CSE5014: Cryptography and Network Security
2024 Spring Semester Written Assignment # 2
Due: Apr. 16th, 2024, please submit at the beginning of class

Sample Solutions

Q.1 In the following, $g(n)$ and $h(n)$ are both *negligible* functions, and p is a *polynomially-bounded* function. For each function below, decide if the function is guaranteed to be negligible or not. If so, prove it. If not, provide an example of negligible g and h and polynomially bounded p such that the function below is *not* negligible.

(a) $f_a(n) = \max(g(n), h(n))$.

(b) $f_b(n) = \max\left(g(n), \frac{1}{p(n)}\right)$.

(c) $f_c(n) = \min(g(n), h(n))$.

(d) $f_d(n) = \min\left(g(n), \frac{1}{p(n)}\right)$.

(e) $f_e(n) = g(n) + h(n)$.

(f) $f_f(n) = p(n)h(n)$.

(g) $f_g(n) = g(n)^{1/2}$.

(h) $f_h(n) = g(n)^{\frac{1}{\log n}}$.

Solution: We use the definition of *negligible function* in Katz & Lindell book. A function f is *negligible* if for every polynomial $p(\cdot)$ there exists an N such that for all integers $n > N$ it holds that $f(n) < \frac{1}{p(n)}$.

(a) Yes. Since both g and h are negligible, we know that there exist N_1 and N_2, such that for all integers $n > N_1$, it holds that $g(n) < \frac{1}{p(n)}$ and for all integers $n > N_2$, it holds that $h(n) < \frac{1}{p(n)}$ for every polynomial $p(\cdot)$. Therefore, for all integers $n > \max(N_1, N_2)$, both $g(n)$ and $h(n)$ are less than $\frac{1}{p(n)}$ for every polynomial $p(\cdot)$, so is $f_a(n) = \max(g(n), h(n))$.

(b) No. We give a counterexample: let $g(n) = 2^{-n}$, and $p(n) = n^2$. Then for large enough n, $f_b(n) = \max(2^{-n}, n^{-2}) = n^{-2}$, which is not negligible.

(c) Yes. The proof is similar to (a).

(d) Yes. By definition, there exists an N such that for all integers $n > N$ it holds that $g(n) < \frac{1}{p(n)}$ for every polynomial $p(n)$. Thus, for all integers $n > N$, we have

$$f_d(n) = \min\left(g(n), \frac{1}{p(n)}\right) = g(n),$$

which is negligible.

(e) Yes. Since both f and g are negligible, we know that there exist N_1 and N_2, such that for all integers $n > N_1$, it holds that $f(n) < \frac{1}{p(n)}$ and for all integers $n > N_2$, it holds that $g(n) < \frac{1}{p(n)}$ for every polynomial $p(\cdot)$. Therefore, for all integers $n > \max(N_1, N_2)$, both $f(n)$ and $g(n)$ are less than $\frac{1}{p(n)}$ for every polynomial $p(\cdot)$. Then $f_e(n) = g(n) + h(n) < \frac{2}{p(n)} = \frac{1}{p'(n)}$.

(f) Yes. By definition, there exists an N such that for all integers $n > N$ it holds that $h(n) < \frac{1}{p'(n)}$ for every polynomial $p'(n)$. Thus, for all integers $n > N$, we have $f_f(n) = p(n)h(n) < \frac{p(n)}{p'(n)} < \frac{1}{p'(n)/p(n)}$, where $p'(n)/p(n)$ is again a polynomial.

(g) Yes. Similar to (f) ($(p(n)^{1/2}$ is also a polynomial).

(h) No. Counterexample: let $g(n) = n^{-\log n}$ which is negligible, then $f_h(n) = g(n)^{\frac{1}{\log n}} = n^{-1}$, which is not negligible.

$\square$

Q.2 Prove that the property of computational indistinguishability: if $X_n \approx Y_n$ (X_n and Y_n are computationally distinguishable) and f is a polynomial-time computable function, then $f(X_n) \approx f(Y_n)$.

Solution: Since $X_n \approx Y_n$, for every polynomial-time A and sufficiently large n, there exists a negligible function ϵ such that

$$|\Pr[A(X_n) = 1] - \Pr[A(Y_n) = 1]| \leq \epsilon(n).$$

For every polynomial-time algorithm A, we consider a new algorithm A', which first computes $f(X_n)$, and then takes $f(X_n)$ as input of the algorithm A. Since f is a polynomial-time computable function, we see that the algorithm A' is also polynomial-time. Thus, we have

$$|\Pr[A'(X_n) = 1] - \Pr[A'(Y_n) = 1]| \leq \epsilon(n),$$

which is

$$|\Pr[A(f(X_n)) = 1] - \Pr[A(f(Y_n)) = 1]| \leq \epsilon(n).$$

This means that $f(X_n) \approx f(Y_n)$.

