Homework 1

Cryptography & Network Security
CS 406 : Spring 2018

Released: Tue Jan 30
Due: Mon Feb 12

CPA-Secure Symmetric-Key Encryption [Total 100 pts]

1. **Next-Bit Unpredictability implies Pseudorandomness.** [25 pts]

   Let Y denote a distribution ensemble over $\{0,1\}^n$, where n is a polynomial function of the security parameter k. Y is said to be *next-bit unpredictable* if, for all PPT algorithms B, $\max_{i\in[n]} |\Pr_{y\leftarrow Y_k}[B(y_1^{i-1}) = y_i] - 1/2|$ is negligible. Y is said to be *pseudorandom* if for all PPT A, $|\Pr_{y\leftarrow Y_k}[A(y) = 0] - \Pr_{y\leftarrow Un}[A(y) = 0]|$ is negligible, where U_n denote the uniform distribution over $\{0,1\}^n$.

   Given a PPT distinguisher A, define a PPT predictor B to be as follows:

   On input $z \in \{0,1\}^{i-1}$, pick $b \leftarrow \{0,1\}, r \leftarrow \{0,1\}^{n-i}$ and output $A(z||b||r) \oplus b$. (Here $||$ denotes concatenation)

   For each $i \in [n]$, define the distribution H_i over n-bit strings as the distribution of the string z produced by taking $y \leftarrow Y_k$, $r \leftarrow U_{n-i}$, and letting $z := y_1^i||r$. Note that $H_0 = U_n$ and $H_n = Y_k$. For parts (a)-(e), fix an $i \in [n]$.

   (a) Let $\alpha(z) := \Pr_{y\leftarrow Y_k}[y_1^{i-1} = z]$ and $q(z,b) := \Pr_{r\leftarrow\{0,1\}^{n-i}}[A(z||b||r) = 0]$ for all $z \in \{0,1\}^{i-1}$ and $b \in \{0,1\}$. Compute $\Pr_{y\leftarrow H_{i-1}}[A(y) = 0]$ in terms of these two functions.

   (b) Also, let $\beta(z) := \Pr_{y\leftarrow Y_k}[y_i = 0 \mid y_1^{i-1} = z]$. Now, compute $\Pr_{y\leftarrow H_i}[A(y) = 0]$.

   (c) For each $z \in \{0,1\}^{i-1}$, let $\gamma(z) := \Pr[B(z) = 0]$ (where the probability is over the randomness of the algorithm B above). Compute $\gamma(z)$ in terms of the quantities $q(z,0)$ and $q(z,1)$.

   (d) Show that $\Pr_{y\leftarrow Y_k}[B(y_1^{i-1}) = y_i \mid y_1^{i-1} = z] = \frac{1}{2} + 2(\beta(z) - \frac{1}{2})(\gamma(z) - \frac{1}{2})$, for each $z \in \{0,1\}^{i-1}$.

   (e) From the above parts establish that

   $$|\Pr_{y\leftarrow Y_k}[B(y_1^{i-1}) = y_i] - \frac{1}{2}| = |\Pr_{y\leftarrow H_i}[A(y) = 0] - \Pr_{y\leftarrow H_{i+1}}[A(y) = 0]|.$$

   (f) Using the fact that part (e) holds for all $i \in [n]$, show that

   $$|\Pr_{y\leftarrow Y_k}[A(y) = 0] - \Pr_{y\leftarrow U_n}[A(y) = 0]| \leq n \cdot \max_{i\in[n]} |\Pr_{y\leftarrow Y_k}[B(y_1^{i-1}) = y_i] - \frac{1}{2}|.$$

   If Y is next-bit unpredictable, then the RHS above is negligible (n being polynomial in k), and hence so is the LHS. Since this holds for every PPT adversary A, we conclude that if Y is next-bit unpredictable, then it is pseudorandom.

   In the lecture we saw that pseudorandomness implies next-bit unpredictability. The above completes the argument that the two definitions are equivalent.

2. **Impossibility of deterministic CPA-secure encryption.** Suppose a symmetric key encryption scheme has a deterministic encryption algorithm. Give an adversary in the IND-CPA experiment for SKE to show that this scheme cannot be CPA-secure. [15 pts]

   A consequence of the above is that the so-called "Electronic Code Book" mode of using a block-cipher is not an IND-CPA secure SKE scheme.

3. **One-Timeness of One-Time Pad.** Consider a deterministic "two-message encryption scheme" to be a function $\mathsf{Enc}^2 : \mathcal{K} \times \mathcal{M} \times \mathcal{M} \to \mathcal{C} \times \mathcal{C}$.

