CS355: Topics in cryptography **Spring 2014**

Assignment #2

Due: Tuesday, June 3, 2014.

Problem 1: Consider the following ElGamal-like encryption system in a group $\mathbb{G}$ of prime order q: the public key is $g, h \in \mathbb{G}$ and an encryption of message $m \in \{0,1\}$ is $(g^r,\ h^r g^m)$ where r chosen at random in $\mathbb{Z}_q$. Your goal is to devise an honest-verifier zero-knowledge proof for proving that an ElGamal ciphertext is an encryption of 0 or 1. That is, the proof system should recognize the language

$$\big\{(g,h,\ g^r,\ h^r)\big\}_{r\in\mathbb{Z}_q} \bigcup \big\{(g,h,\ g^r,\ h^r g)\big\}_{r\in\mathbb{Z}_q} \quad \subseteq \mathbb{G}^2 \ .$$

Remember to prove completeness, soundness, and zero-knowledge.
Hint: start from the Chaum-Pedersen protocol for proving equality of discrete-log. Generalize the protocol into an OR proof as we did in class. If you get stuck, this paper might help: `www.win.tue.nl/~berry/papers/crypto94.pdf`

Extra credit: Design an efficient zero-knowledge proof that a 4-tuple is not a Diffie-Hellman tuple. That is, the protocol should recognize the language $\{(g,h,g^r,h^s)\ :\ r \neq s\}$.

Problem 2: In this problem we consider a candidate construction for Identity Based Encryption based on the discrete-log problem in a group $\mathbb{G}$ of prime order q with generator g. The group $\mathbb{G}$ need not have a pairing.

The setup algorithm generates a random $a, b, c \in \mathbb{Z}_q$ and outputs the public parameters $\mathsf{pp} = (g,\ g_1 := g^a,\ g_2 := g^b,\ g_3 = g^c)$ and master key $\mathsf{mk} = (a, b, c)$. Let $H : \{0,1\}^* \to \mathbb{Z}_q$ be a hash function and define the secret key for identity id as $\mathsf{sk}_{\mathrm{id}} := (\mathsf{pp}, \alpha, \beta)$ where $\alpha, \beta \in \mathbb{Z}_q$ is a random pair satisfying $(a + H(\mathrm{id}))\alpha + b\beta = c$ in $\mathbb{Z}_q$. To encrypt a message $m \in \mathbb{G}$ to identity id the encryption algorithm chooses a random $r \in \mathbb{Z}_q$ and outputs the ciphertext $\mathrm{ct} := \big(\ (g^{H(\mathrm{id})} g_1)^r\ ,\ g_2^r\ ,\ m \cdot g_3^r\ \big)$.

a. Explain how the key generation algorithm, $\mathrm{KeyGen}(\mathsf{mk}, \mathrm{id})$, and decryption algorithm, $\mathrm{Dec}(\mathsf{sk}_{\mathrm{id}}, \mathrm{ct})$, work.

b. Show that if an attacker obtains the secret keys of any three identities $\mathrm{id}_1, \mathrm{id}_2, \mathrm{id}_3$ (where $H(\mathrm{id}_1), H(\mathrm{id}_2), H(\mathrm{id}_3)$ are distinct) he can completely break the system. That is, he can decrypt all ciphertexts, even those not intended for identities $\mathrm{id}_1, \mathrm{id}_2, \mathrm{id}_3$.

Problem 3: Aggregate signatures. Let $\mathbb{G}$ be a pairing group of order q where $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ denotes the pairing in $\mathbb{G}$. Let $g \in \mathbb{G}$ be a generator. In class we defined the BLS signature scheme: the public key is $\mathsf{vk} = g^\alpha$ and a signature on a message $m \in \{0,1\}^*$ is defined as $\sigma := H(\mathsf{vk}, m)^\alpha$ where $H : \mathbb{G} \times \{0,1\}^* \to \mathbb{G}$ is a hash function.

Suppose we have n public keys $\mathsf{vk}_1 = g^{\alpha_1}, \ldots, \mathsf{vk}_n = g^{\alpha_n}$ and n messages $m_1, \ldots, m_n \in \{0,1\}^*$. We are given n signatures $\sigma_i := H(\mathsf{vk}_i, m_i)^{\alpha_i}$ for $i = 1, \ldots, n$. We wish to aggregate all the signatures $\sigma_1, \ldots, \sigma_n$ into a single signature σ that will serve as a signature validating the fact that user i signed m_i for all $i = 1, \ldots, n$.

Let us define $\sigma := \prod_{i=1}^n \sigma_i$. This σ is called an *aggregate signature*. Show how a verifier, given $(\mathsf{vk}_1, m_1), \ldots, (\mathsf{vk}_n, m_n)$ and σ, can verify that indeed user i signed m_i for all $i = 1, \ldots, n$.

Note: this construction can be used to compress all the signatures in a certificate chain into a single signature. The construction can be proven secure under standard assumptions in bilinear groups.