CS6111: Foundations of Cryptography August 24, 2018

Homework 1

Instructor: Shweta Agrawal *Due: Sept 5th, 2018*

Problem 1: Negligible Functions (2+16+3 pts)

In cryptography, we usually define security by requiring that the probability of some undesirable event (e.g. Eve guesses the message) be so small that one would never notice it. To that end, we define a negligible function as follows:

Definition 1. *(Negligible function) A function $\nu(k) : \mathbb{N} \mapsto [0,1]$ is called* negligible *if for every polynomial p, there exists some $k_0 \geq 1$ such that for all $k > k_0$, $\nu(k) < |1/p(k)|$.*

In this problem we will develop some intuition for this useful concept and how to work with it.

a. Give an example of a negligible function $\nu(k)$ where $\nu(k) > 0$ for all k.

b. Suppose that ν is a negligible function. Let p be a polynomial such that $p(k) \geq 0$ for all $k > 0$. Which of the following functions are negligible?

   1) $\nu(p(k))$
   2) $p(\nu(k))$
   3) $\sum_{i=1}^{p(k)} \nu_i(k)$, where each ν_i is negligible
   4) $\nu(k) * p(k)$
   5) $\nu(k)^{\frac{1}{p(k)}}$
   6) $\nu(k)^{\frac{1}{c}}$, for some positive constant c
   7) $\frac{1}{p(k)} - \nu(k)$
   8) $\nu(k)^{-c}$, for some positive constant c

c. Suppose that $\epsilon : \mathbb{N} \mapsto [0,1]$ is not a negligible function. Does it follow that for some polynomial p (where $p(k) > 0$ for all k) and some k_0, $\epsilon(k) > 1/p(k)$ for all $k > k_0$? If your answer is yes, prove it. If your answer is no, give a counter-example.

Problem 2: One-Way Function: Definition (3+3 pts)

Recall the standard definition for a one-way function: A function $f : \{0,1\}^* \to \{0,1\}^*$ is called **one-way** if the following two conditions hold:

1. **Easy to compute:** There exists a deterministic polynomial-time algorithm A such that on input x, algorithm A outputs $f(x)$ (i.e. $(A(x) = f(x))$.

2. **Hard to invert:** For every probabilistic polynomial-time algorithm A, there exists a negligible function ν such that :

$$\Pr\left(A(1^k, f(x)) \to x' \mid x \overset{R}{\leftarrow} \{0,1\}^k \wedge f(x') = f(x)\right) \leq \nu(k)$$

   Notation: The above notation $x \overset{R}{\leftarrow} \{0,1\}^k$ means that x of length k is chosen uniformly at random from the set of k bit strings. The notation $A(1^k, f(x)) \to x'$ denotes that A takes as input $(1^k, f(x))$ and returns x'. The probability that A succeeds (i.e. $f(x') = f(x)$) is negligible.

Suppose we define the "hard to invert" part differently: A function $f : \{0,1\}^* \rightarrow \{0,1\}^*$ is called **uninvertible** if it is easy to compute f (as defined above), but there does *not* exist a probabilistic polynomial-time algorithm A such that, for every string x, on input $(1^k, f(x))$, A outputs x' such that $f(x) = f(x')$.

a. Show that if f is a one-way function, then it is an uninvertible function.

b. Below is a proof that an uninvertible function is also one-way. Is this proof correct? If not, describe where it went wrong (potentially in more than one place).

   Reduction: We show that an algorithm A that breaks the "one-wayness" of f also breaks that "uninvertibleness" of A. Thus, the reduction accomplishes the contrapositive: not one-way implies not uninvertible.

   The reduction proceeds as follows: on input $y = f(x)$, run A, giving it input y. With non-negligible probability, A outputs x' such that $f(x') = y = f(x)$. If A outputs such x', output it. Else, run A again until it does.

   Analysis of the reduction: Correctness follows because the reduction does not halt until it finds a correct x'. Expected polynomial-time follows because A outputs a correct x' with non-negligible probability $\epsilon(k)$, and $\epsilon(k) \geq 1/p(k)$ for some polynomial $p(k)$, so we need to run A $1/\epsilon(k) \leq p(k)$ times before it produces a correct x'.

   Therefore, if f is an uninvertible function, then it is also a one-way function.

Problem 3: Combining OWF (3+3+3 pts)

Let f, g be length preserving one way functions, i.e. $|f(x)| = |x|$. We will construct new functions f' using arbitrary one-way f, g. Prove or disprove that f' is one way for each of the following constructions. If it is, prove it, else provide a counter example.

a. $f'(x) = f(x) \oplus g(x)$

b. $f'(x) = f(f(x))$

c. $f'(x_1||x_2) = f(x_1)||g(x_2)$ (here $||$ denotes concatenation)

Problem 4: RSA.(4 pts)

In class we saw the RSA trapdoor permutation. In this problem we construct a candidate encryption scheme from RSA. Recall that we have a public modulus $n = p \cdot q$ where p, q are large primes. A user's public key is $e \in \mathbb{Z}^*_{\phi(n)}$ and secret key is d s.t. $e \cdot d = 1 \mod \phi(n)$. To encrypt a message m, a user computes the ciphertext as $\mathsf{CT} = m^e \mod n$ and to decrypt she computes $\mathsf{CT}^d \mod n$.

Assume that Sita and Ram have RSA keys with the *same* public modulus n but with different public exponents e_s and e_r respectively where e_s and e_r are relatively prime. Say that RSA encryption is used to send the *same* message m to both Sita and Ram. Prove that if Ravan knows n, e_s, e_r and sees the two ciphertexts $c_s = m^{e_s} \mod n$ and $c_r = m^{e_r} \mod n$, he can reconstruct the message m.

Note: this question illustrates the problem of using deterministic (as against randomized) encryption.

Problem 5: Definitions (5 pts)

Write down the definitions of eavesdropping adversary and Chosen-Plaintext-Attack (CPA) security that we saw in class for symmetric key encryption. Notice that CPA security gives more power to the adversary. Argue why CPA security is the better definition of the two.