INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR

Stamp / Signature of the Invigilator

EXAMINATION (End 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 2015–2016

End-Semester Test

26–April–2016 CSE-107/119/120, 2:00–5:00pm Maximum marks: 60

Roll no: _____ **Name:** _____

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

1. Supply brief (one/two-sentence) answers to the following parts. **(2 × 8)**

(a) What is the computational Diffie–Hellman problem in $\mathbb{Z}_p$?

Solution Given $g, g^a, g^b \pmod p$, compute the element $g^{ab} \pmod p$.

(b) What is the decisional Diffie–Hellman problem in $\mathbb{Z}_p$?

Solution Given $g, g^a, g^b, g^c \pmod n$, decide whether $c \equiv ab \pmod{\mathrm{ord}_p(g)}$.

(c) What is the parity oracle for textbook RSA encryption?

Solution A (hypothetical) polynomial-time algorithm that, given the RSA ciphertext c of a message m, returns the least significant bit of m.

(d) What is the difference between the security notions CCA and CCA2 for encryption schemes?

Solution In CCA, decryption assistance stops after the challenge ciphertext is presented to the attacker. In CCA2, decryption assistance does not stop even after the challenge ciphertext is presented to the attacker.

(e) What is adaptive chosen message attack for a digital signature scheme?

Solution The adversary gets the victim's signatures on messages chosen by the adversary.

(f) Show that RSA signatures are existentially forgeable.

Solution For any $s \in \mathbb{Z}_n$, take $m \equiv s^e \pmod n$. Then, s is a valid RSA signature on m.

(g) What is the difference between a zero-knowledge proof and a zero-knowledge argument?

Solution In a zero-knowledge proof protocol, the prover is assumed to have unbounded computational resources. In a zero-knowledge argument protocol, the prover is assumed to be bounded (poly-time).

(h) Why cannot we take long challenges in the Schnorr identification protocol?

Solution To prevent a dishonest verifier from getting Schnorr signatures of the prover on sensitive messages.

2. Let E be a public-key encryption scheme. Define a public-key encryption scheme E' as

$$E'_{pub}(m) = E_{pub}(m) \,||\, H(m),$$

where H is an unkeyed cryptographic hash function. Here, E and E' use the same public key *pub*.

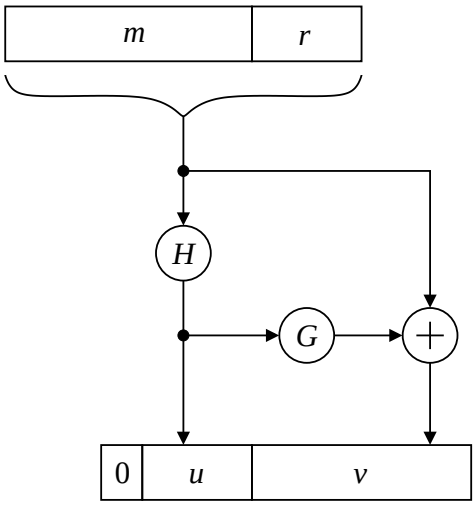
(a) Prove or disprove: If E is IND-CPA secure, then E' is IND-CPA secure. **(4)**

Solution *False.* The appended hash value clearly indicates whether m_0 or m_1 was encrypted. The assumption that $H(m_0) = H(m_1)$ violates the collision-resistance property of H. In any case, the IND-CPA adversary can arrange two messages m_0, m_1 satisfying $H(m_0) \neq H(m_1)$.

(b) Prove or disprove: If E is NM-CPA secure, then E' is NM-CPA secure. **(4)**

Solution *False.* Like Part (a), it is easy to see that E' is not IND-CPA secure. Now, use the fact that NM-CPA security implies IND-CPA security (equivalently, IND-CPA insecurity implies NM-CPA insecurity).

