Fundamentals of Cryptography Homework 3

周书予
2000013060@stu.pku.edu.cn

October 11, 2022

Problem 1

$\mathsf{Dec}(k,(r,c)) = F_k^{-1}(c) \oplus r$.

Denote $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ as the encryption scheme mentioned in the problem, and $\tilde{\Pi} = (\widetilde{\mathsf{Gen}}, \widetilde{\mathsf{Enc}}, \widetilde{\mathsf{Dec}})$ exactly the same as Π, except that a truly random permutation f is used in place of F_k.

The proof is divided into two parts:

- In the first part we prove that for any PPT adversary $\mathcal{A}$, there is some negligible function $\varepsilon(n)$ such that

$$\left|\Pr\left[\mathsf{PrivK}^{\mathsf{cpa}}_{\mathcal{A},\Pi}(n)=1\right] - \Pr\left[\mathsf{PrivK}^{\mathsf{cpa}}_{\mathcal{A},\tilde{\Pi}}(n)=1\right]\right| < \varepsilon(n) \tag{1}$$

- In the second part we show that for any PPT adversary $\mathcal{A}$,

$$\Pr\left[\mathsf{PrivK}^{\mathsf{cpa}}_{\mathcal{A},\tilde{\Pi}}(n)=1\right] \leqslant \frac{1}{2} + \frac{q(n)}{2^n} \tag{2}$$

  for some polynomial $q(n)$.

When finished the proof of the two parts mentioned above, one can see that obviously $\Pr\left[\mathsf{PrivK}^{\mathsf{cpa}}_{\mathcal{A},\Pi}(n)=1\right] \leqslant \frac{1}{2} + \frac{q(n)}{2^n} + \varepsilon(n)$, which means Π is secure under CPA attack.

Proof of eq. (1)

For any PPT adversary $\mathcal{A}$, a PPT distinguisher $\mathcal{D}$ can be built, which has access to an oracle $\mathcal{O}: \{0,1\}^n \to \{0,1\}^n$ (here it refers to F_k or f) and interacts with $\mathcal{A}$ like this:

1. when $\mathcal{A}$ queries the ciphertext for message $m \in \{0,1\}^n$, choose uniformly random $r \in \{0,1\}^n$ and return $(r, \mathcal{O}(r \oplus m))$.
2. when $\mathcal{A}$ outputs m_0 and m_1, choose a random bit $b \in \{0,1\}$ and uniformly random $r \in \{0,1\}^n$, then return $(r, \mathcal{O}(r \oplus m_b))$.
3. continue answering $\mathcal{A}$'s queries until $\mathcal{A}$ outputs a bit b', then output $\mathbb{1}[b = b']$.

It is easy to see that

$$\Pr\left[\mathsf{PrivK}^{\mathsf{cpa}}_{\mathcal{A},\Pi}(n)=1\right]=\Pr_{k\leftarrow\{0,1\}^n}\left[\mathcal{D}^{F_k}(1^n)=1\right]$$
$$\Pr\left[\mathsf{PrivK}^{\mathsf{cpa}}_{\mathcal{A},\tilde{\Pi}}(n)=1\right]=\Pr_{f\leftarrow\mathsf{Perm}_n}\left[\mathcal{D}^{f}(1^n)=1\right]$$

where Perm_n denotes the collection of all permutations over $\{0,1\}^n$.

Since F is a PRP, by definition we know that

$$\left|\Pr_{k\leftarrow\{0,1\}^n}\left[\mathcal{D}^{F_k}(1^n)=1\right]-\Pr_{f\leftarrow\mathsf{Perm}_n}\left[\mathcal{D}^{f}(1^n)=1\right]\right|<\varepsilon(n)$$

for some negligible $\varepsilon(n)$, so eq. (1) is proved as desired.

Proof of eq. (2)

Notice that $\mathcal{A}$ runs in polynomial time, so it can only queries the ciphertext for polynomially many m, say, $q(n)$. Whenever $\mathcal{A}$ queries m it obtains $f(r\oplus m)$ where r is known to $\mathcal{A}$ and chosen uniformly random. That is, each query gives $\mathcal{A}$ a pair $(x,f(x))$ which is a point value of f, where $x=r\oplus m$ is chosen uniformly random.

When $\mathcal{A}$ outputs m_0,m_1 and receives $(r^*,f(r^*\oplus m_b))$, it checks out all the recordings from the interaction, and if the point value for $r^*\oplus m_0$ or $r^*\oplus m_1$ is found, it can break the encryption scheme with 100% confidence, otherwise it learns nothing about $f(r^*\oplus m_0)$ and $f(r^*\oplus m_1)$, and probability of outputing the correct answer is exactly $1/2$.

