CSE5014 · Cryptography and Network Security

Assignment 2

Liu Leqi / 12011327

April 10, 2023

1 Q1

1. Let $p(n)$ be a positive polynomial. By the definition of negligible function, there exist $N_1, N_2 \in N^+$ such that

$$\forall n \geq N_1, negl_1(n) < 1/p(n)$$
$$\forall n \geq N_2, negl_2(n) < 1/p(n)$$

Let $N' = max(N_1, N_2)$. Then for $n \geq N'$, we have

$$\begin{aligned} negl_3(n) &= negl_1(n) + negl_2(n) \\ &< 1/p(n) + 1/p(n) \\ &= 2/p(n) = 1/p'(n) \end{aligned}$$

where $p'(n) = p(n)/2$ also a polynomial. Thus, $negl_3$ is also negligible.

2. Let $p''(n) = p(n)p'(n)$ be a positive polynomial as the product of two positive polynomials. By the definition of negligible function, there exist $N_1 \in N^+$ such that

$$\forall n \geq N_1, negl_1(n) < 1/p''(n) = 1/p(n)p'(n)$$

Then for $n \geq N_1$, we have

$$\begin{aligned} negl_4(n) &= p(n)negl_1(n) \\ &< p(n)/p(n)p'(n) \\ &= 1/p'(n) \end{aligned}$$

Thus, $negl_4$ is also negligible.

2 Q2

Suppose there is a polynomial-time algorithm A. Since A and f are both polynomial-time, then the composite $A \circ f$ is also polynomial-time.
Since $X_n \approx Y_n$, then there is a negligible function ϵ such that $|Pr[A \circ f(X_n) = 1] - Pr[A \circ f(Y_n) = 1]| \leq \epsilon(n)$, i.e., $|Pr[A(f(X_n)) = 1] - Pr[A(f(Y_n)) = 1]| \leq \epsilon(n)$. Therefore, $f(X_n) \approx f(Y_n)$.

3 Q3

Assume that there exists a polynomial-time algorithm Eve such that it can win the game with probability no smaller than 0.34 ($i.e.$, 1/3) for large enough n, i.e.,

$$Pr[Eve\ wins] \geq \frac{1}{3}$$

Since there are 3 possible choices for i, the probability of Eve winning the game by guessing randomly is 1/3. Then we can define the advantage of Eve as following,

$$Adv(Eve) = |Pr[Eve\ wins] - \frac{1}{3}| = Pr[Eve\ wins] - \frac{1}{3}$$

Besides, in terms of law of total probability, we have

$$\begin{aligned}
Adv(Eve) &= Pr[Eve\ wins] - \frac{1}{3} \\
&= Pr[Eve\ correctly\ guesses\ i] - \frac{1}{3} \\
&= (Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i=0]Pr[|Alice\ chooses\ i=0] \\
&\quad + Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i=1]Pr[Alice\ chooses\ i=1] \\
&\quad + Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i=2]Pr[Alice\ chooses\ i=2]) - \frac{1}{3} \\
&= \frac{1}{3}(Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i=0] \\
&\quad + Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i=1] \\
&\quad + Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i=2] - 1)
\end{aligned}$$

Alice chooses $i \leftarrow_R \{0,1,2\}$, so $Pr[Alice\ chooses\ i=j] = 1/3$ for $j \in \{0,1,2\}$. And there follows 2 cases,

- $Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i=j]$
  $= Pr[Eve\ outputs\ j] - Pr[Eve\ incorrectly\ guesses\ i|Alice\ chooses\ i \neq j]$
  $= Pr[Eve\ outputs\ j] - 1 + Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i \neq j]$
- $Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i \neq j] \leq |Pr[Eve(E_k(x_i)) = x_i] - Pr[Eve(E_k(x_j)) = x_i]|$

Since the scheme $\Pi = (Gen, Enc, Dec)$ is computationally secure, then

$$|Pr[Eve(E_k(x_i)) = x_i] - \frac{1}{2}| \leq \epsilon(n)$$

where ϵ is a negligible function.
Since $|Pr[Eve(E_k(x_i)) = x_i] - 1/2| \leq \epsilon(n)$ and $|Pr[Eve(E_k(x_j)) = x_i] - 1/2| \leq \epsilon(n)$ as computationally secure scheme, we have

$$\begin{aligned}
Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i \neq j] &\leq |Pr[Eve(E_k(x_i)) = x_i] - Pr[Eve(E_k(x_j)) = x_i]| \\
&= |(Pr[Eve(E_k(x_i)) = x_i] - 1/2) - (Pr[Eve(E_k(x_j)) = x_i] - 1/2)| \\
&= |Pr[Eve(E_k(x_i)) = x_i] - 1/2| + |Pr[Eve(E_k(x_j)) = x_i] - 1/2| \\
&\leq 2\epsilon(n)
\end{aligned}$$

