دانشکده‌ی علوم ریاضی

مقدمه‌ای بر رمزنگاری — مهلت اصلی: ۳ تیر ۹۸

تمرین شماره ۵

مدرّس: دکتر شهرام خزائی — مهلت نهایی: ۱۰ تیر ۹۸

- Upload your answers on courseware with the name: StudentNumber.pdf
- Upload a PDF file. Image and zip formats are not accepted.
- Similar answers will not be graded.
- NO answers will be accepted via e-mail.
- Deadline time is always at 23:55 and will not be extended.
- You should submit your answers before soft deadline.
- You will lose 5 percent for each day delay before hard deadline.
- You can not submit any time after hard deadline.
- All problem sets include at least 150 points which is the full score.
- Answering questions marked with (*) is mandatory.
- You can gain up to 180 points by answering unmarked questions.
- For any question contact `mr.comp97@gmail.com`

Problem 1*

Let (G, S, V) be a secure signature scheme with message space $\{0,1\}^n$. Generate two sigining/verification key pairs $(pk_0, sk_0)\overset{R}{\leftarrow} G()$ and $(pk_1, sk_1)\overset{R}{\leftarrow} G()$. Which of the following are secure signature schemes? Show an attack or prove security. (30 Points)

- Accept one valid: $S_1((sk_0, sk_1), m) := (S(sk_0, m), S(sk_1, m))$. Verify:

$$V_1((pk_0, pk_1), m, (\sigma_0, \sigma_1)) = accept \quad \Longleftrightarrow$$

$$[V(pk_0, m, \sigma_0) = accept \;\; or \;\; V(pk_1, m, \sigma_1) = accept]$$

- Sign halves: $S_2((sk_0, sk_1), (m_L, m_R)) := (S(sk_0, m_L), S(sk_1, m_R))$. Verify:

$$V_2((pk_0, pk_1), (m_L, m_R), (\sigma_0, \sigma_1)) = accept \quad \Longleftrightarrow$$

$$V(pk_0, m_L, \sigma_0) = V(pk_1, m_R, \sigma_1) = accept$$

- Sign with randomness: $for \;\; m \in \{0,1\}^n \;\; do :$

  $S_3(sk_0, m) := [choose \;\; random \;\; r \;\; from \;\; \{0,1\}^n, \;\; output \;\; (r, S(sk_0, m \oplus r), S(sk_0, r))]$.

  $V_3(pk_0, m, (r, \sigma_0, \sigma_1)) = accept \quad \Longleftrightarrow \quad V(pk_0, m \oplus r, \sigma_0) = V(pk_0, r, \sigma_1) = accept$

Problem 2*

Let $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a CCA-secure public-key encryption scheme defined over $(\mathcal{M}, \mathcal{C})$ where $\mathcal{C} := \{0,1\}^l$. Consider the encryption scheme $\Pi' = (\mathsf{Gen}, \mathsf{Enc}', \mathsf{Dec}')$ defined over $(\mathcal{M}, \mathcal{C}')$ where $\mathcal{C}' := \{0,1\}^{l+1}$ as follows:

$$\mathsf{Enc}'(pk, m) := \mathsf{Enc}(pk, m)||0 \;\; and \;\; \mathsf{Dec}'(sk, c) := \mathsf{Dec}(sk, c[0 \cdots l-1])$$

Show that Π' is not CCA-secure. (15 Points)

Problem 3*

(20 Points)

- Solve the equation $3x + 2 = 7$ in $\mathbb{Z}_{19}$.
- How many elements are there in $\mathbb{Z}^*_{35}$?
- Compute $2^{10001} \mod 11$ without using a calculator.

- Justify why there must exist integers a and b such that $7a + 23b = 1$. Find such a pair (a, b) with the smallest possible $a > 0$ and then determine the inverse of 7 in $\mathbb{Z}_{23}$?
- Compute $2^{245} \mod 35$ without using a calculator.
- What is the order of 2 in $\mathbb{Z}_{35}^*$?
- Solve the equation $x^2 + 4x + 1 = 0$ in $\mathbb{Z}_{23}$.
- Determine all generators and subgroups of $\mathbb{Z}_{13}^*$?
- What is the 11th root of 2 in Z_{19}?
- What is the discete log of 5 base 2 in Z_{13}?

Problem 4*

Consider the following public-key encryption scheme. The public key is $(\mathbb{G}, q, g, h) \leftarrow \mathcal{G}$ and the private key is x, generated exactly as in the El Gamal encryption scheme. In order to encrypt a bit b, the sender does the following:

- If $b = 0$ then choose a random $y \in \mathbb{Z}_q$ and compute $c_1 := g^y$ and $\mathrm{c}_2 := h^y$. The ciphertext is $\langle c_1, c_2 \rangle$.
- If $b = 1$ then choose independent random $y, z \in \mathbb{Z}_q$, compute $c_1 := g^y$ and $c_2 := g^z$, and set the ciphertext equal to $\langle c_1, c_2 \rangle$.

