

Homework 2

Yanbin LU

(+ some edits by Stanislaw Jarecki)

Information and Computer Sciences

University of California, Irvine

yanbinl@uci.edu

November 16, 2007

1 Problem 1

$$\begin{aligned} \log 2^{n^{1/3}(\log n)^{2/3}} &\leq \log(4 * 10^9 * 100 * 365 * 24 * 3600) \\ n^{1/3}(\log n)^{2/3} &\leq 63.4517 \end{aligned}$$

Since $n^{1/3}(\log n)^{2/3}$ is monotonically increasing and, when $n = 2098$, it is 63.4519, $n \leq 2098$.¹

2 Problem 2

Let Gen, Enc, Dec be a secure private-key encryption scheme in the sense of definition 3.8. Define a private-key encryption scheme Gen', Enc', Dec' that'll be secure for unequal-length challenge messages with length up to $l(n)$, as follows:

- Gen' is just Gen .
- Enc' on input a key $k \in \{0, 1\}^n$ and a message $m \in \{0, 1\}^h$ for some $h \leq l(n)$, first pad the message m with 1 and then pad 0s until its length reaches $l(n) + 1$ and then encrypt :

$$c := Enc_k([m|10^{l(n)-|m|}])$$

- Dec' on input a key $k \in \{0, 1\}^n$ and a ciphertext c

$$m' := Dec_k(c)$$

then remove the zeros and the first 1 at the end of the message m' and output the new message m .

¹**SJ:** The exact answer depended on whether you interpret $\log n$ as a base 10, e , or 2 logarithm.

Adversary \mathcal{A} cannot tell the difference between padded messages m'_0 and m'_1 given c according to construction 3.15. So neither can it tell the difference between original messages m_0 and m_1 .²

3 Problem 3, Part 1

Proof Let $\tilde{\Pi} = (\widetilde{Gen}, \widetilde{Enc}, \widetilde{Dec})$ be an encryption scheme that is exactly the same as $\Pi = (Gen, Enc, Dec)$ in Problem 3, except that a truly random string r is used. We claim that for any PPT adversary \mathcal{A} within limit of messages it can see encrypted, we have

$$Pr \left[PrivK_{\mathcal{A}, \tilde{\Pi}}^{cpa}(n) = 1 \right] = \frac{1}{2}$$

This is because $\widetilde{Enc}_k(m, p) = m \oplus [r]_p^{p+|m|-1}$ where $[r]_p^{p+|m|-1}$ is truly random and the part of r used before is never used to encrypt to next message.

Define ϵ as:

$$\epsilon(n) \triangleq Pr[PrivK_{\mathcal{A}, \Pi}^{cpa}(n) = 1] - \frac{1}{2}.$$

Let's define a **Distinguisher** D as follows:

D is given as input a string $w \in \{0, 1\}^{l(n)}$.

1. Run $\mathcal{A}(1^n)$. Whenever \mathcal{A} queries its encryption oracle on a message m , answer this query in the following way: return the ciphertext $m \oplus [w]_p^{p+|m|-1}$ to \mathcal{A} and update $p \leftarrow p + |m|$.
2. When \mathcal{A} outputs messages $m_0, m_1 \in \{0, 1\}^n$, choose a random bit $b \leftarrow \{0, 1\}$ and then return $m_b \oplus [w]_p^{p+|m_b|-1}$ to \mathcal{A} and update $p \leftarrow p + |m|$.
3. Continue answering any encryption oracle queries of \mathcal{A} as before. Eventually, \mathcal{A} outputs a bit b' . Output 1 if $b' = b$, and output 0 otherwise.

The key points are as follows:

1. If w is chosen uniformly at random from $\{0, 1\}^{l(n)}$, then the view of \mathcal{A} when run as a sub-routine of D is distributed identically to the view of \mathcal{A} in experiment $PrivK_{\mathcal{A}, \tilde{\Pi}}^{cpa}(n) = 1$. Therefore:

$$Pr[D(w) = 1] = Pr[PrivK_{\mathcal{A}, \tilde{\Pi}}^{cpa}(n) = 1] = \frac{1}{2},$$

2. If w is equal to $G(k)$ for $k \leftarrow \{0, 1\}^n$ chosen uniformly at random, then the view of \mathcal{A} when run as a sub-routine by D is distributed identically to the view of \mathcal{A} in experiment $PrivK_{\mathcal{A}, \Pi}^{cpa}(n) = 1$. Therefore:

$$Pr[D(G(k)) = 1] = Pr[PrivK_{\mathcal{A}, \Pi}^{cpa}(n) = 1] = \frac{1}{2} + \epsilon(n)$$

²**SJ:** One could give a formal reduction but it's not necessary in this case: If you have encryption that is secure on same-length challenge messages, the security of this construction follows.

