Data Privacy Homework 3

傅申　PB20000051

1. (10′) Permutation Cipher

(a) (5′) Consider the permutation π on the set $1, 2, ..., 8$ defined as follows. Find the inverse permutation π^{-1}.

x	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
$\pi(x)$	**4**	**1**	**6**	**2**	**7**	**3**	**8**	**5**

(b) (5′) Decrypt the following ciphertext encrypted using a permutation cipher with the key being the permutation π from part (a).

TGEEMNELNNTDROEOAAHDOETCSHAEIRLM

(a) The inverse permutation is shown below.

x	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
$\pi^{-1}(x)$	**2**	**4**	**6**	**1**	**8**	**3**	**5**	**7**

(b) If **Columnar Transposition** is used to encrypt the plaintext, the plaintext should be written in rows of length 8 and then the columns are permuted according to the permutation π from part (a). The ciphertext is then obtained by reading the columns in order. The grid below shows the process of encryption.

x	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
$\pi(x)$	**4**	**1**	**6**	**2**	**7**	**3**	**8**	**5**
Plaintext	R	T	O	M	S	N	I	A
	O	G	E	N	H	N	R	A
	E	E	T	E	A	T	L	H
	O	E	C	L	E	D	M	D

Thus, the plaintext is

RTOMSNIAOGENHNRAEETEATLHOECLEDMD.

Otherwise, if we **permutate the plaintext block by block**, the plaintext should be divided into blocks of length 8 and then permuted according to the permutation π from part (a). The ciphertext is then obtained by reading the blocks in order. To decrypt the ciphertext, first divide the ciphertext into blocks of length 8: *TGEEMNEL NNTDROEO AAHDOETC SHAEIRLM*, and then permute each block: *ETNGEELM DNONETOR DAEATHCO ESRHLAMI*. So the plaintext is

ETNGEELMDNONETORDAEATHCOESRHLAMI.

2. (20′) Perfect Secrecy

(a) (10′) Let n be a positive integer. An n-th order Latin square is an $n \times n$ matrix L such that each of the n integers $1, 2, ..., n$ appears exactly once in each row and each column of L. The pollowing is an example of a Latin square of order 3:

$$\begin{matrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 2 & 3 & 1 \end{matrix}$$

For any n-th order Latin square L, we can define a related encryption scheme. Let $\mathscr{M} = \mathscr{C} = \mathscr{K} = \{1, 2, ..., n\}$. For $1 \le i \le n$, the encryption rule e_i is defined as $e_i(j) = L(i, j)$ (thus, each row provides an encryption rule). Prove that if the key is chosen uniformly at random, the Latin square cipher has perfect secrecy.

(b) (10′) Prove that if a cipher has perfect secrecy and $|\mathscr{M}| = |\mathscr{C}| = |\mathscr{K}|$, then each ciphertext is equiprobable.

(a) Accroding to Shannon's theorem, the Latin square cipher has perfect secrecy because

(1) Each key is chosen with equal probability.

(2) Knowing j, there is only one key that encrypt j to a $L(i, j)$, because each number appears only once on a row.

(b) Since the cipher has perfect secrecy, each key is chosen with equal probability $1/|\mathscr{K}|$. For every $m \in \mathscr{M}$ and $c \in \mathscr{C}$, there is a unique $k \in \mathscr{K}$ such that $e_k(m) = c$. Thus, the probability of c is

$$\Pr[c] = \sum_{m \in \mathscr{M}} \Pr[m] \Pr[c \mid m] = \sum_{m \in \mathscr{M}} \frac{1}{|\mathscr{M}|} \Pr[e_k(m) = c \mid m] = \sum_{m \in \mathscr{M}} \frac{1}{|\mathscr{M}|} \frac{1}{|\mathscr{K}|} = \frac{1}{|\mathscr{K}|} = \frac{1}{|\mathscr{C}|}.$$

So each ciphertext is equiprobable.

3. (25′) RSA

Assuming that Bob uses RSA and selects two *large* prime numbers $p = 101$ and $q = 113$:

(a) (5′) How many possible public keys Bob can choose?

(b) (10′) Assuming that Bob uses a public encryption key $e = 3533$. Alice sends Bob a message $M = 9726$. What will be the ciphertext received by Bob? Show the detailed procedure that Bob decrypts the received ciphertext.

