Fundamentals of Cryptography: Problem Set 7

Due Wednesday Nov 26, 3PM

Collaboration is permitted (and encouraged); however, you must write up your own solutions and acknowledge your collaborators.

If a problem has **0pt**, it will not be graded.

Problem 0 For more information about pairing-based IBE, you can read the paper "Identity-Based Encryption from the Weil Pairing".

If you are interested in elliptic curves, you are encouraged to read Section 15 of "A Graduate Course in Applied Cryptography" by Boneh & Shoup.

For LWE-based fully homomorphic encryption, you can check the MIT cryptography course 6.875 website `mit6875.org`.

Problem 1 (6pt) Leftover Hash Lemma A k-source X is a random variable (taking values in $\{0,1\}^n$) such that, for all $x \in \{0,1\}^n$, $\Pr[X = x] \le 2^{-k}$.

A family $\mathcal{H}$ of functions $h : \{0,1\}^n \to \{0,1\}^\ell$ is universal if for distinct $x, y \in \{0,1\}^n$,

$$\Pr[h(x) = h(y)] = \frac{1}{2^\ell}$$

where the probability is taken over a uniform selection of h from $\mathcal{H}$.

If the family $\mathcal{H}$ of functions $h : \{0,1\}^n \to \{0,1\}^\ell$ is universal, $\ell = k - 2\log(1/\epsilon) - O(1)$, and X is a k-source random variable, then

$$\Delta((H, H(X)), (H, U)) = \frac{1}{2} \sum_{h \in \mathcal{H}, s \in \{0,1\}^\ell} |\Pr[(H, H(X)) = (h, s)] - \Pr[(H, U) = (h, s)]| \le \epsilon/2$$

Here $\Delta(X, Y)$ is the statistical distance between X, Y, and let U be an uniformly random ℓ-bit string.

Part A. Given a random variable X, define the collision probability:

$$\mathrm{Col}(X) = \Pr[X = X'] = \sum_x \Pr[X = x]^2$$

where X' are drawn independently from the same distribution as X. Prove

$$\mathrm{Col}(H, H(X)) \le |\mathcal{H}|^{-1}(2^{-k} + 2^{-\ell})$$

Part B. We now turn to analyze the squared l_2 distance

$$\|(H, H(X)) - (H, U)\|_2^2 = \sum_{h \in \mathcal{H}, s \in \{0,1\}^\ell} \Big(\Pr[(H, H(X)) = (h, s)] - \Pr[(H, U) = (h, s)]\Big)^2$$

Prove the l_2 distance is less than $\frac{\epsilon^2}{|\mathcal{H}|2^\ell}$.

Part C. Using the result in Part B to prove

$$\Delta((H, H(X)), (H, U)) \le \epsilon/2.$$

Problem 2 (6pt) LWE-based Encryption The decisional Learning with Errors (LWE) assumption says that, for any prime $p \le 2^n$, for any $\ell = \text{poly}(n)$, (here n is the security parameter)

$$(\mathbf{A}, \mathbf{s}^T\mathbf{A} + \mathbf{e}^T) \approx_c (\mathbf{A}, \mathbf{b}^T)$$

where $\mathbf{A} \leftarrow \mathbb{Z}_p^{n\times\ell}$, $\mathbf{s} \leftarrow \mathbb{Z}_p^n$, $\mathbf{e} \leftarrow \{-B(n), \dots, B(n)\}^\ell$, $\mathbf{b} \leftarrow \mathbb{Z}_p^\ell$.

Here B should be viewed as a parameter of the assumption. The smaller B is, the stronger the assumption is.

Consider the following public-key encryption scheme.