3. Coron's version of RSA-PSS-R padding scheme is shown in the adjacent figure. An RSA modulus n is used. The message m is of bit length k_0, and the random salt r is of bit length k_1. We have $|n| = k_0 + k_1 + k_2 + 1$ for some k_2. Two hash functions $H : \{0,1\}^{k_0+k_1} \to \{0,1\}^{k_2}$ and $G : \{0,1\}^{k_2} \to \{0,1\}^{k_0+k_1}$ are used. The padded message is $y = 0 \,||\, u \,||\, v$. The RSA-PSS-R signature on m is $s \equiv y^d \pmod{n}$, where (e,d) is the RSA key pair of the signer. Here are the padding computations:



$$\begin{aligned} r &\in_U \{0,1\}^{k_1}, \\ \mu &= m \,||\, r, \\ u &= H(\mu), \\ v &= G(u) \oplus \mu \\ y &= 0 \,||\, u \,||\, v. \end{aligned}$$

(a) Explain how the message is recovered and verified from an RSA-PSS-R signature s. **(4)**

Solution The padded message is first recovered as $y \equiv s^e \pmod{n}$. If the msb of y is not zero, failure is reported. Otherwise, y is decomposed to u and v with $|u| = k_2$ and $|v| = k_0 + k_1$. Next, $\mu = G(u) \oplus v$ is computed. If $H(\mu) \neq u$, failure is returned. Otherwise, the first k_0 bits of μ are returned as the recovered message m.

We now prove the security of the scheme against adaptive-chosen-message attacks in the random oracle model. Assume that there exists a PPT adversary Vera that, upon signing assistance, can produce a forged RSA-PSS-R signature on some (new) message. The simulator Ronald is a random oracle, and plays the CMA game with Vera to compute $\eta^d \pmod{n}$ for a random η input to him.

(b) Explain how Ronald simulates a signing query from Vera. **(4)**

Solution Let Vera make a signing query on m. Ronald takes $s \in_U \mathbb{Z}_n$, and computes $y \equiv s^e \pmod{n}$. Ronald repeats until the msb of y is 0, and u does not reside in the G-table of Ronald. Once y (and so u, v) are fixed, Ronald chooses a random k_1-bit salt r such that $m \,||\, r$ does not reside in the H-table of Ronald. Ronald then stores $H(m \,||\, r) = u$ and $G(u) = v \oplus (m \,||\, r)$. Finally, Ronald returns s to Vera. Under these new hash values, s is evidently a valid signature on m.

(c) Explain how G and H oracle queries are answered by Ronald. **(4)**

Solution A query $G(Q)$ is attended in the usual manner. If Q resides in the G-table, the stored value is returned. Otherwise, a uniformly random $k_0 + k_1$-bit string is returned.

For a query $H(Q)$, Ronald first looks at his H-table. If the query was made earlier, the stored value is returned. Otherwise, Ronald repeats computing $x \in_U \mathbb{Z}_n$, $z \equiv x^e \pmod{n}$, and $y \equiv \eta z \pmod{n}$ until the msb of y is 0, and the corresponding u does not reside in the G-table. Ronald sets $H(Q) = u$ and $G(u) = v \oplus Q$, remembers x against Q, and returns u to Vera.

(d) Explain how Ronald achieves his objective of computing $\eta^d \pmod{n}$. **(4)**

Solution At the end, Vera produces a valid signature s on some message m. Notice that m was not signed by Ronald. However, the signature corresponds to a salt value r. It is very improbable for Vera to create the forged signature s without knowing $H(m \,||\, r)$. So with overwhelmingly large probability, Vera has made the query $H(m \,||\, r)$.

Ronald now has $u = H(m \,||\, r)$, $v = G(u) \oplus (m \,||\, r)$, and $y = 0 \,||\, u \,||\, v$. Moreover $y \equiv z\eta \pmod{n}$, so the forged signature is $s \equiv y^d \equiv z^d \eta^d \equiv x\eta^d \pmod{n}$. Since Ronald remembered x in connection with the query $H(m \,||\, r)$, he can now compute $\eta^d \equiv x^{-1} s \pmod{n}$.