Therefore,

$$|Pr[D(w) = 1] - Pr[D(G(k)) = 1]| = \epsilon(n)$$

By the definition of pseudorandom generator, it must be the case that ϵ is negligible, and so Π is CPA secure.

4 Problem 3, Part 2

Given ciphertext c which is encrypted by $Enc_k(p^e, m)$, where m maybe m_0 or m_1 generated by adversary \mathcal{A} , \mathcal{A} composes another ciphertext c' with the same length with c . Then \mathcal{A} accesses $Dec_k(c', p^d)$ to get the plaintext m' . Assume that $p^e = p^d$ and use p for simplicity. Note that $c' \oplus m' = [G(k)]_p^{p+|m|-1}$ and also $c \oplus m_b = [G(k)]_p^{p+|m|-1}$. Therefore \mathcal{A} can output 0 if $c' \oplus m' = c \oplus m_0$ and 1 otherwise.

5 Problem 4

Define $G_c(k)$ to be

$$G_c(k) = [F_k(c+1)|F_k(c+2)|\cdots|F_k(c+p(n))];$$

where c is a constant.³

Assume G_c is not secure PRG, then there exists PPT Adversary \mathcal{A} , such that

$$|Pr[\mathcal{A}(r) = 1] - Pr[\mathcal{A}(G(s)) = 1]| > \text{negl}(n)$$

where r is chosen uniformly at random from $\{0, 1\}^{l(n)}$, the seed s is chosen uniformly at random from $\{0, 1\}^n$

Then we claim there exists an adversary \mathcal{A}' such that it can tell the difference between a truly random function $f(\cdot)$ and $F_k(\cdot)$. we construct adversary \mathcal{A}' as follows: given 1^n and access to an oracle $O : \{0, 1\}^n \rightarrow \{0, 1\}^n$.

1. Query the oracle $O(c+1), O(c+2), \dots, O(c+p(n))$, and gets $y_1, y_2, \dots, y_{c+p(n)}$.
2. compose $y = y_1|y_2|\cdots|y_{c+p(n)}$ and input y to \mathcal{A}
3. if \mathcal{A} outputs 0 indicating y is a random string, \mathcal{A}' also outputs 0 indicating oracle O is truly random function. Otherwise, \mathcal{A}' outputs 1 indicating oracle O is pseudorandom function.

By above construction, we have

$$|Pr[\mathcal{A}'^{F_k(\cdot)}(1^n) = 1] - Pr[\mathcal{A}'^{f(\cdot)}(1^n) = 1]| > \text{negl}(n)$$

which contradicts the definition of PRF.

³**SJ:** Some students made a mistake treating c as random. But PRG has to be deterministic so it's not clear where this randomness can come from. It's also not needed. Another good construction was from the OFB mode, i.e. $G_c(k) = [F_k(c) | F_k(F_k(c)) | F_k(F_k(F_k(c))) | \cdots]$. The proof for this one is very similar because if F_k was a true random function then this would be a random string.

6 Problem 5

Proof Let $\tilde{\Pi} = (\widetilde{Gen}, \widetilde{Enc}, \widetilde{Dec})$ be an encryption scheme that is exactly the same as $\Pi = (Gen, Enc, Dec)$ in Problem 5, except that a truly random function f is used in place of F_k . We claim that for any adversary \mathcal{A} that makes at most $q(n)$ queries to its encryption oracle, we have

$$Pr \left[PrivK_{\mathcal{A}, \tilde{\Pi}}^{cpa}(n) = 1 \right] \leq \frac{1}{2} + \frac{q(n)}{2^{n/2}}$$

This is because everytime a message is encrypted, a random $r \leftarrow \{0, 1\}^{n/2}$ is chosen and ciphertext is set equal to $f(r|m)$. Let r_c denote the random string used when generating the challenge ciphertext $c = f(r_c|m_b)$. There are two subcases:

1. The value r_c is used by encryption oracle to answer at least one of \mathcal{A} 's queries of message m_b : In this case, \mathcal{A} can determine which message was encrypted by comparing the previous ciphertext and the challenge. The probability of this event is at most $q(n)/2^{n/2}$.
2. The value r_c is never used by the encryption oracle to answer any of \mathcal{A} 's queries: In this case, $f(r_c|m_b)$ is completely random as far as \mathcal{A} is concerned. So the probability that \mathcal{A} outputs $b' = b$ in this case is exactly $1/2$.