$\mathsf{Gen}(1^n)$ chooses a prime $p \le 2^n$, samples $\mathbf{A} \leftarrow \mathbb{Z}_p^{n\times m}$, $\mathbf{s} \leftarrow \mathbb{Z}_p^n$, $\mathbf{e} \leftarrow \{-B, \dots, B\}^m$, computes $\mathbf{b}^T = \mathbf{s}^T\mathbf{A} + \mathbf{e}^T$, outputs

$$pk = (\mathbf{A}, \mathbf{b}), \qquad sk = \mathbf{s}.$$

$\mathsf{Enc}(pk, x)$, for $x \in \{0,1\}$, samples $\mathbf{r} \leftarrow \{0,1\}^m$ and outputs the ciphertext

$$ct = (\mathbf{A}\mathbf{r}, \mathbf{b}^T\mathbf{r} + \lfloor \frac{p}{2} \rfloor \cdot x).$$

Since the scheme only encrypts 1 bit, CPA-security means $(pk, \mathsf{Enc}(pk, 0))$ and $(pk, \mathsf{Enc}(pk, 1))$ are computationally indistinguishable. This is proved by a hybrid argument.

Hybrid 0 (Real world) The distinguisher receives $(pk, ct = \mathsf{Enc}(pk, x))$, where
$pk = (\mathbf{A}, \mathbf{b}) = (\mathbf{A}, \mathbf{s}^T\mathbf{A} + \mathbf{e}^T)$ as sampled in $\mathsf{Gen}(1^n)$
$ct = \mathbf{A}\mathbf{r}, \mathbf{b}^T\mathbf{r} + \lfloor \frac{p}{2} \rfloor \cdot x$ for random $\mathbf{r} \leftarrow \{0,1\}^m$.

Hybrid 1 The distinguisher receives (pk, ct), where
$pk = (\mathbf{A}, \mathbf{b})$ for random matrix $\mathbf{A} \leftarrow \mathbb{Z}_p^{n\times m}$ and random vector $\mathbf{b} \leftarrow \mathbb{Z}_p^m$.
$ct = \mathbf{A}\mathbf{r}, \mathbf{b}^T\mathbf{r} + \lfloor \frac{p}{2} \rfloor \cdot x$ for random $\mathbf{r} \leftarrow \{0,1\}^m$.

Hybrid 2 The distinguisher receives (pk, ct), where
$pk = (\mathbf{A}, \mathbf{b})$ for random matrix $\mathbf{A} \leftarrow \mathbb{Z}_p^{n\times m}$ and random vector $\mathbf{b} \leftarrow \mathbb{Z}_p^m$.
$ct = \mathbf{a}, v + \lfloor \frac{p}{2} \rfloor \cdot x$ for random $(\mathbf{a}, v) \leftarrow \mathbb{Z}_p^n \times \mathbb{Z}_p$.

Apparently, in hybrid 2, the distinguisher guesses x correctly with probability 1/2.

Why are hybrid 0 and hybrid 1 computationally indistinguishable?

Why are hybrid 1 and hybrid 2 statistically indistinguishable? What condition should n, m, p satisfies? (We may need the Leftover hash Lemma.)

Problem 3 (8pt) Paillier Encryption Let n be the security parameter. Sample two n-bit safe primes $p = 2p' + 1$, $q = 2q' + 1$. Let $N = pq$. We focus on the multiplicative group $\mathbb{Z}_{N^2}^*$, and its subgroup $\mathbb{QR}_{N^2} := \{x^2 | x \in \mathbb{Z}_{N^2}^*\}$. Apparently, $\mathbb{QR}_{N^2} \cong \mathbb{QR}_{p^2} \times \mathbb{QR}_{q^2}$, here $\cong$ is the notion of group isomorphism. The size of $\mathbb{QR}_{p^2}$ is

$$|\mathbb{QR}_{p^2}| = \frac{1}{2}|\mathbb{Z}_{p^2}^*| = \frac{1}{2}\varphi(p^2) = \frac{1}{2}p(p-1) = p \cdot p',$$

which is the product of two distinct primes. Thus $\mathbb{QR}_{p^2}$ must be isomorphic to $\mathbb{Z}_p \times \mathbb{Z}_{p'}$. Similarly, $\mathbb{QR}_{q^2}$ is isomorphic to $\mathbb{Z}_q \times \mathbb{Z}_{q'}$.

