Due at the start of lecture on Tuesday, April 29, 2008.

**For problem 1, no collaboration is allowed.**

**Problem 1** *Linear Encryption (30 points)*

Recently, in the cryptographic literature, the *Decision Linear* assumption has been made. Informally, this assumption is described as follows. Let $g, f, h$ be random generators of a group $G$ of prime order $q$. Given input $(g, f, h, g^a, f^b, h^c)$, where $a, b$ are random values in $\mathbb{Z}_q$, it is hard to decide if $c = (a + b) \mod q$ or not. We formalize this assumption as:

**Definition 1 (Decision Linear Assumption)** Let $G$ be a group of prime order $q$, where $q$ is $k$-bits. Then for all ppt adversaries $A$, there exists a negligible function $\epsilon$ such that

$$\Pr\left[ (g, f, h, w_0) \leftarrow G; a, b \leftarrow \mathbb{Z}_q; w_1 = h^{a+b}; d \leftarrow \{0, 1\}; d' \leftarrow A(G, q, g, f, h, g^a, f^b, w_d) : d = d' \right] \leq 1/2 + \epsilon(k).$$

Consider the following cryptosystem, which we’ll call LE for short.

**Key Generation**: $\text{Gen}$ chooses a random generator $h$ of a group $G$ of prime order $q$, chooses random values $x, y \in \mathbb{Z}_q$, sets $g = h^{1/x}$ and $f = h^{1/y}$, and outputs a public key $pk = (G, q, g, f, h)$ and $sk = (G, q, g, f, h, x, y)$.

**Encryption**: $\text{Enc}(pk, m)$, where $m \in G$, parse $pk = (G, q, g, f, h)$ and choose random values $r, s \in \mathbb{Z}_q$ and output the ciphertext $(g^r, f^s, h^{r+s} \cdot m)$.

**Decryption**: $\text{Dec}(sk, c)$, where $c = (c_1, c_2, c_3)$, output ??

1. (5 points) State a decryption algorithm $\text{Dec}$ for the above cryptosystem LE.
2. (15 points) Prove that LE is CPA-secure under the Decision Linear assumption.
3. (10 points) Prove that LE is *not* CCA2-secure.

Note that here we are referring to the *public key* definitions of CPA and CCA2 security.

**Problem 2** *Random Message Security (30 points)*

Consider the following definition of security, called random message (RM) security. We say that a cryptosystem $(\text{Gen}, \text{Enc}, \text{Dec})$ on a sequence of message spaces $\mathcal{M} = \{M_k\}$ is RM secure if for all ppt adversaries $A$, there exists a negligible function $\epsilon$ such that for all $k \in \mathbb{N}$:

$$\Pr[(pk, sk) \leftarrow \text{Gen}(1^k); m \leftarrow A(1^k, pk); r \leftarrow M_k; c_0 \leftarrow \text{Enc}(pk, m); c_1 \leftarrow \text{Enc}(pk, r); b \leftarrow \{0, 1\}; b' \leftarrow A(c_b) : b = b'] \leq 1/2 + \epsilon(k).$$

11-1
Compare this to the Goldwasser-Micali (GM) definition of security. Recall that we say that a cryptosystem \((\text{Gen}, \text{Enc}, \text{Dec})\) on a sequence of message spaces \(\mathcal{M} = \{M_k\}\) is GM secure if for all ppt adversaries \(A\), there exists a negligible function \(\epsilon\) such that for all \(k \in \mathbb{N}\):

\[
\Pr[(pk, sk) \leftarrow \text{Gen}(1^k); (m_0, m_1) \leftarrow A(1^k, pk); c_0 \leftarrow \text{Enc}(pk, m_0); c_1 \leftarrow \text{Enc}(pk, m_1); b \leftarrow \{0, 1\}; b' \leftarrow A(c_b) : b = b'] \leq 1/2 + \epsilon(k).
\]

Prove or disprove that RM security is equivalent to GM security.

**Problem 3** Hybrid Encryption (10 points, due to Katz/Lindell)

The natural way of applying hybrid encryption to the El Gamal encryption scheme is as follows. The public key is \(pk = (G, q, g, g^x)\) and secret key \(sk = (G, q, g, x)\), as in the El Gamal scheme, and to encrypt a message \(m\) the sender chooses random \(k \leftarrow \{0, 1\}^n\) and sends

\[
(g^r, g^{xr \cdot k}, \text{Enc}_k(m)),
\]

where \(r \leftarrow \mathbb{Z}_q\) is chosen at random and \(\text{Enc}\) represents a private-key encryption scheme. Suggest an improvement that results in a shorter ciphertext containing only a single element of \(G\) followed by a private-key encryption of \(m\). (You do not need to prove your answer.)

**Problem 4** Offline/Online Signatures (30 points)

Public-key signatures are quite expensive. The idea of designing offline/online signatures is to split the (expensive) signing process into two components. The offline component will prepare some information \(\sigma_1\) even before the message to be signed is known. This component could be a little slow since it is done offline. The online component is performed after the message \(m\) arrives. It uses \(\sigma_1\) (together with \(m\) and the signing key) to produce the “final” signature \(\sigma\). The online component should be “fast”.

Assume \((G, S, V)\) is a regular secure (from now on, this means existentially unforgeable under the chosen message attack) signature scheme, and let \((vk, sk) \leftarrow G(1^k)\) be the verification and signing keys of the offline/online signatures below.

1. Assume \((\text{Gen}, \text{Tag}, \text{Ver})\) is a secure MAC. Consider the following scheme. In the offline phase, pick the random MAC key \(s \leftarrow \text{Gen}(1^k)\), and sign \(s\) using the regular signing key \(\sigma_1 = S_{sk}(s)\). In the online phase, MAC the message \(m\) as \(\sigma_2 \leftarrow \text{Tag}_s(m)\). The overall signature is \(\sigma = (\sigma_1, \sigma_2, s)\). Verification is obvious. Is the resulting signature scheme secure? Either prove your answer, or give a forgery algorithm.

2. Assume \((\text{Gen}, \text{Sig}, \text{Ver})\) is a secure one-time signature scheme. Consider the following scheme. In the offline phase, pick the random one-time keys \((vk', sk') \leftarrow \text{Gen}(1^k)\), and sign \(vk'\) using the regular signing key \(\sigma_1 = S_{sk}(vk')\). In the online phase, one-time sign the message \(m\) as \(\sigma_2 = \text{Sig}_{sk}(m)\). The overall signature is \(\sigma = (\sigma_1, \sigma_2, vk')\). Verification is obvious. Is the resulting signature scheme secure? Either prove your answer, or give a forgery algorithm.

\(^1\)We use an alternative name here for a definition you already know.
Problem 5  Signature Schemes in the Random Oracle Model (10 bonus points)

In the “random oracle model”, we make the assumption that some function (e.g., a hash function) behaves as if it were a random oracle $O$; that is, for every $x \in \{0, 1\}^*$, $O(x)$ is a truly random string of some length $\ell$. For the purpose of this problem, let’s assume that, for all $x$, we have $\ell = |x|$. We assume the adversary, signer and verifier all have access to the oracle.

Assume that a trapdoor permutation family $P_{PK}$ exists, and design a simple signature scheme using $P$ and a function $h$ (which we will treat as a random oracle). Prove that the scheme is secure in the random oracle model. (Hint: in order to do this, you will have to describe exactly how the oracle $O$ works, but $O$ must still be truly random.)