $6 \cdot 10 \cdot 2^8$  seconds     
  (D)  $10^6$  seconds     
  (F) —

**Solution:** Since the attacker knows the salt—it's stored next to the password in plaintext—the worst-case number of tries the attacker must attempt doesn't change from the previous subpart: the answer is  $10^6$  once again.

Q3.6 (4 min) Assume that the attacker is conducting an **online** brute-force attack against Alice's account. Which of the following changes, if implemented individually, would make it more difficult for the attacker to access Alice's account? Select all that apply.

(G) Alice uses a random, alphanumeric, 32-character password instead of a 6-digit numeric password

(H) Alice enables two-factor authentication on her account

(I) RISELab imposes a timeout which doubles after each password attempt

(J) RISELab enables TLS for its login page

(K) None of the above

(L) —

**Solution:** Using a longer password with more possibilities would make it more difficult to brute force.

Enabling 2FA would prevent an attacker from compromising Alice's account even if the attacker managed to brute force Alice's password.

Enabling a timeout would successfully mitigate an online brute force attack – especially one where the timeout doubles after each attempt!

However, enabling TLS wouldn't make it more difficult to prevent an online brute-force attack; the attacker's sole goal in brute force is to access Alice's account by trying many, many times; we didn't specify that this attacker had any other information (e.g. we didn't say the attacker was on-path, etc.), so TLS wasn't a valid answer choice here.