□

Q.3 Prove that if (Gen, Enc, Dec) is a *computationally secure* encryption with $\ell(n)$-long messages, then for every polynomial-time algorithm Eve and large enough n, the probability that Eve wins in the following game is smaller than 0.34:

1. Eve gets as inputs 1^n, and gives Alice three strings $x_0, x_1, x_2 \in \{0,1\}^{\ell(n)}$.
2. Alice chooses a random key $k \leftarrow_R \{0,1\}^n$ and $i \leftarrow_R \{0,1,2\}$ and computes $y = Enc_k(x_i)$.
3. Eve gets y as input, and outputs an index $j \in \{0,1,2\}$.
4. Eve *wins* if $j = i$.

Note: This proof can be generalized to show that the probability that Eve guesses which one of c messages was encrypted is at most $1/c + \epsilon(n)$ where ϵ is a negligible function.

Solution: Define $p_{i,j} := \Pr[Eve(Enc_{U_n}(x_i)) = j]$. Clearly, we know that

$$\sum_{j=0}^{2} p_{i,j} = 1,$$

for each $i = 0, 1, 2$. On the other hand, by the definition of computational security, for every polynomially-bounded ϵ and large enough n, we have

$$|p_{i,j} - p_{k,j}| \leq \epsilon(n),$$

where $i, j, k \in \{0, 1, 2\}$. It then follows that $p_{k,j} \geq p_{j,j} - \epsilon$ for $k \neq j$. Then we have

$$\begin{aligned} 3 &= \sum_{i=0}^{2} \sum_{j=0}^{2} p_{i,j} \\ &\geq 3 \sum_{i=0}^{2} p_{i,i} - 6\epsilon(n). \end{aligned}$$

Thus, we have

$$\frac{1}{3} \sum_{i=0}^{2} p_{i,i} \leq \frac{1}{3} + \frac{2}{3}\epsilon(n) < 0.34,$$

for large enough n.

$\square$

Q.4 We call a sequence $\{X_n\}_{n \in \mathbb{N}}$ of distributions *pseudorandom* if it's computationally indistinguishable from the sequence $\{U_n\}$ where U_n is the uniform distribution over $\{0, 1\}^n$. Are the following sequences pseudorandom? Prove or refute it.

1. $\{X_n\}$ where X_n is the following distribution: we pick $x_1, \ldots, x_{n-1}$ uniformly at random in $\{0, 1\}^{n-1}$, and let x_n be the parity (i.e., XOR) of $x_1, \ldots, x_{n-1}$, we output $x_1, \ldots, x_n$.
2. $\{Z_n\}$ where for n large enough, with probability $2^{-n/10}$ we output an n bit string encoding the text "This is not a pseudorandom distribution" (say encode in ASCII and pad with zeros), and with probability $1 - 2^{-n/10}$ pick a random string. For n that is not large enough to encode the text, Z_n always outputs the all zeros string.

Solution:

1. No. We define a polynomial-time algorithm A as: for a sequence $x_0, x_1, \ldots, x_n$, check whether $x_n = x_0 \oplus x_1 \oplus \cdots \oplus x_{n-1}$, if so, output 1, otherwise output 0. Then it is easy to check that

$$|\Pr[A(X_n) = 1] - \Pr[A(U_n) = 1]| = |1 - \frac{1}{2}| = \frac{1}{2},$$

which means X_n is not pseudorandom.

2. Yes. Since for large enough n, Z_n is the same as U_n with probability $1 - 2^{-n/10}$. Thus, we have, for every polynomial-time algorithm A,

$$|\Pr[A(Z_n) = 1] - \Pr[A(U_n) = 1]| \leq 2^{-n/10}.$$

Note that $2^{-n/10}$ is a negligible function. Therefore, $\{Z_n\}$ is pseudorandom.

$\square$

Q.5 Let G be a pseudorandom generator where $|G(s)| > 2|s|$. Take $s = s_1 \cdots s_n$, and for simplicity, n even.

(1) Define $G'(s) := G(s0^{|s|})$. Is G' necessarily a pseudorandom generator?

(2) Define $G'(s) := G(s_1 \cdots s_{n/2})$, where $s = s_1 \cdots s_n$. Is G' necessarily a pseudorandom generator?