The probability that the point value for $r^*\oplus m_0$ or $r^*\oplus m_1$ can be found equals to the probability of finding out two specific items among 2^n during $q(n)$ times of random choosing, which by union bound is not greater than $2q(n)/2^n$. Thus,

$$\Pr\left[\mathsf{PrivK}^{\mathsf{cpa}}_{\mathcal{A},\tilde{\Pi}}(n)=1\right]\leqslant\frac{2q(n)}{2^n}\cdot 1+\left(1-\frac{2q(n)}{2^n}\right)\cdot\frac{1}{2}=\frac{1}{2}+\frac{q(n)}{2^n}$$

Problem 2

Part A: F' is a PRF

First we show that for any PPT distinguisher $\mathcal{D}$,

$$\left|\Pr_{k\leftarrow\{0,1\}^n}\left[\mathcal{D}^{g\circ F_k}(1^n)\right]-\Pr_{f\leftarrow\mathsf{Func}_n}\left[\mathcal{D}^{g\circ f}(1^n)\right]\right|<\mathsf{negl}(n)\tag{3}$$

(here $f_1\circ f_2$ denotes the composition of function f_1 and f_2.)

This can be done by constructing another distinguisher $\mathcal{D}'$, which always queries the same message m as $\mathcal{D}$ does except that the oracle used here is F_k or f instead of $g\circ F_k$ or $g\circ f$, and outputs the same as $\mathcal{D}$ does.

It is easy to see that

$$\Pr_{k\leftarrow\{0,1\}^n}\left[\mathcal{D}^{g\circ F_k}(1^n)\right]=\Pr_{k\leftarrow\{0,1\}^n}\left[\mathcal{D}'^{F_k}(1^n)\right]$$
$$\Pr_{f\leftarrow\mathsf{Func}_n}\left[\mathcal{D}^{g\circ f}(1^n)\right]=\Pr_{f\leftarrow\mathsf{Func}_n}\left[\mathcal{D}'^{f}(1^n)\right]$$

Since F is a PRF, from its definition it is clear to see that eq. (3) can be proved.

Then we can show that for any PPT distinguisher $\mathcal{D}$,

$$\left|\Pr_{f\leftarrow\mathsf{Func}_n}\left[\mathcal{D}^{g\circ f}(1^n)\right]-\Pr_{h\leftarrow\mathsf{Func}_{n,2n}}\left[\mathcal{D}^{h}(1^n)\right]\right|<\mathsf{negl}(n) \tag{4}$$

(here $\mathsf{Func}_{n,2n}$ is defined as $\{h:\{0,1\}^n\to\{0,1\}^{2n}\}$.)

This can be done by using hybird argument: assume $\mathcal{D}$ interacts with oracle for $p(n)$ rounds and, WLOG, we assume $\mathcal{D}$ never queries for the same x for encryption (that is obviously suboptimal). Based on $\mathcal{D}$, distinguisher $\mathcal{D}'$ can be built, which on input $r\in\{0,1\}^{2n}$ works as follows:

- randomly sample t from $\{1,2,\cdots,p(n)\}$, and randomly fix some $f\leftarrow\mathsf{Func}_n$ and $h\leftarrow\mathsf{Func}_{n,2n}$. ($f$ and h do not need to be fully stored.)
- interact with $\mathcal{D}$. Whenever queried with x in round i, return $\begin{cases} g(f(x)), & i<t \\ r, & i=t \\ h(x), & i>t \end{cases}$.
- output the same as $\mathcal{D}$ does.

From this construction we know that

$$\begin{aligned}&\Pr_{s\leftarrow\{0,1\}^n}\left[\mathcal{D}'(g(s))=1\right]-\Pr_{r\leftarrow\{0,1\}^{2n}}\left[\mathcal{D}'(r)=1\right]\\&\qquad=\frac{1}{p(n)}\left(\Pr_{f\leftarrow\mathsf{Func}_n}\left[\mathcal{D}^{g\circ f}(1^n)\right]-\Pr_{h\leftarrow\mathsf{Func}_{n,2n}}\left[\mathcal{D}^{h}(1^n)\right]\right)\end{aligned}$$

Since g is a PRG, both sides of the equation are negligible, and eq. (4) is proved as desired.