$$\mathbb{QR}_{N^2} \cong \mathbb{QR}_{p^2} \times \mathbb{QR}_{q^2} \cong \mathbb{Z}_p \times \mathbb{Z}_{p'} \times \mathbb{Z}_q \times \mathbb{Z}_{q'} \cong \mathbb{Z}_N \times \mathbb{Z}_{p'q'}.$$

Therefore, $\mathbb{QR}_{N^2}$ can be decomposed into two groups $\mathbb{G}_N$ and $\mathbb{H}_N$. Group $\mathbb{G}_N$ is the only size-N subgroup of $\mathbb{QR}_{N^2}$. Group $\mathbb{H}_N$ is the only size-$(p'q')$ subgroup of $\mathbb{QR}_{N^2}$. Moreover, for every $x \in \mathbb{QR}_{N^2}$, there exists unique $(g, h) \in \mathbb{G}_N \times \mathbb{H}_N$ such that $x = gh$.

Part A. $\mathbb{G}_N$ is called the "easy" subgroup of $\mathbb{QR}_{N^2}$. Show that $1 + N$ is a generator of $\mathbb{G}_N$.

Part B. Show that the discrete log problem is easy in $\mathbb{G}_N$. In particular, given N, $g \in \mathbb{G}_N$ and g^a, show how compute a' in poly(n) time such that $g^{a'} = g^a$.

Part C. $\mathbb{H}_N$ is called the "hard" subgroup of $\mathbb{QR}_{N^2}$. How to sample a random element in $\mathbb{H}_N$ when N is given but p, q are hidden?

Part D. By the Decisional Composite Residuosity (DCR) assumption, a random element in $\mathbb{H}_N$ is indistinguishable from a random element in $\mathbb{QR}_{N^2}$. That is,

$$(N, h) \approx_{\mathrm{c}} (N, x),$$

where N is sampled as mentioned, $h \leftarrow \mathbb{H}_N$ and $x \leftarrow \mathbb{QR}_{N^2}$.

Show that, under the DCR assumption, the following public-key encryption scheme is CPA-secure.

$\mathsf{Gen}(1^n)$ samples safe primes $p = 2p' + 1, q = 2q' + 1$, lets $N = pq$. Output $pk = N$, $sk = p'q'$.

$\mathsf{Enc}(pk, m)$, for $m \in \mathbb{Z}_N$, samples $h \leftarrow \mathbb{H}_N$ and outputs the ciphertext $c = h \cdot (1 + N)^m$.

$\mathsf{Dec}(sk, c) =$ _____fill the blank_____

Problem 4 (4pt) Homomorphic Encryption Most public-key encryption schemes we explored so far are inherently not CCA-secure because they support conflicting features: rerandomization and/or homomorphic evaluation.

A public-key encryption scheme is *rerandomizable*, if given the public key pk and a ciphertext c encrypting x (i.e., there exists a random tape so that c is the output of $\mathsf{Enc}(pk, x)$), there is an efficient algorithm Rerand who samples another ciphertext c' that also encrypts x. It should satisfy one of the following rerandomization requirement depending on the level of security.

Indistinguishable. If two ciphertexts c_0, c_1 both encrypts the same message, and c' is the rerandomization of c_b, no efficient distinguisher can guess b with significant between than 50-50 chance given c', sk;

Statistical indistinguishable. The sample as indistinguishable, except the distinguisher is computationally unbounded.

Statistical. The distribution of c' is close to that of a fresh encryption of x.

Perfect. The distribution of c' is the same as a fresh encryption of x.

A public-key encryption scheme is *add-homomorphic*, if the message space is an Abelian group $\mathcal{G}$, and given the public key and two ciphertexts c_1, c_2 encrypting $x, y \in \mathcal{G}$ respectively, there is an efficient algorithm Eval to compute another ciphertext c which encrypts $x+y$. Note that, since the algorithm Eval does not know the secret key, it has to compute $x+y$ "blindly".