Solution:

(1) Such a $G'(s)$ is not a PRG because otherwise $G(0^{|s|})$ would also be a PRG which is clearly not the case: As D knows G and its expansion factor ℓ, it can on output r just check whether $r = G(0^n)$ for n with $\ell(n) = |r|$. Then D wins with probability $1 - 2^{-|r|}$ which is not negligible.

(2) Yes. Let $|G(s)| = \ell(n)$ and

$$\epsilon := |\Pr_{r \leftarrow \{0,1\}^{\ell(n)}}[D(r) = 1] - \Pr_{s \leftarrow \{0,1\}^n}[D(G'(s)) = 1]|,$$

for a probabilistic polynomial-time distinguisher D. By definition of G' we have:

$$\Pr_{s \leftarrow \{0,1\}^{n/2}}[D(G'(s)) = 1] = \Pr_{s \leftarrow \{0,1\}^{n/2}}[D(G(s_1 \cdots s_{n/2} \cdot 0^{n/2})) = 1],$$

and thus

$$|\Pr_{r \leftarrow \{0,1\}^{\ell(n)}}[D(r) = 1] - \Pr_{s \leftarrow \{0,1\}^{n/2}}[D(G(s_1 \cdots s_{n/2} \cdot 0^{n/2})) = 1]| = \epsilon(n) = \epsilon'(n/2),$$

where $\epsilon'(n) := \epsilon(2n)$. Since ϵ' must be negligible, we conclude that ϵ is negligible.

$\square$

Q.6 Define the keyed, length-preserving function F_k by $F_k(x) = F(k,x) = k \oplus x$. It is known that for any input x, the value of $F_k(x)$ is uniformly distributed if k is uniformly chosen. Prove or disprove that F_k is a PRF or not.

Solution: F_k is not a PRF. We may construct a distinguisher D as follows. D queries its oracle $\mathcal{O}$ on arbitrary, distinct points x_1, x_2 to obtain values $y_1 = \mathcal{O}(x_1)$ and $y_2 = \mathcal{O}(x_2)$ and outputs 1 if and only if $y_1 \oplus y_2 = x_1 \oplus x_2$.

On one hand, if $\mathcal{O} = F_k$, for any k, then D outputs 1.

On the other hand, if $\mathcal{O} = f$ for f chosen uniformly from $Func_n$, the probability that $f(x_1) \oplus f(x_2) = x_1 \oplus x_2$ is exactly the probability that $f(x_2) = x_1 \oplus x_2 \oplus f(x_1)$, i.e., D outputs 1 with probability 2^{-n}. The difference is $|1 - 2^{-n}|$, which is not negligible.

$\square$

Q.7 Prove that if F_k is a length-preserving PRF, then

$$G(S) = F_s(\langle 1 \rangle)|F_s(\langle 2 \rangle)| \cdots |F_s(\langle \ell \rangle)$$

is a PRG with expansion factor $\ell \cdot n$, where $\langle i \rangle$ denotes the n-bit binary expression of the number i.

Solution: Since F is length-preserving, the expansion faction is immediate. We will prove that G is a PRG by reduction. Assume that there is a PPT algorithm D that can distinguish between $G(s)$ and a random string r with non-negligible probability, we construct an adversary D_F that can distinguish F_k from a random function as follows.

Given input 1^n and access to an oracle $\mathcal{O}$, we compute $t = \mathcal{O}(\langle 1 \rangle)|\mathcal{O}(\langle 2 \rangle)| \cdots |\mathcal{O}(\langle \ell \rangle)$ by posing ℓ queries for the oracle $\mathcal{O}$. Then we simulate D on t and output 1 if and only if D outputs 1. Now we have

$$\begin{aligned}
&\left|\Pr_{s\leftarrow\{0,1\}^n}[D_F^{F_s(\cdot)}(1^n) = 1] - \Pr_{f\leftarrow Func_n}[D_F^{f(\cdot)}(1^n) = 1]\right| \\
&= \left|\Pr_{s\leftarrow\{0,1\}^n}[D(F_s(\langle 1 \rangle)| \cdots |F_s(\langle \ell \rangle)) = 1] - \Pr_{f\leftarrow Func_n}[(f(\langle 1 \rangle)| \cdots |f(\langle \ell \rangle)) = 1]\right| \\
&= \left|\Pr_{s\leftarrow\{0,1\}^n}[D(G(s)) = 1] - \Pr_{r_1\leftarrow\{0,1\}^n,\ldots,r_\ell\leftarrow\{0,1\}^n}[D(r_1| \cdots |r_\ell) = 1]\right| \\
&= \left|\Pr_{s\leftarrow\{0,1\}^n}[D(G(s)) = 1] - \Pr_{r\leftarrow\{0,1\}^{\ell\cdot n}}[D(r) = 1]\right| \\
&\geq \frac{1}{n^c}
\end{aligned}$$

for a constant c as D can distinguish with non-negligible probability by assumption. Therefore, D_F can also distinguish with non-negligible probability.