4. Alice wants to convince Bob that she can generate valid ElGamal signatures. Let $g \in \mathbb{Z}_p^*$ be an element of large prime order q. Alice's key pair consists of $x \in_U \mathbb{Z}_q$ (the private key) and $y \equiv g^x \pmod{p}$ (the public key). Alice's knowledge of x allows her to generate valid ElGamal signatures. However, to avoid chosen message attacks, she must not use x itself for generating signatures.

Commitment Alice chooses $t \in_U \mathbb{Z}_q^*$, and sends t to Bob.

Challenge Bob chooses a random message $m \in_U \mathbb{Z}_q$, and sends m to Alice.

Response Alice chooses $k \in_U \mathbb{Z}_q$. She then computes $r \equiv g^k \pmod{p}$ and $s \equiv k^{-1}(m - txr) \pmod{q}$. Alice sends (r,s) to Bob.

(a) Explain the verification step by Bob. **(4)**

Solution We have $m \equiv sk + txr \pmod{p}$. So $g^m \equiv (g^k)^s (g^x)^{tr} \pmod{p}$, that is, Bob accepts Alice if and only if $g^m \equiv r^s y^{tr} \pmod{p}$.

(b) Deduce the completeness and soundness-error probabilities of the protocol. **(4)**

Solution Clearly, if Alice knows x, she can compute the correct response (r,s), so $\varepsilon = 1$.

Now, suppose that Alice is a cheating prover. She does not know x. Equivalently, she does not know tx. But then, producing a valid response to a message m chosen by Bob is equivalent to generating an ElGamal signature under the signing key tx. Therefore the soundness-error probability is the same as Alice's probability of forging ElGamal signatures without knowing the signing key. Since the message is chosen by Bob, this is not existential forgery. So under the assumption that ElGamal signatures are secure, the soundness-error probability is negligible.

(c) Prove the zero-knowledge property of the protocol. **(4)**

Solution An equator uses existential forgery. It chooses $t, u, v \in_U \mathbb{Z}_q$ (with $v \neq 0$). It takes $r \equiv g^u y^v \pmod{p}$. Verification requires $g^m \equiv (g^u y^v)^s y^{tr} \pmod{p}$. So the equator can take $vs + tr \equiv 0 \pmod{q}$, that is, $s \equiv -v^{-1}tr \pmod{q}$, and $m \equiv su \equiv -v^{-1}tru \pmod{q}$. The equated transcript is t, m, r, s.

5. A zero-knowledge protocol is called *specially sound* if the use of the same commitment value in two different runs of the protocol discloses to the verifier (or any eavesdropper) the private input of the prover.

(a) Show that the Schnorr identification protocol is specially sound. **(4)**

Solution Let k be the same committal used in two runs of the Schnorr protocol. For two different challenges c_1, c_2, the responses are $r_1 \equiv k - xc_1 \pmod{q}$ and $r_2 \equiv k - xc_2 \pmod{q}$. Elimination of k gives $r_1 + xc_1 \equiv r_2 + xc_2 \pmod{q}$. Therefore if $c_1 \not\equiv c_2 \pmod{q}$, we have $x \equiv (c_1 - c_2)^{-1}(r_2 - r_1) \pmod{q}$.

(b) Is the protocol of Exercise 4 specially sound? **(4)**

Solution No. Here, t is the commitment. Even if it remains the same in two runs, the session secret k is assumed to be different in the two runs. So Bob (or an eavesdropper) sees two ElGamal signatures of Alice under the private key $tx \pmod{q}$. If Bob gains the knowledge of x, he also knows the private key $tx \pmod{q}$. But no such attack on the ElGamal signature scheme is known. Of course, if t remains constant in many runs of the protocol, Bob can potentially mount a chosen-message attack which may reveal tx (and so x) to him. But only two runs are believed to be insufficient. However, if both t and k are repeated in two runs, then x is disclosed to Bob.