CS-7880 Graduate Cryptography October 27, 2015

Problem Set 2

Lecturer: Daniel Wichs *Due: Nov. 10, 2015*

Problem 1 (Fun with PRFs) 15 pts

Let $\{F_k : \{0,1\}^n \to \{0,1\}^n\}_{n\in\mathbb{N},k\in\{0,1\}^n}$ be a PRF family with n-bit key, n-bit input and n-bit output. For each of the following candidate constructions F' say whether F' is also necessarily a PRF. If so, give a proof else give a counter-example (if PRFs exist, then there exists a PRF F such that F' is not a PRF). Some of the constructions F' have different input/output lengths than F.

1. $F'_k(x) := F_k(x)||F_k(x+1)$ where $||$ denotes string concatenation and addition is modulo 2^n.
2. $F'_k(x) := F_k(x||0)||F_k(x||1)$ where $x \in \{0,1\}^{n-1}$.
3. $F'_k(x) := F_k(x) \oplus x$ where $\oplus$ denotes the bit-wise XOR operation.
4. $F'_k(x) := F_k(x) \oplus k$.
5. $F'_k(x) := F_x(k)$.

Problem 2 (CHRHFs are OWFs) 10 pts

Let $\{H_s : \{0,1\}^{2n} \to \{0,1\}^n\}_{n\in\mathbb{N},s\in\{0,1\}^n}$ be a collision resistant hash function family that compresses $2n$ bits to n bits. Show that H_s is a seeded one-way function in the following sense: for all PPT $\mathcal{A}$ we have

$$\Pr[H_s(x') = y \;:\; s \leftarrow \{0,1\}^n, x \leftarrow \{0,1\}^{2n}, y = H_s(x), x' \leftarrow \mathcal{A}(s,y)] = \mathsf{negl}(n).$$

Note that in the above there is no requirement that $x' \neq x$; the adversary $\mathcal{A}$ wins if it finds any pre-image of y.

Problem 3 (CPA Security - Alternate Definition) 10 pts

Let $(\mathsf{Enc}, \mathsf{Dec})$ be an symmetric-key encryption scheme with n-bit keys and $\ell(n)$-bit messages. In class, we gave a definition of CPA security by defining the following experiment $\mathbf{CPAExp}^{\mathbf{b}}_{\mathcal{A}}(\mathbf{1^n})$ with a stateful adversary[1] $\mathcal{A}$:

1. Choose $k \leftarrow \{0,1\}^n$.
2. $\mathcal{A}^{\mathsf{Enc}(k,\cdot)}(1^n) \to m_0, m_1 \in \{0,1\}^{\ell(n)}$
3. $c_b \leftarrow \mathsf{Enc}(k, m_b)$

[1] The adversary maintains state throughout the experiment and when invoked in each step it remembers what occurred in previous steps

4. $\mathcal{A}^{\mathsf{Enc}(k,\cdot)}(c_b) \to b'$
5. Output b'

We required that $\mathbf{CPAExp^0} \approx \mathbf{CPAExp^1}$ meaning that for all PPT $\mathcal{A}$ we have

$$|\Pr[\mathbf{CPAExp^0_{\mathcal{A}}(1^n)} = 1] - \Pr[\mathbf{CPAExp^1_{\mathcal{A}}(1^n)}]| = \mathsf{negl}(n).$$

Intuitively, the above definition says that encryptions of any two messages m_0, m_1 are indistinguishable even given access to the encryption oracle $\mathsf{Enc}(k, \cdot)$.

Show that the above definition implies the following alternate definition of CPA security. Define $\mathsf{Enc}^b(k, m_0, m_1) = \mathsf{Enc}(k, m_b)$ for $b \in \{0, 1\}$. Then for all PPT $\mathcal{A}$ we have:

$$\Pr[\mathcal{A}^{\mathsf{Enc}^0(k,\cdot,\cdot)}(1^n) = 1] - \Pr[\mathcal{A}^{\mathsf{Enc}^1(k,\cdot,\cdot)}(1^n) = 1] = \mathsf{negl}(n)$$

where $k \leftarrow \{0, 1\}^n$ is chosen uniformly at random.

Intuitively the alternate definition says that $\mathcal{A}$ cannot distinguish between having access to an oracle $\mathsf{Enc}^0(k, \cdot, \cdot)$ that, when given as input two message $m_0, m_1 \in \{0, 1\}^{\ell(n)}$, always encrypts m_0 vs. an oracle $\mathsf{Enc}^1(k, \cdot, \cdot)$ that always encrypts m_1. The adversary $\mathcal{A}$ can call the oracle as many times as it wants.

(Optional: show that the two definitions are actually equivalent, by also showing that the alternate definition implies the original.)

Problem 4 (Yet Another Attempt at CPA Definition) 5 pts

Let us modify the definition of CPA security by taking the experiment $\mathbf{CPAExp^b_{\mathcal{A}}(1^n)}$ defined in the previous problem and modifying step 2 so that the adversary does not get access to the encryption oracle when choosing the messages m_0, m_1. That is, step 2 becomes:

2. $\mathcal{A}(1^n) \to m_0, m_1 \in \{0, 1\}^{\ell(n)}$

Show that this modified definition is weaker than the original. In other words, show that assuming pseudorandom functions exist, you can construct a contrived scheme which satisfies the modified definition but does not satisfy the original definition.

Problem 5 (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 $\mathbb{Z}_q^2 \to \mathbb{G}$. 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 $g_1 \ldots, g_m$ are random group elements. Show that, under the discrete log assumption, the above is a collision resistant hash function meaning that for all PPT $\mathcal{A}$:

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

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 6 (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 $r \leftarrow \mathbb{Z}_q$.

- 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).

- 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.