From eq. (3) and eq. (4) one can draw that

$$\left|\Pr_{k\leftarrow\{0,1\}^n}\left[\mathcal{D}^{g\circ F_k}(1^n)\right]-\Pr_{h\leftarrow\mathsf{Func}_{n,2n}}\left[\mathcal{D}^{h}(1^n)\right]\right|<\mathsf{negl}(n) \tag{5}$$

which suggests that $F'_k=g\circ F_k$ is a PRF.

Part B: F' may not be a PRF

Let g be a PRG which drops its first bit of input. It is easy to see that such PRG exists.

Then for any $x\in\{0,1\}^{n-1}$, $F'_k(0\|x)=F_k(g(0\|x))=F_k(g(1\|x))=F'_k(1\|x)$, which suggests that F'_k is not that "random" and can be easily distinguished from a truly random function.

Problem 3

Part A: F' may not be a strong PRP

3-round Feistel is a PRP but not a strong one.

Part B: F' is a PRF

For any distinguisher $\mathcal{D}'$ which tries to distinguish F' from truly random functions, another distinguisher $\mathcal{D}$ can be built, which

- simulates $\mathcal{D}'$, when $\mathcal{D}'$ queries x, use its oracle and get $\mathcal{O}(x)$, and then return $x \oplus \mathcal{O}(x)$ to $\mathcal{D}'$.
- outputs whatever $\mathcal{D}'$ outputs.

It is easy to see that

$$\Pr_{k\leftarrow\{0,1\}^n}\left[\mathcal{D}^{F_k}(1^n)\right] = \Pr_{k\leftarrow\{0,1\}^n}\left[\mathcal{D}'^{F'_k}(1^n)\right]$$
$$\Pr_{f\leftarrow\mathsf{Func}_n}\left[\mathcal{D}^{f}(1^n)\right] = \Pr_{f\leftarrow\mathsf{Func}_n}\left[\mathcal{D}'^{x\oplus f}(1^n)\right]$$

Thus F' is a PRF.

Part C: F' is a PRP

First we can show that for any PPT distinguisher $\mathcal{D}$,

$$\left|\Pr_{k_1\|k_2\leftarrow g(\$)}\left[\mathcal{D}^{F_{k_2}\circ F_{k_1}}(1^n)=1\right] - \Pr_{k_1,k_2\leftarrow\$}\left[\mathcal{D}^{F_{k_2}\circ F_{k_1}}(1^n)=1\right]\right| < \mathsf{negl}(n)$$

This is because g is a PRG and no PPT distinguisher can distinguish $g(\$)$ from $\$\|\$$ with non-negligible advantage.

Then we can show that for any PPT distinguisher $\mathcal{D}$,

$$\left|\Pr_{k_1,k_2\leftarrow\$}\left[\mathcal{D}^{F_{k_2}\circ F_{k_1}}(1^n)=1\right] - \Pr_{f_1,f_2\leftarrow\mathsf{Perm}_n}\left[\mathcal{D}^{f_2\circ f_1}(1^n)=1\right]\right| < \mathsf{negl}(n)$$

This is because F itself is a PRP.

Together it has proved that $F'_k = F_{k_2} \circ F_{k_1}$ is a PRP.

Part D: F' is a PRP

First we show that for any PPT distinguisher $\mathcal{D}$,

$$\left|\Pr_{k\leftarrow\{0,1\}^n}\left[\mathcal{D}^{F_k\circ F_k}(1^n)=1\right] - \Pr_{f\leftarrow\mathsf{Perm}_n}\left[\mathcal{D}^{f\circ f}(1^n)=1\right]\right| < \mathsf{negl}(n)$$

This is because for any PPT distinguisher $\mathcal{D}$ which tries to distinguish $F_k \circ F_k$ from $f \circ f$, another distinguisher $\mathcal{D}'$ can be built, which queries the same as $\mathcal{D}$ does, uses its oracle twice to get $\mathcal{O}(\mathcal{O}(x))$, returns the result to $\mathcal{D}$ and finally outputs the same bit as $\mathcal{D}$. Thus

$$\Pr_{k\leftarrow\{0,1\}^n}\left[\mathcal{D}^{F_k\circ F_k}(1^n)\right] = \Pr_{k\leftarrow\{0,1\}^n}\left[\mathcal{D}'^{F_k}(1^n)\right]$$
$$\Pr_{f\leftarrow\mathsf{Perm}_n}\left[\mathcal{D}^{f\circ f}(1^n)\right] = \Pr_{f\leftarrow\mathsf{Perm}_n}\left[\mathcal{D}'^{f}(1^n)\right]$$

since F_k is a PRP, $\mathcal{D}$ can not have non-negligible advantage.