(c) (10′) Let $n = pq$ be a product of two distinct primes. Show that if $\phi(n)$ and n are known, then it is possible to compute p and q in polynomial time. *Hint: Derive a quadratic equation (over the integers) in the unknown p.*

$$n = pq = 11413, \phi(n) = (p - 1)(q - 1) = 11200$$

(a) The possible public keys are the integers in $(1, 11200)$ that are coprime to 11200, so there are $\phi(11200)$ such integers. Since $11200 = 2^6 \times 5^2 \times 7$, there are

$$\phi(11200) = 11200\left(1 - \frac{1}{2}\right)\left(1 - \frac{1}{5}\right)\left(1 - \frac{1}{7}\right) = 3840$$

possible public keys.

(b) The private key d should satisfy $d \equiv e^{-1} (\mathrm{mod}\, \phi(n))$. Since $e = 3533$ and $\phi(n) = 11200$, it can be computed that $d = 6597$.

Given the plaintext $m = 9726$, the ciphertext Bob received is

$$c = m^e \bmod n = 9726^{3533} \bmod 11413 = 5761.$$

To decrypt the ciphertext, Bob computes

$$m = c^d \bmod n = 5761^{6597} \bmod 11413 = 9726,$$

which is the plaintext.

(c) Since $n = pq$ is a product of two distinct primes, $\phi(n) = (p-1)(q-1) = n - p - q + 1$. Thus, we have

$$\begin{cases} pq &= n \\ p + q = n + 1 - \phi(n) \end{cases}'$$

Accroding to Vieta's formulas, p and q are the roots of equation $x^2 - (n + 1 - \phi(n))x + n = 0$. So p and q can be computed by solving the equation, which can be done in polynomial time:

$$p, q = \frac{(n + 1 - \phi(n)) \pm \sqrt{(n + 1 - \phi(n))^2 - 4n}}{2}$$

4. (20′) Multi-Party Computation

(a) (10′) Paillier Encryption. Assuming Alice employs the Paillier encryption scheme with the prime numbers $p = 11$ and $q = 17$, along with a randomly chosen value of $r = 83$ and $g = n + 1$. Alice transmits a message $M = 175$ to Bob. What ciphertext will Bob receive? Additionally, please prove the Homomorphic addition property of Paillier:

$$\text{Decrypt}\big((c_1 \cdot c_2) \bmod n^2\big) = m_1 + m_2$$

(b) (10′) Secret Sharing. We define a 2-out-of-3 secret sharing scheme as follows. In order to share a bit v, the dealer chooses three random bits $x_1, x_2, x_3 \in \{0, 1\}$ under the constraint that $x_1 \oplus x_2 \oplus x_3 = 0$. Then:

- P_1's share is the pair (x_1, a_1) where $a_1 = x_3 \oplus v$.
- P_2's share is the pair (x_2, a_2) where $a_2 = x_1 \oplus v$.
- P_3's share is the pair (x_3, a_3) where $a_3 = x_2 \oplus v$.

Let $(x_1, a_1), (x_2, a_2), (x_3, a_3)$ be a secret sharing of v_1, and let $(y_1, b_1), (y_2, b_2), (y_3, b_3)$ be a secret sharing of v_2. Try to explain that no communication is needed in order to compute a secret sharing of $v_1 \oplus v_2$. ($\oplus$ means XOR.)

(a) *(Encryption)* First, run the key generation procedure as follows:

(1) Pick $p = 11$ and $q = 17$.
(2) Compute $n = 11 \times 17 = 187$.
(3) Compute $\lambda = \text{lcm}(p - 1, q - 1) = \text{lcm}(10, 16) = 80$.
(4) Pick $g = n + 1 = 188$ is picked.
(5) Compute $\mu = \Big(L\big(g^\lambda \bmod n^2\big)\Big)^{-1} \bmod n = 180$

Thus, the public key is $(n, g) = (187, 188)$ and the private key is $(\lambda, \mu) = (80, 180)$. Given the plaintext $M = 175$, the ciphertext Bob received is

$$C = g^M r^n \bmod n^2 = \big(188^{175} \times 83^{187}\big) \bmod 187^2 = 23911.$$

(Proof of Homomorphic addition property) For two arbitrary plaintext m_1, m_2, the ciphertexts are

$$c_1 = g_1^m r_1^n \bmod n^2, c_2 = g_2^m r_2^n \bmod n^2 \Rightarrow c_1 \cdot c_2 = g^{m_1 + m_2}(r_1 r_2)^n \bmod n^2.$$

Thus, the product is decrypted as

$$\begin{aligned}\text{Decrypt}\big(c_1 \cdot c_2 \bmod n^2\big) &= \text{Decrypt}\big(g^{m_1+m_2}(r_1 r_2)^n \bmod n^2\big)\\ &= \text{Decrypt}\big(g^{m_1+m_2} r_*^n \bmod n^2\big)\\ &= m_1 + m_2.\end{aligned}$$

So the Homomorphic addition property of Paillier holds.

