INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR

Stamp / Signature of the Invigilator

EXAMINATION (Mid Semester)										**SEMESTER (Spring)**			
Roll Number										**Section**		**Name**	
Subject Number	C	S	6	0	0	8	8	**Subject Name**		*Foundations of Cryptography*			
Department / Center of the Student											**Additional sheets**		

Important Instructions and Guidelines for Students

1. You must occupy your seat as per the Examination Schedule/Sitting Plan.
2. Do not keep mobile phones or any similar electronic gadgets with you even in the switched off mode.
3. Loose papers, class notes, books or any such materials must not be in your possession, even if they are irrelevant to the subject you are taking examination.
4. Data book, codes, graph papers, relevant standard tables/charts or any other materials are allowed only when instructed by the paper-setter.
5. Use of instrument box, pencil box and non-programmable calculator is allowed during the examination. However, exchange of these items or any other papers (including question papers) is not permitted.
6. Write on both sides of the answer script and do not tear off any page. **Use last page(s) of the answer script for rough work.** Report to the invigilator if the answer script has torn or distorted page(s).
7. It is your responsibility to ensure that you have signed the Attendance Sheet. Keep your Admit Card/Identity Card on the desk for checking by the invigilator.
8. You may leave the examination hall for wash room or for drinking water for a very short period. Record your absence from the Examination Hall in the register provided. Smoking and the consumption of any kind of beverages are strictly prohibited inside the Examination Hall.
9. Do not leave the Examination Hall without submitting your answer script to the invigilator. **In any case, you are not allowed to take away the answer script with you.** After the completion of the examination, do not leave the seat until the invigilators collect all the answer scripts.
10. During the examination, either inside or outside the Examination Hall, gathering information from any kind of sources or exchanging information with others or any such attempt will be treated as **'unfair means'**. Do not adopt unfair means and do not indulge in unseemly behavior.

Violation of any of the above instructions may lead to severe punishment.

Signature of the Student

To be filled in by the examiner

Question Number	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Total**
Marks Obtained											

Marks obtained (in words)	**Signature of the Examiner**	**Signature of the Scrutineer**

CS60088 Foundations of Cryptography, Spring 2016–2017

Mid-Semester Test

16–February–2017 NC-141/142, NR-321/322, 2:00–4:00pm Maximum marks: 40

[*Write your answers in the question paper itself. Be brief and precise. Answer all questions.*]

1. In a cryptographic application, two types of (pseudo)random bit streams are needed:

(i) a stream $A = a_1a_2a_3\ldots$ in which $\Pr[a_i = 0] = \Pr[a_i = 1] = \frac{1}{2}$ for all i, and
(ii) a stream $B = b_1b_2b_3\ldots$ in which $\Pr[b_i = 0] = \frac{2}{3}$ and $\Pr[b_i = 1] = \frac{1}{3}$ for all i.

(a) You are given a generator G_A for A. Propose an efficient construction that uses G_A to generate B. **(5)**

Solution Take two bits $a_{2i-1}a_{2i}$ from the stream A generated by G_A. If $a_{2i-1}a_{2i} = 00$ or 01, send a zero bit to the output sequence. If $a_{2i-1}a_{2i} = 10$, send a one bit to the output sequence. If $a_{2i-1}a_{2i} = 11$, discard this pair (do not send anything to the output sequence). The probability that a bit pair is not discarded is $\frac{3}{4}$, so a output bit b_j is zero with probability $\frac{2}{4}/\frac{3}{4} = \frac{2}{3}$, and is one with probability $\frac{1}{4}/\frac{3}{4} = \frac{1}{3}$.

(b) You are given a generator G_B for B. Propose an efficient construction that uses G_B to generate A. **(5)**

Solution Take two bits $b_{2i-1}b_{2i}$ from the stream B generated by G_B. If $b_{2i-1}b_{2i} = 01$, send a zero bit to the output sequence. If $b_{2i-1}b_{2i} = 10$, send a one bit to the output sequence. If $b_{2i-1}b_{2i} = 00$ or 11, discard this bit pair (do not send anything to the output sequence). An input pair is not discarded with probability $\frac{2}{3} \times \frac{1}{3} + \frac{1}{3} \times \frac{2}{3} = \frac{4}{9}$. Given that a bit pair is not discarded, the output bit is zero with probability $(\frac{2}{3} \times \frac{1}{3})/\frac{4}{9} = \frac{1}{2}$, and is one with probability $(\frac{1}{3} \times \frac{2}{3})/\frac{4}{9} = \frac{1}{2}$.

2. Let E be a deterministic public-key encryption algorithm, that is, each plaintext $m \in \{0,1\}^l$ has a unique ciphertext $c = E_{pub}(m) \in \{0,1\}^l$. A probabilistic encryption scheme E' is constructed in the following way.

1. Choose $x \in_R \{0,1\}^l$. (Here, the notation $\in_R$ stands for a uniformly random choice.)
2. Compute $u = E_{pub}(x) \in \{0,1\}^l$.
3. Compute $v = x \oplus m$.
4. The pair (u,v) is an encryption $E'_{pub}(m)$ of m.

(a) Describe how a ciphertext (u,v) generated by E' can be decrypted. **(5)**

Solution The recipient uses the private key to compute the session secret $x = D_{prv}(u)$. The message is then recovered as $m = x \oplus v$.

(b) Prove/Disprove: E' is IND-CPA secure. **(5)**

Solution *False.* To see why, let an IND-CPA attacker choose any two plaintext messages m_0, m_1, and get a challenge ciphertext $c^* = (u^*, v^*)$ from the encryption oracle. If m_0 was encrypted by the oracle, then u^* is *the* encryption of $v^* \oplus m_0$ (by E), but not of $v^* \oplus m_1$. Likewise, if m_1 was encrypted by the oracle, then u^* is *the* encryption of $v^* \oplus m_1$, but not of $v^* \oplus m_0$. Since E is deterministic, this can be easily decided by the attacker.