Show that it is possible to decrypt efficiently given knowledge of x. Prove that this encryption scheme is CPA-secure if the decisional Diffie– Hellman problem is hard relative to $\mathcal{G}$. (15 Points)

Problem 5

Suppose Alice and Bob live in a country with 50 states. Alice is currently in state $a \in \{1, \ldots, 50\}$ and Bob is currently in state $b \in \{1, \ldots, 50\}$. They can communicate with one another and Alice wants to test if she is currently in the same state as Bob. If they are in the same state, Alice should learn that fact and otherwise she should learn nothing else about Bob's location. Bob should learn nothing about Alice's location. They agree on the following scheme:

- They fix a group G of prime order p and generator g of G.
- Alice chooses random x and y in $\mathbb{Z}_p$ and sends to Bob $\langle A_0, A_1, A_2 \rangle = \langle g^x, g^y, g^{xy+a} \rangle$.

- Bob choose random r and s in $\mathbb{Z}_p$ and sends back to Alice $\langle B_1, B_2 \rangle = \langle A_1^r g^s, (\frac{A_2}{g^b})^r A_0^s \rangle$

1. What should Alice do now to test if they are in the same state (i.e. to test if $a = b$)? (5 Points)
2. Justify why Bob learns nothing from this protocol. (10 Points)

Problem 6

Recall that an RSA public key consists of an RSA modulus N and an exponent e. One might be tempted to use the same RSA modulus in different public keys. For example, Alice might use $\langle N, 3 \rangle$ as her public key while Bob may use $\langle N, 5 \rangle$ as his public key. Alice's secret key is $d_a = 3^{-1} \mod \phi(N)$ and Bob's secret key is $d_b = 5^{-1} \mod \phi(N)$. In this question we will show that it is insecure for Alice and Bob to use the same modulus N. In particular, we show that either user can use their secret key to factor N. Alice can use the factorization to compute $\phi(N)$ and then compute Bob's secret key.

- As a first step, show that Alice can use her public key $\langle N, 3 \rangle$ and private key d_a to construct an integer multiple of $\phi(N)$. (5 Points)
- Now that Alice has a multiple of $\phi(N)$ let's see how she can factor $N = pq$. Let x be the given muliple of $\phi(N)$. Then for any g in $\mathbb{Z}_N^*$ we have $g^x = 1$ in $\mathbb{Z}_N$. Alice chooses a random g in $\mathbb{Z}_N^*$ and computes the sequence

  $$g^x, g^{\frac{x}{2}}, g^{\frac{x}{4}}, g^{\frac{x}{8}}, \ldots$$

  in $\mathbb{Z}_N$ and stops as soon as she reaches the first element $y = g^{\frac{x}{2^i}}$ such that $y \neq 1$ (if she gets stuck because the exponent becomes odd, she picks a new random g and tries again). It can be shown that with probability $\frac{1}{2}$ this y satisfies

  $$((y = 1 \mod p) \wedge (y = -1 \mod q)) \vee ((y = -1 \mod p) \wedge (y = 1 \mod q))$$

  How can Alice use this y to factor N? (10 Points)

Problem 7

An administrator comes up with the following key management scheme: he generates an RSA modulus N and an element s in $\mathbb{Z}_N^*$. He then gives user number i the secret key $s_i = s^{r_i}$ in $\mathbb{Z}_N$ where r_i is the i'th prime (i.e., 2 is the first prime, 3 is the second, and so on).
Now, the administrator encrypts a file that is accssible to users i, j and t with the key

$k = s^{r_i r_j r_t}$ in $\mathbb{Z}_N$. It is easy to see that each of the three users can compute k. For example, user i computes k as $k = (s_i)^{r_j r_t}$. The administrator hopes that other than users i, j and t, no other user can compute k and access the file.
We want to show that this system is insecure by showing that any two colluding users can combine their secret keys to recover the master secret s and then access all files on the system. Suppose users 1 and 2 collude. Show how they can compute s from their secret keys s_1 and s_2. (15 Points)

