

# COL759 Quiz 4

Anish

TOTAL POINTS

**6 / 10**

QUESTION 1

1 True False **6 / 6**

- ✓ **+ 2 pts** (a) *Correct answer and explanation*
  - + 1 pts** (a) correct, no/incorrect explanation
- ✓ **+ 2 pts** (b) *correct answer and explanation*
  - + 1 pts** (b) correct, no/incorrect explanation
- ✓ **+ 2 pts** (c) *correct answer and explanation*
  - + 1 pts** (c) correct, no/incorrect explanation
- + 0 pts** all incorrect

QUESTION 2

2 **0 / 4**

- + 3 pts** Correct scheme
- + 1 pts** correctnes
- ✓ **+ 0 pts** *incorrect / vague*

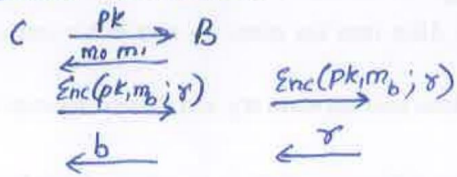
### True False (6 marks)

State whether the following are true or false. Prove a short (one/two line) explanation for your answer.

1. A public key encryption scheme is said to have ‘learnable randomness’ if there exists an efficient algorithm such that, given the public key and a ciphertext, it can learn the randomness used to encrypt. It is possible to have CPA secure public key encryption with learnable randomness.

False

False  
Suppose A learns the randomness of the encryption



A Upon receiving  $ct = \text{Enc}(pk, m_0; r)$   
B forwards it to A to obtain  $r$   
Then it computes  $\text{Enc}(pk, m_0; r)$   
and  $\text{Enc}(pk, m_1; r)$  and checks  
which one matches with  $ct$

2. Let  $N = p \cdot q$  where  $p$  and  $q$  are distinct primes. For any integer  $y \in \mathbb{Z}_N^*$ , there exists a unique integer  $x$  such that  $x^2 \bmod N = y$ .

False

Note: It is assumed that  $x \in \mathbb{Z}_N^*$  [Not needed]

Take  $N=6$   $\mathbb{Z}_N^* = \{1, 5\}$  for  $y=5$ , we have no  $x \in \mathbb{Z}_N^*$  such that  $x^2 \bmod 6 = 5$ . In fact the assumption is not needed. Take  $x \in \mathbb{Z}$

$x^2 \bmod 6 = 5$   
~~Infact for any even~~ Infact the assumption is not needed. Take  $x \in \mathbb{Z}$   
 $x = 6K + r$   $0 \leq r < 6$   $x^2 \bmod 6 = r^2 \bmod 6$  which can be in  $\{1^2 \bmod 6, 2^2 \bmod 6, \dots, 5^2 \bmod 6\} = \{1, 4, 3, 4, 1\}$

3. Suppose there exists an efficient algorithm that, given any  $N$  which is a product of two distinct primes, can find a non-zero element  $x \in \mathbb{Z}_N \setminus \mathbb{Z}_N^*$ . Then this algorithm can be used to break the RSA assumption.

## Tone

Let  $N = pq$ . If A finds  $x \in \mathbb{Z}_N \setminus \mathbb{Z}_N^*$  then  $\gcd(x, N) \neq 1$

Thus  $\gcd(x, N) = p$  or  $q$ . Since  $p$  and  $q$  are the only prime factors of  $N$ . So, using  $A$  we can factor  $N$  and break the RSA assumption: compute  $\gcd(x, N)$

Output  $(\gcd(x, N), \frac{N}{\gcd(x, N)})$

## From Key Exchange to Public Key Encryption (4 marks)

In this exercise, we will study Diffie-Hellman's two-message key-exchange protocol abstractly. A two-message key-exchange protocol with key space  $\{0, 1\}^\lambda$  consists of four algorithms AliceMsg, BobMsg, AliceKey, BobKey with the following properties:

- $(\text{msg}_A, \text{st}_A) \leftarrow \text{AliceMsg}(1^\lambda)$ : Alice uses the security parameter  $\lambda$  to generate her message  $\text{msg}_A$ , and keeps the state  $\text{st}_A$ .
- $(\text{msg}_B, \text{st}_B) \leftarrow \text{BobMsg}(\text{msg}_A)$ : Bob, on receiving  $\text{msg}_A$ , computes his response  $\text{msg}_B$ , and also keeps the state  $\text{st}_B$ .
- $k \leftarrow \text{AliceKey}(\text{st}_A, \text{msg}_B)$ : Alice uses her state  $\text{st}_A$  and Bob's message  $\text{msg}_B$  to compute the key  $k \in \{0, 1\}^\lambda$ .
- $k' \leftarrow \text{BobKey}(\text{st}_B, \text{msg}_A)$ : Bob uses his state  $\text{st}_B$  and Alice's message  $\text{msg}_A$  to compute the key  $k' \in \{0, 1\}^\lambda$ .

For correctness, we require the following: if  $(\text{msg}_A, \text{st}_A) \leftarrow \text{AliceMsg}(1^\lambda)$ ,  $(\text{msg}_B, \text{st}_B) \leftarrow \text{BobMsg}(\text{msg}_A)$ ,  $k \leftarrow \text{AliceKey}(\text{st}_A, \text{msg}_B)$  and  $k' \leftarrow \text{BobKey}(\text{st}_B, \text{msg}_A)$ , then  $k = k'$ .

For security, we require that no p.p.t. adversary can distinguish between the following two distributions:

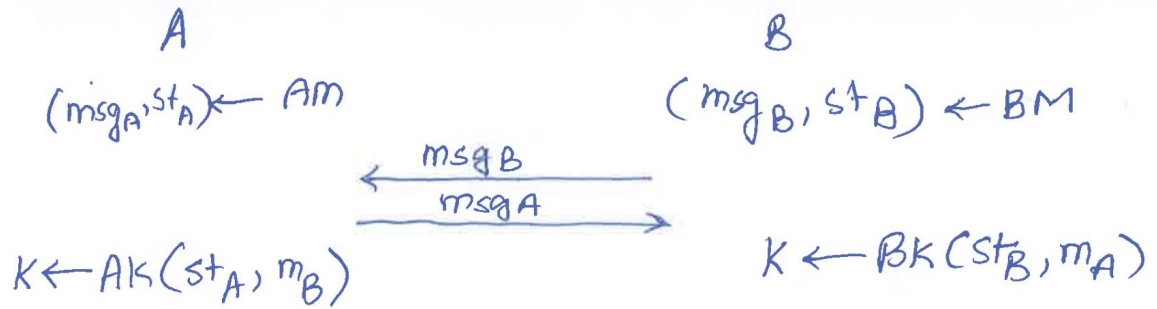
$$\mathcal{D}_0 = \left\{ (k, \text{msg}_A, \text{msg}_B) : \begin{array}{l} (\text{msg}_A, \text{st}_A) \leftarrow \text{AliceMsg}(1^\lambda) \\ (\text{msg}_B, \text{st}_B) \leftarrow \text{BobMsg}(\text{msg}_A) \\ k \leftarrow \text{AliceKey}(\text{st}_A, \text{msg}_B) \end{array} \right\}$$

$$\mathcal{D}_1 = \left\{ (r, \text{msg}_A, \text{msg}_B) : \begin{array}{l} (\text{msg}_A, \text{st}_A) \leftarrow \text{AliceMsg}(1^\lambda) \\ (\text{msg}_B, \text{st}_B) \leftarrow \text{BobMsg}(\text{msg}_A) \\ r \leftarrow \{0, 1\}^\lambda \end{array} \right\}$$

Use the two-round key exchange protocol to design a CPA secure public key encryption scheme with message space  $\{0, 1\}^\lambda$ . You should not use any other cryptographic primitives. Define the three algorithms (Setup, Enc, Dec), and discuss why your scheme is correct. You do not need to prove CPA security.

Setup  $pk = (\text{msg}_A, \text{msg}_B) \quad sk = k$

(Rough work)



$$PK = (g, g^a)$$

$$a = sk$$

$$g^x, g^{x_0 \cdot m}$$