Goldwasser-Micali encryption is an example of rerandomizable and add-homomorphic encryption where the Abelian group is $\mathbb{F}_2 = \{0,1\}$. Recall that in Goldwasser-Micali, the public key is $pk = (N, a)$ where $N = pq$ and a is a quadratic non-residue modulo either p or q. To rerandomize a ciphertext $c = r^2 a^x = \mathsf{Enc}(pk, x)$, sample random $s \in \mathbb{Z}_N^*$ and output the rerandomization $cs^2 = (rs)^2 a^x$. To homomorphically add two ciphertexts $c_1 = r^2 a^x, c_2 = s^2 a^y$, simply output $c_1 c_2 = (rs)^2 a^{x+y}$.

Show how the following three encryption schemes are rerandomizable and add-homomorphic. Specify the rerandomization algorithm, the Abelian group, and the homomorphic evaluation algorithm. In the first two parts, the rerandomization should be perfect. You do not need to prove the correctness of your algorithms.

Part A DDH-based Encryption.

Part B Paillier Encryption (Problem 3).

Part C LWE-based Encryption (Problem 2).

The Abelian group is $\mathbb{Z}_q$ where $q \ll p$, thus you will need to slightly modifies the encryption algorithm.

Note that for LWE-based encryption, the distribution of the resulting ciphertext (from the rerandomization algorithm or the homomorphic evaluation algorithm) is different from that of a fresh encryption.

Remark: It was shown by Bogdanov and Lee in "Limits of provable security for homomorphic encryption." that add-homomorphism implies rerandomization.

Problem 5 (8pt) Circular Security This problem considers circular security, which has been mentioned in the class to bootstrap somewhat homomorphic encryption schemes to fully homomorphic encryption schemes.

Definition 1 (Circularly secure PKE)**.** A public key encryption scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is said to be circularly secure if any p.p.t. algorithm $\mathcal{A}$ wins the following game (interacting with a challenger) with probability at most $\frac{1}{2} + \text{negl}(\lambda)$:

i. The challenger samples $(pk, sk) \leftarrow \mathsf{Gen}(1^\lambda)$ and sends (pk, c^*) to $\mathcal{A}$, where $c^* \leftarrow \mathsf{Enc}(pk, sk)$ is a ciphertext of the secret key.

ii. $\mathcal{A}$ sends two messages (m_0, m_1) to the challenger.

iii. The challenger samples $b \leftarrow \{0,1\}$ and sends $c \leftarrow \mathsf{Enc}(pk, m_b)$ to $\mathcal{A}$.

iv. $\mathcal{A}$ outputs a bit b'. We say that $\mathcal{A}$ wins if $b' = b$.

Part A It turns out that not every CPA secure public key encryption scheme is also circularly secure. Construct a public-key encryption scheme which is CPA secure but not circularly secure, relying only on the existence of public-key encryption schemes. Prove that your scheme is CPA secure but not circularly secure.

Part B In this part, you will show that a variant of the LWE-based secret key encryption we saw in class does satisfy circular security (under an LWE-like assumption). In particular, we consider a variant of the LWE problem where the secret $\mathbf{s}$ is a uniformly random binary string.

Definition 2 (Binary-secret LWE Assumption)**.** The assumption is parameterized by a modulus q and an error distribution χ (q and χ may depend on the LWE security parameter n), stating that for any polynomial $m(n)$, the following two distributions are computationally indistinguishable:

$$\{(\mathbf{A}, \mathbf{s}^T\mathbf{A}+\mathbf{e}^T) : \mathbf{s} \xleftarrow{\$} \{0,1\}^n, \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m}, \mathbf{e} \leftarrow \chi^m\} \approx_c \{(\mathbf{A}, \mathbf{b}^T) : \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m}, b \xleftarrow{\$} \mathbb{Z}_q^m\}.$$

Under the new assumption, prove that the encryption scheme defined by

$$\mathsf{Enc}(\mathbf{s} \in \{0,1\}^n, \mathbf{m} \in \{0,1\}^m; \mathbf{R} \xleftarrow{\$} \mathbb{Z}_q^{n\times m}, \mathbf{e} \xleftarrow{\$} \chi^m) := \left(\mathbf{R}, \mathbf{s}^T\mathbf{R} + \mathbf{e}^T + \left\lfloor \frac{q}{2} \right\rfloor \mathbf{m}^T\right)$$

is a circularly secure secret key encryption scheme.