$\square$

Q.8 Consider a variant of CBC-mode encryption where the sender simply increments the IV by 1 each time a message is encrypted (rather than choosing IV at random each time). Show that the resulting scheme is *not* CPA-secure.

Solution: We design an adversary $\mathcal{A}$ that wins over guessing with non-negligible probability as follows:

(a) Query the encryption oracle with $m = 0^{n-1}1$ and receive a ciphertext $\langle IV, c\rangle$.

(b) If IV is odd, i.e., has as last bit 1, then output a random bit.

(c) If IV is even, i.e., has as last bit 0, then output $m_0 = 0^n$ and arbitrary m_1 to be encrypted.

(d) Receive the challenge ciphertext $\langle IV + 1, c'\rangle$, and output 0 if $c' = c$, and 1 otherwise.

We claim that this adversary succeeds with probability that is greater than 1/2 by a non-negligible function (in fact, even a constant). First, by guessing randomly, $\mathcal{A}$ succeeds with probability 1/2 if IV is odd, which is 1/4 of the cases.

If IV is even, then $IV + 1 = IV \oplus 0^{n-1}1$. Therefore, $c = F_k(IV \oplus m_0) = F_k(IV \oplus 0^{n-1}1) = F_k(IV + 1) = F_k(IV + 1 \oplus 0) = F_K((IV + 1) \oplus m_0)$, and so if m_0 was encrypted, then $c = c'$. On the other hand, if m_1 was encrypted, then $c \neq c'$. That is, whenever IV is even, $\mathcal{A}$ decides correctly which message was encrypted. This covers exactly 1/2 of the cases.

In total, this shows that $\mathcal{A}$ wins in 3/4 of all cases.

$\square$

Q.9 Recall that in the CBC mode (see Fig. Q.9 (1)), the ciphertext blocks are generated by applying the block cipher to the XOR of the current plaintext block and the previous ciphertext block, i.e., $c_0 = IV$ and then for $i = 1$ to

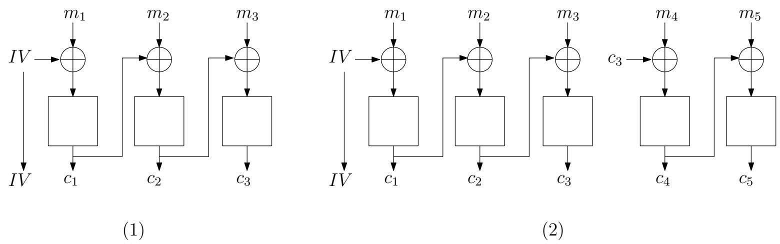

Figure 1: Q.9: CBC mode and Chained CBC mode

ℓ, $c_i = F_k(c_{i-1} \oplus m_i)$; the final ciphertext is $\langle c_0, c_1, \ldots, c_\ell \rangle$. Now we consider a variant of the CBC mode, called *chained CBC* mode (see Fig. Q.9 (2)), in which the last block of the previous ciphertext is used as IV when encrypting the next message. This reduced the bandwidth, since the IV need not be sent each time. In Fig. Q.9 (2), an initial message m_1, m_2, m_3 is encrypted using a random IV, and subsequently a second message m_4, m_5 is encrypted using c_3 as the IV. However, the chained CBC mode is not as secure as CBC mode. Please provide a chosen-plaintext attack.

Solution: Assume that the attacker knows that $m_1 \in \{m_1^0, m_1^1\}$, and observes the first ciphertext IV, c_1, c_2, c_3.

The attacker then requests an encryption of a second message m_4, m_5 with $m_4 = IV \oplus m_1^0 \oplus c_3$, and observes a second ciphertext c_4, c_5. One can verify that $m_1 = m_1^0$ if an only if $c_4 = c_1$, since

$$c_4 = F_k(m_4 \oplus c_3) = F_k(IV \oplus m_1^0 \oplus c_3 \oplus c_3) = F_k(IV \oplus m_1^0).$$

□