Question 1 ***Dual Asymmetry*** ()

Alice wants to send two messages M_1 and M_2 to Bob, but they do not share a symmetric key.

Assume that p is a large prime and that g is a generator $\bmod p$, like in ElGamal. Assume that all computations are done modulo p in Scheme A.

Q1.1 Scheme A: Bob publishes his public key $B = g^b$. Alice randomly selects r from 0 to p - 2. Alice then sends the ciphertext $(R, S_1, S_2) = (g^r, M_1 \times B^r, M_2 \times B^{r+1})$.

Select the correct decryption scheme for M_1:

- ○ $R^{-b} \times S_1$
- ○ $B^{-b} \times S_1$
- ○ $R^b \times S_1$
- ○ $B^b \times S_1$

Q1.2 Select the correct decryption scheme for M_2:

- ○ $B^{-1} \times R^{-b} \times S_2$
- ○ $B^{-1} \times R^b \times S_2$
- ○ $B \times R^{-b} \times S_2$
- ○ $B^{-1} \times R \times S_2$

Q1.3 Is Scheme A IND-CPA secure? If it is secure, briefly explain why (1 sentence). If it is not secure, briefly describe how you can learn something about the messages.

Clarification during exam: For Scheme A, in the IND-CPA game, assume that a single plaintext is composed of two parts, M_1 and M_2.

- ○ Secure
- ○ Not secure

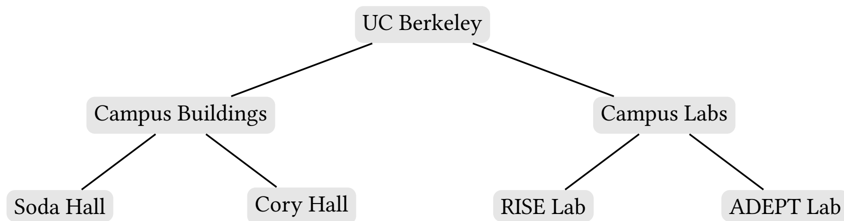
Q1.4 Scheme B: Alice randomly chooses two 128-bit keys K_1 and K_2. Alice encrypts K_1 and K_2 with Bob's public key using RSA (with OAEP padding) then encrypts both messages with AES-CTR using K_1 and K_2. The ciphertext is RSA$(\mathsf{PK}_{\mathsf{Bob}}, K_1 \| K_2)$, Enc$(K_1, M_1)$, Enc$(K_2, M_2)$.

Which of the following is required for Scheme B to be IND-CPA secure? Select all that apply.

- ☐ K_1 and K_2 must be different
- ☐ A different IV is used each time in AES-CTR
- ☐ M_1 and M_2 must be different messages
- ☐ M_1 and M_2 must be a multiple of the AES block size
- ☐ M_1 and M_2 must be less than 128 bits long
- ☐ None of the above

Question 2 ***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.

Q2.1 (2 min) Which of the following values are included in the certificate issued to Alice? Select all that apply.

- ☐ Alice's public key
- ☐ Alice's private key
- ☐ A signature on Alice's *public* key, signed by RISELab's private key
- ☐ A signature on Alice's *private* key, signed by RISELab's private key
- ☐ None of the above

Q2.2 (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.

- ☐ Certificate for Alice
- ☐ Certificate for Soda Hall
- ☐ Certificate for RISELab
- ☐ Certificate for Campus Labs
- ☐ None of the above

Q2.3 (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.

- ☐ It mitigates attacks where Alice's private key is stolen
- ☐ It mitigates attacks where RISELab's private key is stolen
- ☐ It mitigates attacks where Campus Labs' private key is stolen
- ☐ It forces Alice to renew the certificate more often
- ☐ None of the above

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.

Q2.4 (2 min) Passwords are stored as $H(\mathsf{pwd})$.

- ○ $10^6 \cdot 2 \cdot 8$ seconds
- ○ $10^6 \cdot 2^8$ seconds
- ○ 2^8 seconds
- ○ $6 \cdot 10 \cdot 2^8$ seconds
- ○ 10^6 seconds

Q2.5 (2 min) Passwords are stored as $(\mathsf{salt}, H(\mathsf{salt} \| \mathsf{pwd}))$, where salt is an 8-bit random string.

- ○ $10^6 \cdot 2 \cdot 8$ seconds
- ○ $10^6 \cdot 2^8$ seconds
- ○ 2^8 seconds
- ○ $6 \cdot 10 \cdot 2^8$ seconds
- ○ 10^6 seconds

Question 3 ***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.

Q3.1 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?

Q3.2 Mallory learns that Alice and Bob are using the **simplified** signature scheme described above and decides to trick Bob into beliving 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.

Q3.3 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?