Homework 12

Problem 1. Let G be a pseudorandom generator of stretch ℓ such that $\ell(n) \geq 2n$.

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

(b) Define G'' as $G''(s) = G(s_1 \cdots s_{n/2})$ for $s = s_1 s_2 \cdots s_n$. Is G'' necessarily a pseudorandom generator?

Solution. (a)

G' is not necessarily a pseudorandom generator. Consider the following counterexample. Let G_0 be a pseudorandom generator of stretch $\ell(n) \geq 2n$. Define

$$G_1(s) = \begin{cases} 0^{\ell(n)}, & \text{if } \forall i \geq n/2, s_i = 0, \\ G_0(s), & \text{otherwise.} \end{cases}$$

Then, $G_1(s)$ and $G_0(s)$ are only different by a portion of $\frac{2^{n/2}}{2^n} = \frac{1}{2^{n/2}}$, which is negligible in n. So any polynomial-time distinguisher $\mathcal{A}$ can only distinguish G_1 and G_0 with negligible probability. Therefore, G_1 is also a pseudorandom generator.

However, if let G be G_1, then $G'(s) \equiv 0^{\ell(n)}$, which is not a pseudorandom generator.

(b)

G'' is necessarily a pseudorandom generator. Its stretch is

$$\ell''(n) = |G''(s)| = \left|G(s_1 \cdots s_{n/2})\right| = \ell(n/2) \geq n.$$

Since that G is a pseudorandom generator of stretch ℓ, we have

$$\begin{aligned}
&\left| \Pr_{s \in \{0,1\}^n} \Big(\mathcal{A}(G''(s)) = 1\Big) - \Pr_{r \in \{0,1\}^{\ell''(n)}} \Big(\mathcal{A}(r) = 1\Big) \right| \\
= &\left| \Pr_{s \in \{0,1\}^n} \Big(\mathcal{A}(G(s_1 s_2 \cdots s_{n/2})) = 1\Big) - \Pr_{r \in \{0,1\}^{\ell(n/2)}} \Big(\mathcal{A}(r) = 1\Big) \right| \\
= &\left| \Pr_{s'' \in \{0,1\}^{n/2}} \Big(\mathcal{A}(G(s'')) = 1\Big) - \Pr_{r \in \{0,1\}^{\ell(n/2)}} \Big(\mathcal{A}(r) = 1\Big) \right| \\
\leq &\ \mathrm{negl}(n).
\end{aligned}$$

By definition, G'' is a pseudorandom generator.

Problem 2. A keyed family of functions F_k is a pseudorandom random permutation (PRP) if (a) $F_k(\cdot)$ and $F_k^{-1}(\cdot)$ are efficiently computable given the key k and (b) for any polynomial-time algorithm $\mathcal{A}$,

$$\left|\Pr\left(\mathcal{A}^{F_k(\cdot),F_k^{-1}(\cdot)}(1^n) = 1\right) - \Pr\left(\mathcal{A}^{f_n(\cdot),f_n^{-1}(\cdot)}(1^n) = 1\right)\right| \le \mathrm{negl}(n).$$

Consider the following encryption scheme

1. Sample key k uniformly at random.
2. On input plaintext $x \in \{0,1\}^{n/2}$, algorithm Enc_k samples randomness $r \in \{0,1\}^{n/2}$ and outputs ciphertext $F_k(r\|x)$.

Solve the following problems assuming that F_k is a PRP.

(a) Show how the decryption Dec_k works.

(b) Prove that the encryption scheme is CPA-secure.

Solution. (a)

The decryption Dec_k works as follows:

1. On input ciphertext $y \in \{0,1\}^n$, compute $r\|x = F_k^{-1}(y)$.
2. Return x.

(b)

Proof by contradiction. Suppose the encryption scheme $\Pi = (\mathrm{Enc}, \mathrm{Dec})$ is not CPA-secure. Then there exists a polynomial-time adversary $\mathcal{A}_\Pi$ such that

$$\Pr(\mathcal{A}_\Pi \text{ succ}) \ge \frac{1}{2} + \frac{1}{\mathrm{poly}(n)}. \tag{1}$$

Consider the scheme $\widetilde{\Pi} = (\widetilde{\mathrm{Enc}}, \widetilde{\mathrm{Dec}})$ as the random permutation encryption scheme. Let r_c be the randomness used in the actual encryption, namely, $y = F_k(r_c\|x)$. Suppose $\mathcal{A}$ makes $q(n)$ queries to $\mathrm{Enc}_k(\cdot)$ using randomness $r_1, r_2, \cdots, r_{q(n)}$.