Then we are going to show that for any PPT distinguisher $\mathcal{D}$,

$$\left|\Pr_{f\leftarrow\mathsf{Perm}_n}\left[\mathcal{D}^{f\circ f}(1^n)=1\right]-\Pr_{f\leftarrow\mathsf{Perm}_n}\left[\mathcal{D}^{f}(1^n)=1\right]\right|<\mathsf{negl}(n)$$

WLOG we assume that $\mathcal{D}$ does not query the same x more than once. If all $x^{(i)}, \mathcal{O}(x^{(i)})$ are distinct, we can assert that no distinguisher can figure out the correct answer with probability more than $1/2$, and the thing only fails with negligible probability, so the inequlity above holds for any PPT distinguisher.

Combining these two statements we can show that $F'_k = F_k \circ F_k$ is a PRP.

Problem 4

Part A: $F(k, x \oplus c) = F(k, x) \oplus c$

A distinguisher $\mathcal{D}$ simply queries the encryption for 0^n and 1^n and checks whether $\mathcal{O}(0^n) \oplus \mathcal{O}(1^n) = 1^n$, then outputs 1 if the equation holds and 0 otherwise.

It is easy to see that

$$\Pr_{k\leftarrow\{0,1\}^n}\left[\mathcal{D}^{F_k}(1^n)=1\right]=1$$
$$\Pr_{f\leftarrow\mathsf{Func}_n}\left[\mathcal{D}^{f}(1^n)=1\right]=1/2^n$$

so $\mathcal{D}$ breaks F_k as PRF.

Part B: $F(k \oplus c, x) = F(k, x) \oplus c$

A distinguisher $\mathcal{D}$ can be built, which

- calculates $F(0^n, 0^n)$.
- queries the encryption for 0^n, say, $\mathcal{O}(0^n)$.
- learns the "key" $k' = F(0^n, 0^n) \oplus \mathcal{O}(0^n)$.
- checks the "key" is correct or not, that is, randomly choose $x \in \{0,1\}^n$ and checks whether $F(k', x) = \mathcal{O}(x)$, then outputs 1 is key is correct and 0 otherwise.

It is easy to see that

$$\Pr_{k\leftarrow\{0,1\}^n}\left[\mathcal{D}^{F_k}(1^n)=1\right]=1$$
$$\Pr_{f\leftarrow\mathsf{Func}_n}\left[\mathcal{D}^{f}(1^n)=1\right]=1/2^n$$

so $\mathcal{D}$ breaks F_k as PRF.

Part C: $F(k_1 \oplus k_2, x) = F(k_1, x) \oplus F(k_2, x)$

Let $\varepsilon_i = 0^{i-1}10^{n-i}$ be the i-th "unit vector". For any $k, x \in \{0,1\}^n$, if k has 1 on bit $i_1, i_2, \cdots, i_l$, then we have $F(k, x) = \bigoplus_{j=1}^{l} F(\varepsilon_{i_j}, x)$.

A distinguisher $\mathcal{D}$ first calculates $\alpha_i = F(\varepsilon_i, 0^n)$ and then checks whether the linear span $\mathsf{span}\left(\{\alpha_i\}_{i=1}^n\right)$ equals to the whole linear space $\{0,1\}^n$.

- If the answer is yes, then for any $y \in \{0,1\}^n$ there exists exactly one set $S \subseteq \{1, \cdots, n\}$ such that $y = \bigoplus_{i \in S} \alpha_i$. This means that the distinguisher $\mathcal{D}$ can learn the key by finding the set S such that $\mathcal{O}(0^n) = \bigoplus_{i \in S} \alpha_i$ can let key be $k = \bigoplus_{i \in S} \varepsilon_i$.

  After learning the key, $\mathcal{D}$ takes a single test just like the one in **Part B** does, and then outputs the testing result.

  In this case we can show that

  $$\Pr_{k \leftarrow \{0,1\}^n}\left[\mathcal{D}^{F_k}(1^n) = 1\right] = 1$$
  $$\Pr_{f \leftarrow \mathsf{Func}_n}\left[\mathcal{D}^{f}(1^n) = 1\right] = 1/2^n$$

  so $\mathcal{D}$ breaks F_k as PRF.

- If the answer is no, then for at least half of $y \in \{0,1\}^n$, it can not be represented as the linear combination of α_i. Then $\mathcal{D}$ simply checks whether $\mathcal{O}(0^n)$ can be represented and ouputs the result.

