مقدمه‌ای بر رمزنگاری

پاسخنامه تمرین شماره ۶

مدرّس: دکتر شهرام خزائی

- This problem sets include 75 points.
- For any question contact Sara Sarfaraz via `sarassm60@gmail.com`.

Problem 1

(10 points) Consider the following key-exchange protocol:
(a) Alice chooses a random key k and a random string r both of length n, and sends $s = k \oplus r$ to Bob.
(b) Bob chooses a random string t of length n and sends $u = s \oplus t$ to Alice.
(c) Alice computes $w = u \oplus r$ and sends w to Bob.
(d) Alice outputs k and Bob computes $w \oplus t$.
Show that Alice and Bob output the same key. Analyze the security of the scheme (i.e., either prove its security or show a concrete break).

Solution The statement below proves that Alice and Bob output the same key k:
$w \oplus t = u \oplus r \oplus t = s \oplus t \oplus r \oplus t = s \oplus r = k \oplus r \oplus r = k$
Consider the key-exchange experiment:
1. Two parties holding 1^n execute protocol. This results in a transcript $trans$ containing all the messages sent by the parties, and a key k output by each of the parties.
2. A uniform bit $b \in \{0,1\}$ is chosen. If $b = 0$ set $\hat{k} := k$, and if $b = 1$ then choose uniform $\hat{k} \in \{0,1\}^n$. $\mathcal{A}$ is given $trans$ and $\hat{k}$, and outputs a bit b'. The output of the experiment is defined to be 1 if $b' = b$, and 0 otherwise. (In case $KE^{eav}_{\mathcal{A},\Pi}(n) = 1$, we say that $\mathcal{A}$ succeeds.) The key exchange protocol Π is called secure if for every PPT adversary $\mathcal{A}$ there exists a negligible function negl such that
$\Pr[b' = b] \leq \dfrac{1}{2} + \mathsf{negl}(n)$. We want to prove that the above protocol is not secure.
$s \oplus u \oplus w = (k \oplus r) \oplus (k \oplus r \oplus t) \oplus (k \oplus r \oplus t \oplus r) = k$

Consider the adversary $\mathcal{A}$ that works as follows: $\mathcal{A}$ computes $k' = s \oplus u \oplus w$. Then outputs $b_1 = 0$ if $k_1 = k'$, and $b_1 = 1$ otherwise. $\mathcal{A}$ wins the game if $b = 0$ and when $b = 1$ the uniformly random key k_1 equals the real key k with probability $\frac{1}{2^n}$. Since $\Pr[k_1 = k|b = 1] = \frac{1}{2^n}$ we compute:

$$\Pr[b_1 = b] = 1 - \Pr[k_1 = k|b = 1]. \Pr(b = 1) = 1 - \frac{1}{2^{n+1}} \geq \mathsf{negl}(n) + 0.5$$

Problem 2

(20 Points) Prove that hardness of the CDH problem relative to $\mathcal{G}$ implies hardness of the discrete-logarithm problem relative to $\mathcal{G}$, and that hardness of the DDH problem relative to $\mathcal{G}$ implies hardness of the CDH problem relative to $\mathcal{G}$.