Thus we have,

$$\begin{aligned} Pr \left[PrivK_{\mathcal{A}, \tilde{\Pi}}^{cpa}(n) = 1 \right] &= Pr[PrivK_{\mathcal{A}, \tilde{\Pi}}^{cpa}(n) = 1 \wedge Repeat] + Pr[PrivK_{\mathcal{A}, \tilde{\Pi}}^{cpa}(n) = 1 \wedge \overline{Repeat}] \\ &\leq Pr[Repeat] + Pr[PrivK_{\mathcal{A}, \tilde{\Pi}}^{cpa}(n) = 1 | \overline{Repeat}] \\ &\leq \frac{1}{2} + \frac{q(n)}{2^{n/2}} \end{aligned}$$

Next fix some PPT adversary \mathcal{A} and define the function ϵ by

$$\epsilon(n) \triangleq Pr \left[PrivK_{\mathcal{A}, \Pi}^{cpa}(n) = 1 \right] - \frac{1}{2}$$

Construct the distinguisher D as follows: D is given input 1^n and access to an oracle $O : \{0, 1\}^n \rightarrow \{0, 1\}^n$.

1. Run $\mathcal{A}(1^n)$. Whenever \mathcal{A} queries its encryption oracle on a message m , answer this query in the following way:
 - (a) Choose $r \leftarrow \{0, 1\}^{n/2}$ uniformly at random.
 - (b) Query $O(r|m)$ and return the response to \mathcal{A}
2. When \mathcal{A} outputs messages $m_0, m_1 \in \{0, 1\}^n$, choose a random bit $b \leftarrow \{0, 1\}$ and then:

- (a) Choose $r \leftarrow \{0, 1\}^{n/2}$ at random.
 - (b) Query $O(r|m_b)$ and return the response as challenge to \mathcal{A}
3. Continue answering any encryption oracle queries of \mathcal{A} as before. Eventually, \mathcal{A} outputs a bit b' . Output 1 if $b' = b$, and output 0 otherwise.

The key points are as follows:

1. If D 's oracle is a pseudorandom function, then the view of \mathcal{A} when run as a sub-routine by D is distributed identically to the view of \mathcal{A} in experiment $\text{PrivK}_{\mathcal{A}, \Pi}^{\text{cpa}}(n)$. Thus,

$$\Pr[D^{F_k(\cdot)}(1^n) = 1] = \Pr[\text{PrivK}_{\mathcal{A}, \Pi}^{\text{cpa}}(n)]$$

where $k \leftarrow \{0, 1\}^n$ is chosen uniformly at random in the above.

2. If D 's oracle is a random function, then the view of \mathcal{A} when run as a sub-routine by D is distributed identically to the view of \mathcal{A} in experiment $\text{PrivK}_{\mathcal{A}, \Pi}^{\text{cpa}}(n)$. Thus,

$$\Pr[D^{f(\cdot)}(1^n) = 1] = \Pr[\text{PrivK}_{\mathcal{A}, \Pi}^{\text{cpa}}(n)]$$

where f is chosen uniformly at random in the above.

Therefore we have

$$\Pr[D^{F_k(\cdot)}(1^n) = 1] - \Pr[D^{f(\cdot)}(1^n) = 1] \geq \epsilon(n) - \frac{q(n)}{2^{n/2}}.$$

By the assumption that F is a pseudorandom function, it follows that $\epsilon(n) - q(n)/2^n$ must be negligible. Therefore $\epsilon(n)$ is negligible. So scheme Π is CPA secure.

7 Problem 6

(**SJ**: For example query the encryption on $m = 0^n$ when IV is even. Then the ciphertext you get is $(IV, F_k(IV))$. The new IV is $IV' = IV + 1$ but since IV was even we have $IV' = IV \oplus 0^{n-1}1$. Now send as encryption challenge (m_0, m_1) s.t. $m_0 = 0^{n-1}1$ and $m_1 \neq m_0$. If $b = 0$ then $c = (IV', F_k(IV' \oplus 0^{n-1}1)) = (IV', F_k(IV))$, and since the adversary knows $F_k(IV)$, it can tell if $b = 0$ or not.)

8 Problem 7

8.1 CBC

Just focus on the first two ciphertext blocks. An adversary running in the CCA indistinguishability experiment can choose $m_0 = \{0^n, 0^n\}$, $m_1 = \{1^n, 1^n\}$. Upon receiving ciphertext $c = \{c_1, c_2\}$, it flips the first bit of c_1 and asks for a decryption of the resulting ciphertext c' . The decryption oracle answers with $m' = \{m'_0, m'_1\}$. If $m'_1 = 10^{n-1}$, then $b = 0$. If $m'_1 = 01^{n-1}$, then $b = 1$.

8.2 OFB

Just focus on the first ciphertext block. An adversary running in the CCA indistinguishability experiment can choose $m_0 = 0^n$ and $m_1 = 1^n$. Upon receiving a ciphertext $c = (IV, s)$, adversary can flip the first bit of s and ask for a decryption of the resulting ciphertext c' . Since $c' \neq c$, this query is allowed and the decryption oracle answers with either 10^{n-1} (in which case $b = 0$) or 01^{n-1} (in which case $b = 1$).

8.3 CTR

The adversary is constructed the same as OFB.