   (a) Define perfect secrecy for such an encryption scheme. [7 pts]

   (b) Let $\mathcal{M} = \mathcal{K} = \mathcal{C}$ be the set of n-bit strings. Let $\mathsf{Enc}^2(K, m_1, m_2) = (K \oplus m_1, K \oplus m_2)$, where $\oplus$ is bit-wise xor-ing. Prove that this is **not** perfectly secret, according to your definition. [8 pts]

   In particular, using a one-time pad to encrypt two messages will break perfect secrecy.

4. **Statistical Indistinguishability.** Recall that for two distributions X and Y over n-bit strings, the *statistical difference* (a.k.a. variational distance) between them is denoted by

$$\Delta(X, Y) = \max_{S \subseteq \{0,1\}^n} |\Pr_{x \leftarrow X}[x \in S] - \Pr_{x \leftarrow Y}[x \in S]|.$$

   (Alternately, this can be phrased in terms of a statistical test T, which checks if $x \in S$ for some subset S.)

   (a) Suppose $G : \{0,1\}^k \to \{0,1\}^n$ is a deterministic function, where $n > k$. Let X be the distribution of the output of $G(s)$ when $s \leftarrow \{0,1\}^k$ is chosen uniformly at random. Let Y be the uniform distribution over $\{0,1\}^n$. Show that $\Delta(X, Y) \geq \frac{1}{2}$. Conclude that the output of a pseudorandom random generator is quite distinguishable from a truly random distribution, if computationally unbounded distinguishers are considered. [10 pts]

   (b) Suppose X_k and Y_k are distributions over 2-bit strings (for all integers $k > 0$). Further suppose that for all values of k, $\Delta(X_k, Y_k) \geq 0.1$. Show that X_k and Y_k are *not* computationally indistinguishable.

   You may use *non-uniform* PPT distinguishers. i.e., describe a family of distinguishers D_k, each of which runs in time polynomial in k such that $|\Pr_{x \leftarrow X_k}[D_k(x) = 0] - \Pr_{x \leftarrow Y_k}[D_k(x) = 0]| \geq \epsilon(k)$ for some function ϵ that is not negligible. [10 pts]

   Show that X_k and Y_k are in fact distinguishable by a *uniform* PPT distinguisher. [Extra Credit]

5. **PRG and PRF.** True or False (give reasons): [15 pts]

   (a) If $G : \{0,1\}^k \to \{0,1\}^n$ is a PRG, then so is $G' : \{0,1\}^{k+\ell} \to \{0,1\}^{n+\ell}$ defined as $G'(x \circ x') = G(x) \circ x'$ where $x \in \{0,1\}^k$, $x' \in \{0,1\}^\ell$, and $\circ$ denotes concatenation.

   (b) If $F : \{0,1\}^k \times \{0,1\}^m \to \{0,1\}^n$ is a PRF, then so is

      i. $F' : \{0,1\}^k \times \{0,1\}^{m+\ell} \to \{0,1\}^{n+\ell}$ defined as $F'(s; x \circ x') = F(s; x) \circ x'$ where $s \in \{0,1\}^k$, $x \in \{0,1\}^m$, $x' \in \{0,1\}^\ell$.

      ii. $F' : \{0,1\}^{k+\ell} \times \{0,1\}^m \to \{0,1\}^{n+\ell}$ defined as $F'(s \circ s'; x) = F(s; x) \circ s'$ where $s \in \{0,1\}^k$, $x \in \{0,1\}^m$, $s' \in \{0,1\}^\ell$.

6. **Block Ciphers**

   (a) A PetaFLOPS computer can execute over 10^{15} floating point operations per second. Below, you may suppose that a single evaluation of a block-cipher (DES or AES) takes 10 FLOPs.

   Consider an adversary in the IND-CPA experiment against a symmetric key encryption algorithm implemented using a block-cipher in the CTR mode. Describe a brute-force strategy for the adversary to recover the encryption key. If the adversary uses a PetaFLOPS computer and the block-cipher used is DES (which uses 56 bit keys), how long would your strategy take on the average to recover the key (ignoring time taken to acquire the ciphertexts)? What if the block-cipher used is AES with 128-bit keys? [10 pts]

   (b) The triple-DES (3DES) is a block-cipher that uses the DES block-cipher three times, with three different keys. The output ("ciphertext") of 3DES with key (K_1, K_2, K_3), on input ("plaintext") P is defined as $C = \mathrm{DES}_{K_1}(\mathrm{DES}^{-1}_{K_2}(\mathrm{DES}_{K_3}(P)))$ where DES_K and DES^{-1}_K stand for the application of the DES block-cipher in the forward ("encryption") and reverse ("decryption") directions.

   Your goal is to design a key-recovery algorithm for an adversary in the IND-CPA experiment for an SKE scheme using 3DES in CTR mode. Your algorithm can use the DES block-cipher as a black-box (in either forward or reverse directions).

   Can you devise an algorithm which calls the DES block-cipher "only" about 2^{112} times. How much memory does your algorithm use? [Extra Credit]