Solution Let $(G, q, g) \leftarrow G(1^n)$, where G is a cyclic group of order q with bit-size $||q|| = O(n)$ and g a generator of G. To prove that hardness of the CDH implies hardness of the discrete-logarithm problem, we show that any algorithm that solves the discrete-logarithm can be used to solve CDH. Let $\mathcal{A}$ be an arbitrary PPT algorithm for the discrete-logarithm problem with respect to $\mathcal{G}$, i.e., on input (G, q, g, g^x) it outputs $x' \in \mathbb{Z}_q$ and wins the game if $x' = x$. We construct an algorithm $\mathcal{A}'$ for CDH as follows: Given a CDH instance (G, q, g, g^x, g^y), $\mathcal{A}'$ queries $\mathcal{A}$ on (G, q, g, g^x) and receives $x' \in \mathbb{Z}_q$. Then $\mathcal{A}'$ computes$(g^y)^x$. Clearly, $\mathcal{A}'$ succeeds if and only if $\mathcal{A}$ succeeds: $(g^y)^{x'} = DH_g(g^x, g^y) \iff x' = x$. Hardness of CDH relative to $\mathcal{G}$ now implies that the success probability of every PPT algorithm – in particular that of $\mathcal{A}'$– is bounded by some negligible function $\mathsf{negl}(n)$. Thus, we get $\Pr[DLog_{\mathcal{A},\mathcal{G}}(n) = 1] = \Pr[\mathcal{A}'(G, q, g, g^x, g^y) = g^{xy}] \leq \mathsf{negl}(n)$. To prove that CDH is harder than the DDH problem, let $\mathcal{A}$ be an arbitrary PPT algorithm for CDH with respect to $\mathcal{G}$, i.e., on input (G, q, g, g^x, g^y) it outputs $h \in G$ and wins the game if $h = DH_g(g^x, g^y) = g^{xy}$. We construct an algorithm $\mathcal{A}'$ for DDH as follows: Given access to $\mathcal{A}$ and a DDH instance (G, q, g, g^x, g^y, h'), where either $h' = g^{xy}$ or $h' = g^z$ for a $z \in \mathbb{Z}_q$ chosen uniformly at random, the algorithm $\mathcal{A}'$ queries $\mathcal{A}$ on (G, q, g, g^x, g^y) and receives h. $\mathcal{A}'$ outputs 1 if $h' = h$ and 0 else. Thus,

$$\Pr[\mathcal{A}'(G, q, g, g^x, g^y, g^{xy}) = 1] = \Pr[\mathcal{A}(G, q, g, g^x, g^y) = g^{xy}]$$

On the other hand,
$\Pr[\mathcal{A}'(G, q, g, g^x, g^y, g^z) = 1] = \frac{1}{q}$.
Assuming that DDH is hard with respect to $\mathcal{G}$, we get
$|\Pr[\mathcal{A}'(G, q, g, g^x, g^y, g^z) = 1] - Pr[\mathcal{A}'(G, q, g, g^x, g^y, g^{xy}) = 1]| \leq \mathsf{negl}(n)$. This implies
$\Pr[\mathcal{A}(G, q, g, g^x, g^y) = g^{xy}] \leq \mathsf{negl}(n) + \frac{1}{q}$,
which is negligible since $||q|| = n$. This proves hardness of CDH.

Problem 3

(25 points) Consider the following variant of El Gamal encryption. Let $p = 2q + 1$, let G be the group of squares modulo p(so G is a subgroup of $\mathbb{Z}_p^*$ of order q), and let g be a generator of G. The private key is (G, q, g, x) and the public key is (G, q, g, h), where $h = g^x$ and $x \in \mathbb{Z}_q$ is chosen uniformly. To encrypt a message $m \in \mathbb{Z}_q$, choose a uniform $r \in \mathbb{Z}_q$, compute $c_1 = g^r$ mod p and $c_2 = h^r + m \bmod p$, and let the ciphertext be (c_1, c_2). Is this scheme CPA-secure? Prove your answer.

Solution This scheme is not secure. Consider an adversary $\mathcal{A}$ who chooses two random plaintexts $m_0, m_1 \in \mathbb{Z}_q$ and receives cipher text (c_1, c_2) from the challenger which is the ciphertext corresponding to m_b for $b \in \{0, 1\}$. We know that c_2 is not necessarily in G as it equals to $h^y + m \bmod p$ and addition is not the action of G but
$c_2 - m_b \bmod p = h^y$, hence we must have $(c_2 - m_b \bmod p) \in G$.
We know that G includes half of the elements of $\mathbb{Z}_p^*$, so because m_{1-b} is random we have:

$$\Pr[(c_2 - m_{1-b} \bmod p) \in G] = \frac{1}{2}$$

so the algorithm $\mathcal{A}$ does the following:
1. it first checks if $(c_2 - m_1 \bmod p) \in G$ and $(c_2 - m_0 \bmod p) \in G$. 2. if both of them are True, then $\mathcal{A}$ outputs a random bit. Otherwise, if $(c_2 - m_0 \bmod p) \in G$ it will output 0 and if $(c_2 - m_1 \bmod p) \in G$ it will output 1. The probability of $\mathcal{A}$ winning is :

$$\mathsf{Adv}(\mathcal{A}) \geq \frac{1}{2} . \frac{1}{2} + \frac{1}{2} . 1 = \frac{3}{4}$$

So the advantage of $\mathcal{A}$ is non-negligible and the scheme is not secure.