3. Let $n = pq$ be an RSA modulus, and (e,d) a key pair under this modulus. Let l be the bit-length of n. We write $l = 1 + l_1 + l_2$. The message space is $\{0,1\}^{l_1}$. A random l_2-bit string s (called the *salt*) is appended to the message m to get the padded message $M = 0 \,||\, m \,||\, s$ (here, $||$ means concatenation), which is then encrypted as $c \equiv M^e \pmod{n}$. The most significant bit is taken as 0 to ensure that M (treated as an integer in binary) does not exceed the modulus n.

(a) How can you decrypt a ciphertext c obtained by this scheme? **(5)**

Solution The recipient uses the decryption exponent d to compute the padded message $M \equiv c^d \pmod{n}$. The recipient then decomposes $M = b \,||\, m \,||\, s$ with b a bit, $|m| = l_1$, and $|s| = l_2$. If $b = 1$, decryption fails. Otherwise, m is taken as the recovered plaintext message.

(b) Prove that this padded RSA encryption scheme is IND-CCA2 insecure (whatever the length l_2 is). **(5)**

Solution The IND-CCA2 attacker chooses $m_0 = 0^{l_1}$ and $m_1 = 0^{l_1-1}1$. Let the challenge ciphertext be c^*. The attacker supplies the adaptive ciphertext $c \equiv 2^e c^* \pmod{n}$. Since $c \neq c^*$, the decryption oracle returns the corresponding plaintext message μ. If m_0 was encrypted, this message is $\mu = 0^{l_1}$ or $0^{l_1-1}1$. If m_1 was encrypted, we have $\mu = 0^{l_1-2}10$ or $0^{l_1-2}11$. Therefore by looking at the second least significant bit of μ, the attacker determines with certainty which message was encrypted.

(c) Assume that it is infeasible for an attacker to carry out 2^{80} e-th power exponentiations modulo n. To preclude the possibility of exhaustive search, we therefore take $l_2 = 80$. Suppose that the RSA assumption holds. Prove/Disprove: The padded RSA encryption is IND-CPA secure. **(5)**

Solution *False.* The IND-CPA attacker chooses $m_0 = 0^{l_1}$ and $m_1 = 1^{l_1}$. If m_0 is encrypted, then a short message is encrypted, and the meet-in-the-middle test succeeds with non-negligible probability ε. The attacker runs this test on the challenge ciphertext. If the test succeeds, the attacker outputs the bit 0. Otherwise, the attacker outputs 0 or 1 with equal probability ($\frac{1}{2}$). So the attacker succeeds with probability

$$\begin{aligned} & \frac{1}{2} \times \varepsilon && [m_0 \text{ was encrypted, and the meet-in-the-middle attack succeeds}] \\ + & \frac{1}{2} \times (1-\varepsilon) \times \frac{1}{2} && [m_0 \text{ was encrypted, and the meet-in-the-middle attack fails}] \\ + & \frac{1}{2} \times \frac{1}{2} && [m_1 \text{ was encrypted}] \\ = & \frac{1}{2} + \frac{\varepsilon}{4}. \end{aligned}$$

So the advantage of the attacker is non-negligible. Moreover, the attacker has to make at most $2^{40} + 2^{40} = 2^{41}$ RSA encryptions. This is feasible.

There are better algorithms that, given an RSA ciphertext and a significant number of bits of the corresponding plaintext, recovers the complete plaintext.

4. Let E be a public-key encryption algorithm. Alice generates a ciphertext $c = E_{pub}(m)$ using the public key of Bob. In order to send c to Bob, Alice submits c to a network protocol which encodes c to $c_{enc} = ENCODE(c)$ (for example, encoding includes breaking c into segments, and adding headers). Bob uses the reverse decoding function to recover $c = DECODE(c_{enc})$, and decrypts c by his private key. The encoding and decoding functions are publicly known (like TCP formatting), and do not use any keys. The question is whether the networking interface degrades the security of E. Fortunately, the answer is *no*. More specifically, supply a formal reduction argument to prove that if E is IND-CPA secure, then the composition $ENCODE \circ E$ (which maps m to c_{enc}) is also IND-CPA secure. **(5)**

Solution Call this composition E'. We prove that if E' is IND-CPA insecure, then E is also IND-CPA insecure. Let $\mathscr{A}'$ be a PPT IND-CPA adversary for E' with non-negligible advantage ε, and $\mathscr{O}$ an encryption oracle for E. Simon the simulator communicates with $\mathscr{A}'$ and $\mathscr{O}$ as follows.

$$\boxed{\mathscr{A}'} \iff \boxed{\text{Simon}} \iff \boxed{\mathscr{O}}$$

$\mathscr{A}'$ chooses two plaintext messages m_0, m_1 of the same length, and sends those to Simon. Simon forwards the same messages to $\mathscr{O}$. The oracle $\mathscr{O}$ chooses a random bit $b \in_R \{0,1\}$, and sends $c^* = E_{pub}(m_b)$ to Simon. Simon encodes c^*, and sends $c^*_{enc} = ENCODE(c)$ to $\mathscr{A}'$. Eventually, $\mathscr{A}'$ outputs a bit b' to Simon, which Simon too outputs. Now, $\mathscr{A}'$ wins the IND-CPA game for E' with advantage ε. But then, Simon too wins the IND-CPA game for E with the same advantage ε.