Hint. Show that it is possible to generate an encryption of s given only an encryption of 0.

Problem 6 (6pt) Lossy Encryption In this problem, we will explore an alternate notion of security for public key encryption schemes called *lossy encryption.* This definition of security is more powerful than CPA security, and allows us to construct other primitives like oblivious transfer and encryption schemes secure against chosen ciphertext attacks.

Lossy encryption schemes have two modes of operation: *real* and *lossy.* In *real* mode, a lossy encryption scheme behaves like a public key encryption scheme. In *lossy* mode, the ciphertexts produced by the encryption algorithm contain no information about the message that was encrypted. Formally, a lossy encryption scheme ($\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}$) has the following syntax:

- $\mathsf{Gen}(1^\lambda, \mathsf{mode})$: The Gen algorithm takes the security parameter as input (as usual). It also takes as input a mode which can be either real or lossy. In the real mode, it outputs a pair of keys (pk, sk). In the lossy mode, it outputs a lossy public key $\widetilde{pk}$.
- $\mathsf{Enc}(pk, b)$: The Enc algorithm takes a public key (either a real or a lossy public key) and a bit b and outputs a ciphertext ct. (The definition can be generalized to allow longer messages.)
- $\mathsf{Dec}(sk, ct)$: The Dec algorithm takes as input a secret key (has to be real) and outputs a decrypted bit b.

Furthermore, it has the following properties:

Correctness: The encryption scheme is correct in the real mode. That is, for any b,

$$\Pr[(pk, sk) \leftarrow \mathsf{Gen}(1^\lambda, \mathsf{real}) : b = \mathsf{Dec}(sk, \mathsf{Enc}(pk, b))] = 1$$

where the probability is over the randomness of Gen, Enc and Dec algorithms.

Key Indistinguishability: Real public keys are indistinguishable from lossy public keys. That is,

$$\{pk : (pk, sk) \leftarrow \mathsf{Gen}(1^\lambda, \mathsf{real})\} \approx_c \{\widetilde{pk} : \widetilde{pk} \leftarrow \mathsf{Gen}(1^\lambda, \mathsf{lossy})\}.$$

Lossy encryption: Encryption using the lossy key completely loses information about the message encrypted. That is, output distributions of encryptions of 0 and 1, under lossy keys, are statistically indistinguishable. Formally, for every $\widetilde{pk}$ in the support of $\mathsf{Gen}(1^\lambda, \mathsf{lossy})$,

$$\mathsf{Enc}\left(\widetilde{pk}, 0\right) \approx_s \mathsf{Enc}\left(\widetilde{pk}, 1\right)$$

where the randomness is the coins of the Enc algorithm, and $\approx_s$ means that the statistical distance between the two distributions is negligible in λ.

Part A Show that every lossy encryption scheme also satisfies

$$\left(\widetilde{pk}, \mathsf{Enc}\left(\widetilde{pk}, 0\right)\right) \approx_s \left(\widetilde{pk}, \mathsf{Enc}\left(\widetilde{pk}, 1\right)\right)$$

where $\widetilde{pk} \leftarrow \mathsf{Gen}(1^\lambda, \mathsf{lossy})$, the randomness is the coins of the Gen and Enc.

Part B Show that every lossy encryption scheme is also CPA secure (operating in the real mode).

Part C Define a lossy key generation algorithm for the Goldwasser-Micali encryption scheme. Prove the three properties above (correctness, key indistinguishability and lossy encryption) for the Goldwasser-Micali scheme with your lossy key generation algorithm, assuming the Quadratic Residuosity assumption.