Problem 4

(20 points) Consider the following public-key encryption scheme. The public key is (G, q, g, h) 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 uniform $y \in \mathbb{Z}_q$ and compute $c_1 = g^y$ and $c_2 = h^y$. The ciphertext is (c_1, c_2).
- If $b = 1$ then choose independent uniform $y, z \in \mathbb{Z}_q$, compute $c_1 = g^y$ and $c_2 = g^z$, and set the ciphertext equal to (c_1, c_2).

(a) Show that with high probability we can decrypt the ciphertext efficiently given knowledge of x. Specifically, show how to decrypt a bit that is encrypted correctly.
(b) Prove that this encryption scheme is CPA-secure if the decision Diffie-Hellman

problem is hard relative to G.

Solution A ciphertext (c_1, c_2) can be decrypted as follows: Compute c_1^x. If $c_2 = c_1^x$, then output 0, otherwise output 1. Decryption succeeds with all but negligible probability since for all x, r it holds $\Pr[g^z = h^y] = \Pr[z = xy] = \frac{1}{q}$.
We can find the probability of decrypting the ciphertext correctly:

$$\Pr[Dec(c_1, c_2) = 0|b = 0)] = \Pr[c_1^x = c_2|b = 0] = 1$$

$$\Pr[Dec(c_1, c_2) = 1|b = 1] = \Pr[c_1^x \neq c_2|b = 1]$$
$$= 1 - \Pr[c_1^x = c_2|b = 1] = 1 - \Pr[g^{xy} = g^z] = 1 - \frac{1}{q}$$
$$(\frac{1}{q} \leq negl(n))$$

We now prove CPA-security of the above scheme Π under the DDH assumption. Let $\mathcal{A}$ be an adversary against the CPA-security of the scheme. We construct an adversary A' for DDH which uses $\mathcal{A}$ as a black-box. First, $\mathcal{A}'$ receives a DDH instance $(G, q, g, g^x, g^{x'}, h)$ where either $h = g^{xx'}$ (if $b = 0$) or $h = g^z$ for $z \leftarrow Z_q$ uniformly random (if $b = 1$). $\mathcal{A}'$ sends the public key $pk := (G, q, g, g^x)$ to $\mathcal{A}$. W.l.o.g., we assume that $\mathcal{A}$ outputs the two messages $m_0 = 0$ and $m_1 = 1$ (note, the message space is $\{0, 1\}$). Then $\mathcal{A}'$ sends the challenge ciphertext $c^* := (g^{x'}, h)$ to $\mathcal{A}$. If $b = 0$, then c^* looks like a proper encryption of m_0, if $b = 1$, then c^* is an encryption of m_1. Thus, upon receiving $\mathcal{A}$'s guess b', $\mathcal{A}'$ outputs b'. Assuming DDH is hard relative to $\mathcal{G}$, we get

$$\mathsf{negl}(n) \geq |\Pr[\mathcal{A}'(G, q, g, g^x, g^{x'}, g^{xx'}) = 1] - \Pr[\mathcal{A}'(G, q, g, g^x, g^{x'}, g^z) = 1]|$$
$$= |1 - \Pr[\mathcal{A}'(G, q, g, g^x, g^{x'}, g^{xx'}) = 0] - \Pr[\mathcal{A}'(G, q, g, g^x, g^{x'}, g^z) = 1]| =$$
$$|1 - \Pr[PubK_{A,\Pi}^{cpa}(n) = 1|b = 0] - \Pr[PubK_{A,\Pi}^{cpa}(n) = 1|b = 1]| = |1 - 2\Pr[PubK_{A,\Pi}^{cpa}(n) = 1]|$$

for a negligible function negl. This implies CPA-security of the scheme Π:

$$\Pr[PubK_{A,\Pi}^{cpa}(n) = 1] \leq \frac{1}{2} + \mathsf{negl}(n)$$