  In this case we can show that

  $$\Pr_{k \leftarrow \{0,1\}^n}\left[\mathcal{D}^{F_k}(1^n) = 1\right] = 1$$
  $$\Pr_{f \leftarrow \mathsf{Func}_n}\left[\mathcal{D}^{f}(1^n) = 1\right] \leqslant 1/2$$

  so $\mathcal{D}$ also breaks F_k as PRF.

(All linear algebra calculation can be done in polynomial time via algorithms like Gauss-Jordan Elimination, so $\mathcal{D}$ is indeed a PPT distinguisher.)

Problem 5

Assume there is a length-doubling PRG $G : \{0,1\}^n \to \{0,1\}^{2n}$, and we use the notation $G_0(s)$ and $G_1(s)$ to describe the former and latter half of $G(s)$ respectively, that is

$$G(s) = G_0(s) \| G_1(s)$$

- F is constructed as

  $$F(k, x) = G_{x_1} G_{x_2} \cdots G_{x_n}(k)$$

  (this means the composition of n $G_{0/1}$ functions, the notation $\circ$ is omitted for simplicity.) it is easy to see that F runs in polynomial time.

- The puncture function is constructed as

  $$\mathsf{puncture}(k, x) = \left\{ \left(\overline{x_i} \| x_{i+1 \cdots n}, G_{\overline{x_i}} G_{x_{i+1}} \cdots G_{x_n}(k) \right) \middle| i = 1, 2, \cdots, n \right\}$$

- The eval function $\mathsf{eval}(k_{-x}, x')$ first finds one satisfying that str is a suffix of x' among all (str, key) pairs in k_{-x} (we assert that there exsits and only exsits one such pair since $x' \neq x$), and than calculates $\mathsf{eval}(k_{-x}, x') = G_{x'_1} G_{x'_2} \cdots G_{x'_{n-|str|}}(key)$.

  It is not hard to see that $\mathsf{eval}(k_{-x}, x') = F_k(x')$ for all $x' \neq x$.

  We need to further prove that

- F is a PRF. By hybird argument, consider

$$F_{h_i}(x) = G_{x_1} G_{x_2} \cdots G_{x_i}(h_i(x_{i+1} \cdots x_n))$$

  where $h_i : \{0,1\}^{n-i} \to \{0,1\}^n$ is a truly random function. Clearly F_{h_0} is actually a truly random function over $\{0,1\}^n$, and F_{h_n} is equivalent to F.

  Since G is a PRG, no PPT distinguisher can distinguish F_{h_i} from $F_{h_{i+1}}$ with non-negligible advantage, so F_{h_0} can not be distinguished from F_{h_n} with non-negligible advantage, which means that F is a PRF.

- F is puncturable. Here hybird argument is used again, considering such parameterized security game:

  - With parameter i, the distinguisher $\mathcal{D}$ chooses x, the challenger samples random k, u, computes k_{-x}, and sends $\begin{cases} (k_{-x}, G_{x_1} \cdots G_{x_i}(u)), & i < n \\ (k_{-x}, G_{x_1} \cdots G_{x_n}(k)), & i = n \end{cases}$ to the distinguisher.

  Obviously the two cases in the problem are equivalent to game $i = n$ and $i = 0$ respectively.

  Since G is a PRG, no PPT distinguisher can distinguish security games with adjacent parameter i, so the two cases are also indistinguishable, with non-negligible advantage.

Problem 6

Part A

The distinguisher $\mathcal{D}$ can implement the following steps to break 3-round Feistel:

- Query $\mathsf{Dec}(0^n, 0^n) \to (a, b)$.
- Query $\mathsf{Enc}(0^n, b) \to (c, d)$.
- Query $\mathsf{Dec}(c, a \oplus d) \to (e, f)$.
- Output 1 is $f = c \oplus b$ and 0 otherwise.

Here's why it works: under 3-round Feistel encryption scheme, the first query reveals

$$a = F_{k_3}(0^n) \oplus F_{k_1}(b)$$
$$b = F_{k_2}(F_{k_3}(0^n))$$

and the second query

$$c = b \oplus F_{k_2}(F_{k_1}(b))$$
$$d = F_{k_1}(b) \oplus F_{k_3}(c)$$

and so

$$a \oplus d = F_{k_3}(0^n) \oplus F_{k_3}(c)$$

In the final query, the distinguisher knows that

$$f = c \oplus F_{k_2}(F_{k_3}(0^n)) = c \oplus b$$

.

So distinguisher $\mathcal{D}$ outputs 1 with probability 1 under 3-round Feistel, but with probability $1/2^n$ under truly random permutations, which means that 3-round Feistel is not a strong PRP.

Part B

这咋做啊