1. Case 1(Repeat). This is the case where $r_c \in \{r_1, r_2, \cdots, r_{q(n)}\}$. Then $\mathcal{A}_{\widetilde{\Pi}}$ can obtain r_c, so it can successfully distinguish $\widetilde{\Pi}$ from a random permutation. However, the chance of this case is at most $\frac{q(n)}{2^{n/2}}$.
2. Case 2(No Repeat). Then this is equivalent to the case where adversary doesn't know r_c, which is the same as a perfect OTP. Therefore,

$$\Pr(\mathcal{A}_{\widetilde{\Pi}} \text{ succ}|\text{no repeat}) = \frac{1}{2}.$$

Concluding the two cases above, we have

$$\begin{aligned}
\Pr\big(\mathcal{A}_{\widetilde{\Pi}} \text{ succ}\big) &= \Pr\big(\mathcal{A}_{\widetilde{\Pi}} \text{ succ} \wedge \text{ repeat}\big) + \Pr\big(\mathcal{A}_{\widetilde{\Pi}} \text{ succ } \wedge \text{ no repeat}\big) \\
&\le \Pr(\text{repeat}) + \Pr\big(\mathcal{A}_{\widetilde{\Pi}\text{ succ}} | \text{ no repeat}\big) \\
&\le \frac{1}{2} + \frac{q(n)}{2^n}. \qquad (2)
\end{aligned}$$

Combining (1) and (2), we have

$$\Big|\Pr\big(\mathcal{A}_{\Pi} \text{ succ}\big) - \Pr\big(\mathcal{A}_{\widetilde{\Pi}} \text{ succ}\big)\Big| \ge \frac{1}{\text{poly}(n)}. \qquad (3)$$

We will design a distinguisher $\mathcal{D}$ to show that (3) is impossible. Distinguisher $\mathcal{D}$ has oracle access to $\mathcal{O}$ and $\mathcal{O}^{-1}$, with input 1^n.

1. Run $\mathcal{A}(1^n)$ and whenever $\mathcal{A}$ queries $\text{Enc}_k(\cdot)$ with randomness r, answer the query in the following way:
   (a) Choose $r \in \{0,1\}^{n/2}$ uniformly at random.
   (b) Query $\mathcal{O}(r\|x)$ and obtain response s'.
   (c) Return s' to $\mathcal{A}$.
2. When $\mathcal{A}$ outputs x_0, x_1, choose $b \in \{0,1\}$ uniformly at random and then:
   (a) Choose $r \in \{0,1\}^{n/2}$ uniformly at random.
   (b) Query $\mathcal{O}(r\|x_b)$ and obtain response s'.
   (c) Return s' to $\mathcal{A}$.
3. Continue answering any queries of $\mathcal{A}$ as before. When $\mathcal{A}$ outputs b', output 1 if $b' = b$ and 0 otherwise.

Therefore, $\mathcal{D}$ simulates the experiment with Π and $\widetilde{\Pi}$. If $\mathcal{O} = F_k(\cdot)$, then the view of $\mathcal{A}$ is identical to the view of $\mathcal{A}$ in the experiment with Π. If $\mathcal{O} = f_n(\cdot)$, then the view of $\mathcal{A}$ is identical to the view of $\mathcal{A}$ in the experiment with $\widetilde{\Pi}$.

Hence,

$$\begin{aligned}
&\Big|\Pr\big(\mathcal{D}^{f_n(\cdot), f_n^{-1}(\cdot)}(1^n) = 1\big) - \Pr\big(\mathcal{D}^{F_k(\cdot), F_k^{-1}(\cdot)}(1^n) = 1\big)\Big| \\
=& \Big|\Pr\big(\mathcal{A}_{\widetilde{\Pi}} \text{ succ}\big) - \Pr\big(\mathcal{A}_{\Pi} \text{ succ}\big)\Big| \\
=& \frac{1}{\text{poly}(n)},
\end{aligned}$$

which contradicts to the assumption that F_k is a PRP. So the encryption scheme is CPA-secure.