CS-7810 Graduate Cryptography	November 5, 2017

Problem Set 4

Lecturer: Daniel Wichs — *Due: Nov 15, 2017*

Problem 1 (Better Collision Resistance from DL) 10 pts

Let $(\mathbb{G}, g, q) \leftarrow \mathsf{GroupGen}(1^n)$ be a group generation algorithm that generates a cyclic group $\mathbb{G} = \langle g \rangle$ with generator g of order $|\mathbb{G}| = q$ where q is a prime. In class we showed that, under the discrete log assumption, $H_{g,h}(x_1, x_2) = g^{x_1} h^{x_2}$ is a collision resistant hash function mapping $\mathbb{Z}_q^2 \to \mathbb{G}$ when $h \leftarrow \mathbb{G}$ is a random group element. Let's define a much more compressing function that maps $\mathbb{Z}_q^m \to \mathbb{G}$ for any m as follows:

$$H_{g_1, g_2, \ldots, g_m}(x_1, \ldots, x_m) = \prod_{i=1}^{m} g_i^{x_i}$$

where $g_1 \ldots, g_m \leftarrow \mathbb{G}$ are random group elements. Show that, under the discrete log assumption, the above is a collision resistant hash function meaning that for all PPT $\mathcal{A}$:

$$\Pr\left[\begin{array}{c} \vec{x} \neq \vec{x}' \in \mathbb{Z}_q^m \\ H_{\vec{g}}(\vec{x}) = H_{\vec{g}}(\vec{x}') \end{array} : \begin{array}{c} (\mathbb{G}, g, q) \leftarrow \mathsf{GroupGen}(1^n) \\ \vec{g} = (g_1, \ldots, g_m) \leftarrow \mathbb{G}^m \\ (\vec{x}, \vec{x}') \leftarrow \mathcal{A}(\mathbb{G}, g, , q, \vec{g}) \end{array}\right] = \mathsf{negl}(n)$$

Hint: given a discrete log challenge $g, h = g^x$ where your goal is to find x, define $g_i = g^{a_i} h^{b_i}$ for random $a_i, b_i \leftarrow \mathbb{Z}_q$.

Problem 2 (Playing with ElGamal Ciphertexts) 5 pts

Let $(\mathbb{G}, g, q) \leftarrow \mathsf{GroupGen}(1^n)$ be a group generation algorithm that generates a cyclic group $\mathbb{G} = \langle g \rangle$ with generator g of order $|\mathbb{G}| = q$ where q is a prime. Recall that the ElGamal encryption scheme has public key $pk = (g, h = g^x)$ and $sk = x$. The encryption procedure computes $\mathsf{Enc}(pk, m) = (g^r, h^r \cdot m)$ where $r \leftarrow \mathbb{Z}_q$.

- (Re-randomization) Given a public key pk and an ElGamal ciphertext c encrypting some unknown messages $m \in \mathbb{G}$ show how to create a ciphertext c' which encrypts the same message m under pk but with fresh independent randomness (i.e., given c, the ciphertexts c' should have the same conditional distribution as a fresh encryption of m under pk).
- (Plaintext Multiplication) Show that given a public key pk and any two independently generated ElGamal ciphertexts c_1, c_2 encrypting some unknown messages $m_1, m_2 \in \mathbb{G}$ respectively under the public key pk, we can efficiently create a new ciphertext c^* encrypting $m^* = m_1 \cdot m_2$ under pk without needing to know sk, m_1, m_2.

Problem 3 (Public Key Encryption – Decryption Query) 10 pts

The security definition of public-key encryption that we gave in class gives the adversary the public key which allows him to encrypt arbitrary messages himself. However, it doesn't consider that an adversary might be able to see how ciphertexts are decrypted. In this problem, you're to show that in general this can make a cryptosystem completely insecure.

A. Show that, if there exists any secure public key encryption scheme $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ according to the definition we gave in class then you can modify it to get an encryption scheme $\mathcal{E}' = (\mathsf{KeyGen}', \mathsf{Enc}', \mathsf{Dec}')$ such that:

- $\mathcal{E}'$ is a secure encryption scheme according to the definition we gave in class.
- $\mathcal{E}'$ has the property that, if the attacker can query the decryption function $\mathsf{Dec}'(sk, \cdot)$ even on a single ciphertext c of his choosing and sees the output $m = \mathsf{Dec}(sk, c)$ then the attacker can completely recover the secret key sk.

This is a very undesirable property - if the attacker can learn a single decrypted value for a ciphertext of his choosing he can completely break security of the scheme!

B. You solution in part A might have been a "contrived" scheme which is not very "natural". But there are natural schemes that are completely insecure if an adversary can see decryptions of chosen messages – for example, schemes based on the Rabin trapdoor permutation. Let $N = pq$ be a product of two primes and let $f : QR_N \to QR_N$ be the Rabin trapdoor permutation defined by $f(x) = x^2 \mod N$. We know this permutation is easily invertible given p, q. Show that if an adversary can query $f^{-1}(y)$ for a single value y of its choosing than it can efficiently factor N with non-negligible probability.

Problem 4 (Amplifying a Discrete Log Attack) 10 pts

Fix some cyclic group $\mathbb{G}$ of order q with generator g.

Suppose that there exists a randomized algorithm $\mathcal{A}$ that runs in time T and solves the discrete logarithm problem *on average* with probability .01 meaning that

$$\Pr[\mathcal{A}(g^x) = x \ : \ x \leftarrow \mathbb{Z}_q] \geq .01$$

Note that this probability is over a random x and the randomness of of $\mathcal{A}$.

Show that, if this is the case, then there is also a randomized algorithm $\mathcal{B}$ that runs in time $O(T)$ and solves the discrete logarithm problem *in the worst* case with probability .99, meaning that for *every* $x \in \mathbb{Z}_q$ we have

$$\Pr[\mathcal{B}(g^x) = x] \geq .99.$$

(For this problem, assume all group operations can be performed in $O(1)$ time.)

Note that $\mathcal{B}$ has to improve on $\mathcal{A}$ in two ways: it needs to work for a worst-case x rather than just on average, and its success probability is .99 rather than .01.