Since there are only 3 choices $j \in \{0,1,2\}$, $Pr[Eve\ outputs\ 0] + Pr[Eve\ outputs\ 1] + Pr[Eve\ outputs\ 2] = 1$.
Therefore, we can bound the advantage of Eve as

$$\begin{aligned}
Adv(Eve) &= \frac{1}{3}(Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i=0] \\
&\quad + Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i=1] \\
&\quad + Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i=2] - 1) \\
&= \frac{1}{3}(Pr[Eve\ outputs\ 0] + Pr[Eve\ outputs\ 1] + Pr[Eve\ outputs\ 2] \\
&\quad + Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i \neq 0] \\
&\quad + Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i \neq 1] \\
&\quad + Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i \neq 2] - 4) \\
&= \frac{1}{3}(Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i \neq 0] \\
&\quad + Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i \neq 1] \\
&\quad + Pr[Eve\ correctly\ guesses\ i|Alice\ chooses\ i \neq 2] - 3) \\
&\leq \frac{1}{3}(3 \cdot 2\epsilon(n) - 3) \\
&= 2\epsilon(n) - 1
\end{aligned}$$

Since by definition the advantage of Eve is greater than or equal to 0, then $\epsilon(n) \geq 1/2$, which contradicts that ϵ is negligible. Thus, the probability that Eve wins in the following game is smaller than $1/3$, i.e., 0.34.

4 Q4

1. Not pseudorandom.
Suppose the input is y, there is a distinguisher D such that it outputs 1 if and only if the final bit of y is equal to the XOR of all the preceding bits of y. At this time, we have $\Pr[D(\{X_n\}) = 1] = 1$ but $\Pr[D(\{U_n\}) = 1] = \frac{1}{2}$. The advantage is a constant $\frac{1}{2}$ which is not negligible. Thus, the sequence $\{X_n\}$ is not pseudorandom.

2. Not pseudorandom.
Suppose the input is y, there is a distinguisher D with subroutine A such that A outputs 0^n if n is not large enough to encode the text "This is not a pseudorandom distribution".; outputs y originally otherwise. At this time, we have $\Pr[D(\{Z_n\}) = 1] = 2^{-n/10}$ but $\Pr[D(\{U_n\}) = 1] = 2^{-n} + \epsilon(n)$, where $\epsilon(n)$ is negligible, since only when y is the encoding ASCII of the text "This is not a pseudorandom distribution" or 0^n, can D outputs 1. Therefore, the advantage is $2^{-n/10} - 2^{-n} - \epsilon(n)$, which is non-negligible. Thus, the sequence $\{Z_n\}$ is not pseudorandom.

5 Q5

1. G' is not necessarily a pseudorandom generator. G is pseudorandom for random input in $\{0, 1\}^{2|s|}$, for which the probability is $2^{-2|s|}$, but the probability of an input of $s0^|s|$ is only $2^{-|s|}$. So input of this are not random and therefore the output need not be pseudorandom.

2. G' is necessarily a pseudorandom generator. Suppose $|G(s)| = l(n)$. Since G is pseudorandom, there is a distinguisher D such that

$$|\Pr_{y \leftarrow U_{l(n)}}[D(y) = 1] - \Pr_{s \leftarrow U_n}[D(G(s)) = 1]| \leq \epsilon(n)$$

Then suppose there is a distinguisher D' for G'. If the challenger provides a uniform distributed string y, the success probability is

$$\Pr_{y \leftarrow U_{l(n)}}[D'(y) = 1]| = \Pr_{y \leftarrow U_{l(n)}}[D(y) = 1] = \frac{1}{2}$$

If the challenger provides a string $G'(s)$, the success probability is

$$\Pr_{s \leftarrow U_n}[D'(G'(s)) = 1] = \Pr_{s \leftarrow U_n}[D'(G(s_1 ... s_{n/2})) = 1] = \Pr_{s \leftarrow U_n}[D(G(s)) = 1]$$

Therefore,

$$|\Pr_{y \leftarrow U_{l(n)}}[D'(y) = 1] - \Pr_{s \leftarrow U_n}[D'(G'(s)) = 1]| \leq \epsilon(n)$$

If G is pseudorandom, then $\epsilon(n)$ is negligible. So G' is necessarily pseudorandom.