Problem 8

Let G be a finite cyclic group of order n and let $pk = \langle g, h = g^a \rangle$ and $sk = \langle g, a \rangle$ be an ElGamal public/secret key pair in G. Suppose we want to distribute the secret key to two parties so that both parties are needed to decrypt. Moreover, during decryption the secret key is never re-constructed in a single location. A simple way to do so it to choose random numbers a_1, a_2 in $\mathbb{Z}_n$ such that $a_1 + a_2 = a$. One party is given a_1 and the other party is given a_2. Now, to decrypt an ElGamal ciphertext $\langle u, c \rangle$ we send u to both parties. What do the two parties return and how do we use these values to decrypt? (10 Points)

Problem 9

Say three users have RSA public keys $\langle N_1, 3 \rangle$, $\langle N_2, 3 \rangle$, and $\langle N_3, 3 \rangle$ (i.e.,they all use $e = 3$), with $N_1 < N_2 < N_3$. Consider the following method for sending the same message $m \in \{0,1\}^l$ to reach of these parties: choose a uniform $r \leftarrow \mathbb{Z}^*_{N_1}$, and send to everyone the same ciphertex:

$$\langle [r^3 \; mod \; N_1], [r^3 \; mod \; N_2], [r^3 \; mod \; N_3], H(r) \oplus m \rangle$$

where $H : \mathbb{Z}^*_{N_1} \rightarrow \{0,1\}^l$. Assume $l \gg n$.
Show that this in not CPA-secure, and an adversary can recover m from the ciphertext even when H is modeled as a random oracle. (20 Points)

Hint: see section 11.5.1 from katz and lindell.

Problem 10

Consider the following protocol for two parties A and B to flip a fair coin (more complicated versions of this might be used for Internet gambling): (1) a trusted party T publishes her public key pk; (2) then A chooses a uniform bit b_A , encrypts it using pk, and announces the ciphertext c_A to B and T ; (3) next, B acts symmetrically and announces a ciphertext $c_B \neq c_A$; (4) T decrypts both c_A and c_B , and the parties XOR the results to obtain the value of the coin.

- Argue that even if A is dishonest (but B is honest), the final value of the coin is uniformly distributed. (5 Points)
- Assume the parties use El Gamal encryption (where the bit b is encoded as the group element g^b before being encrypted—note that efficient decrypt is still possible). Show how a dishonest B can bias the coin to any value he likes. (7 Points)
- Suggest what type of encryption scheme would be appropriate to use here. Can you define an appropriate notion of security and prove that your suggestion achieves this definition? (8 Points)

Problem 11

Consider the toy key exchange protocol using an online trusted 3rd party (TTP) discussed in the class. Suppose Alice, Bob, and Carol are three users of this system (among many others) and each have a secret key with the TTP denoted k_a, k_b, k_c respectively. They wish to generate a group session key k_{ABC} that will be known to Alice, Bob, and Carol but unknown to an eavesdropper. How would you modify the protocol in the lecture to accomodate a group key exchange of this type? (note that all these protocols are insecure against active attacks) (5 Points)

1. Alice contacts the TTP. TTP generates a random k_{ABC} and sends to Alice $\mathsf{Enc}_{k_a}(k_{ABC})$, $\mathsf{ticket}_1 \leftarrow \mathsf{Enc}_{k_c}(\mathsf{Enc}_{k_b}(k_{ABC}))$ and $\mathsf{ticket}_2 \leftarrow \mathsf{Enc}_{k_b}(\mathsf{Enc}_{k_c}(k_{ABC}))$.
   Alice sends k_{ABC} to Bob and k_{ABC} to Carol.
2. Alice contacts the TTP. TTP generates a random k_{AB} and a random k_{AC}. It sends to Alice $\mathsf{Enc}_{k_a}(k_{AB})$, $\mathsf{ticket}_1 \leftarrow \mathsf{Enc}_{k_b}(k_{AB})$ and $\mathsf{ticket}_2 \leftarrow \mathsf{Enc}_{k_c}(k_{AB})$.
   Alice sends ticket_1 to Bob and ticket_2 to Carol.
3. Alice contacts the TTP. TTP generates random k_{ABC} and sends to Alice $\mathsf{Enc}_{k_a}(k_{ABC})$, $\mathsf{ticket}_1 \leftarrow \mathsf{Enc}_{k_b}(k_{ABC})$ and $\mathsf{ticket}_2 \leftarrow \mathsf{Enc}_{k_c}(k_{ABC})$. Alice sends ticket_1 to Bob and ticket_2 to Carol.
4. Alice contacts the TTP. TTP generates a random k_{ABC} and sends to Alice $\mathsf{Enc}_{k_a}(k_{ABC})$, $\mathsf{ticket}_1 \leftarrow k_{ABC}$, $\mathsf{ticket}_2 \leftarrow k_{ABC}$. Alice sends ticket_1 to Bob and ticket_2 to Carol.