6 Q6

$F_k = k \oplus x$ is not a PRF.
Suppose the distinguisher D has oracle O. D will output 1 in the game if and only if $O(x_1) \oplus O(x_2) = x_1 \oplus x_2$. If $O = F_k$, for any k, then D always outputs 1. If $O = f$, for f chosen uniformly from $Func_n$, then

$$\Pr[f(x_1) \oplus f(x_2) = x_1 \oplus x_2] = \Pr[f(x_1) = x_1 \oplus x_2 \oplus f(x_2)] = 2^{-n}$$

since $f(x_1)$ is uniform and independent of $x_1, x_2, f(x_2)$. Therefore, $\Pr[D^{F_k(\cdot)}(1^n) = 1] = 1$ and $\Pr[D^{f(\cdot)}(1^n) = 1] = 2^{-n}$, and thus the advantage $1 - 2^{-n}$ is not negligible.

7 Q7

Suppose there is an efficient algorithm A that attacks G with advantage at most $\epsilon(n)$.

$$|\Pr_{y \leftarrow U_{l \cdot n}}[A(y) = 1] - \Pr_{x \leftarrow U_n}[A(G(x)) = 1]| \leq \epsilon(n)$$

In the view of A, if the challenger gives a uniform distributed string y, then the success probability is

$$\Pr_{y \leftarrow U_{l \cdot n}}[A(y) = 1] = \frac{1}{2}$$

If the challenger gives a pseudorandom distributed string $G(x)$, then the success probability is

$$\Pr_{x \leftarrow U_n}[A(G(x)) = 1]$$

Suppose there is an efficient algorithm D with A as a subroutine to attack F_k. In the view of D, if the challenger gives truly random function f, then D will compute using f with input $1^{l \cdot n}$ and give the result to A. Since the result is random, the success probability is

$$\Pr_{f \leftarrow Func_n}[D^{f(\cdot)}(1^{l \cdot n}) = 1] = \Pr_{y \leftarrow U_{l \cdot n}}[A(y) = 1] = \frac{1}{2}$$

If the challenger gives a pseudorandom function F_k, then D will still compute using F_k on each n-bit on input $1^{l \cdot n}$ and give the result to A. If F_k is length-preserving, then $G(S)$ is also length-preserving. Since $F_k(< i >)$ is n-bit, D will output n-bit string for each input $< i >$. All the output of D comes to $G(S) = F_s(< 1 >)|F_s(< 2 >)|...|F_s(< l >)$. Therefore, the success probability is

$$\Pr_{k \leftarrow \{0,1\}^n}[D^{F_k(\cdot)}(1^{l \cdot n}) = 1] = \Pr_{x \leftarrow U_n}[A(G(x)) = 1]$$

After all, we can write

$$|\Pr_{f \leftarrow Func_n}[D^{f(\cdot)}(1^{l \cdot n}) = 1] - \Pr_{k \leftarrow \{0,1\}^n}[D^{F_k(\cdot)}(1^{l \cdot n}) = 1]| \leq \epsilon(n)$$

which says that if $\epsilon(n)$ is negligible, then the advantage of D is negligible. Thus, if F_k is a length-preserving PRF, then G is a PRG with expansion factor $l \cdot n$.

8 Q8

Suppose attacker A outputs messages m_1, m_2 of the same length. Then challenger chooses $b \leftarrow \{1, 2\}$ and encrypts $c \leftarrow Enc_k(m_b)$ and then gives c to A.
A can know that $c = IV||c_1||c_2||...||c_n$ where IV is the initialization vector, $||$ is the concatenation of string. So A can ask the oracle $Enc_k(\cdot)$ with message $m' = IV \oplus m_1 \oplus (IV+1)$ and get the result $c' = (IV+1)||c'_1||c'_2||...||c'_n$.
If $c'_i = c_i$ for $i \in \{1, 2, ..., n\}$, then A outputs 1; otherwise, A outputs 2.
Since F_k in CBC-mode encryption is invertible, the construction of attacker A can successfully distinguish b. Therefore, the scheme is not CPA-secure.

9 Q9

Suppose attacker A outputs messages p_1, p_2 of the same length (for simplification , is 3-bit length). Then challenger chooses $b \leftarrow \{1, 2\}$ and encrypts $c \leftarrow Enc_k(m_b)$ and then gives $c = IV||c_1||c_2||c_3$ to A. So A knows that in the chained CBC mode, $m_i \in \{p_1^i, p_2^i\}$, where p_b^i means the i-th bit of p_b ($i \in \{1, 2, 3\}, b \in \{1, 2\}$).
Then attacker requests an encryption of a message p where $p^1 = IV \oplus p_1^1 \oplus c_3$, and observes the second ciphertext $c' = c_4||c_5$. A can verify that $m_1 = p_1^1$ if and only if $c_4 = c_1$, and so A learns m_1. It is the same for another bits. Therefore, the chained CBC mode is not as secure as CBC mode.