(b) For each P_i, simply compute $(x_i \oplus y_i, a_i \oplus b_i)$ would form a secret sharing of $v_1 \oplus v_2$. It does not require any communication. The result secret sharing is shown as follows:

Player	Computed Secret Sharing
P_1	$(x_1 \oplus y_1, (x_3 \oplus y_3) \oplus (v_1 \oplus v_2))$
P_2	$(x_2 \oplus y_2, (x_1 \oplus y_1) \oplus (v_1 \oplus v_2))$
P_3	$(x_3 \oplus y_3, (x_2 \oplus y_2) \oplus (v_1 \oplus v_2))$

Since $(x_3 \oplus y_3) \oplus (x_1 \oplus y_1) \oplus (x_2 \oplus y_2) = (x_1 \oplus x_2 \oplus x_3) \oplus (y_1 \oplus y_2 \oplus y_3) = 0 \oplus 0 = 0$, the computed secret sharing is valid.

5. (25′) Computational Security

(a) (5′) Explain the difference between *Interchangeable* and *Indistinguishable*.

(b) (10′) Which of the following are negligible functions in λ? Justify your answers.

$$\frac{1}{2^{\lambda/2}} \quad \frac{1}{2^{\log(\lambda^2)}} \quad \frac{1}{\lambda^{\log\lambda}} \quad \frac{1}{\lambda^2} \quad \frac{1}{2^{(\log\lambda)^2}} \quad \frac{1}{(\log\lambda)^2} \quad \frac{1}{\lambda^{1/\lambda}} \quad \frac{1}{\sqrt{\lambda}} \quad \frac{1}{2^{\sqrt{\lambda}}}$$

(c) (10′) Suppose f and g are negligible.

(1) Show that $f + g$ is negligible.

(2) Show that $f \times g$ is negligible.

(3) Give an example f and g which are both negligible, but where $f(\lambda)/g(\lambda)$ is not negligible.

(a) The definition of these two terms are

Interchangeable $\mathscr{L}_1$ and $\mathscr{L}_2$ are interchangeable if for all programs $\mathscr{A}$ that output a single bit, $\Pr[\mathscr{A} \diamond \mathscr{L}_1 \Rightarrow 1] = \Pr[\mathscr{A} \diamond \mathscr{L}_2 \Rightarrow 1]$.

Indistinguishable $\mathscr{L}_1$ and $\mathscr{L}_2$ are indistinguishable if for all polynomial-time programs $\mathscr{A}$ that output a single bit, $\Pr[\mathscr{A} \diamond \mathscr{L}_1 \Rightarrow 1] - \Pr[\mathscr{A} \diamond \mathscr{L}_2 \Rightarrow 1]$ is negligible.

So the difference is that, interchangeable is a stronger condition than indistinguishable. There is no program $\mathscr{A}$ that can distinguish two interchangeable libraries, but there may exist a (non-polynomial-time) program $\mathscr{A}$ that can distinguish two indistinguishable libraries.

(b)

- $\frac{1}{2^{\lambda/2}}$ is negligible, because $2^{\lambda/2} = \left(\sqrt{2}\right)^{\lambda}$ is exponential.
- $\frac{1}{2^{\log(\lambda^2)}}$ is not negligible, because $2^{\log(\lambda^2)} = \lambda^{\frac{2}{\log_2 e}}$ is a lower order infinity than some polynomial (e.g. λ^2).
- $\frac{1}{\lambda^{\log\lambda}}$ is negligible, because for any finite order n, there exists $\lambda_0 \geq \exp(n)$ such that $\forall\lambda > \lambda_0$, $\lambda^{\log\lambda} > \lambda^n$, proving that $\lambda^{\log\lambda}$ is a higher order infinity than any polynomial.
- $\frac{1}{\lambda^2}$ is obviously not negligible, because λ^2 is a polynomial.
- $\frac{1}{2^{(\log\lambda)^2}}$ is negligible. Since $2^{(\log\lambda)^2} = \lambda^{\frac{2}{\log_2 e}\log\lambda}$, for any finite order n, there exists $\lambda_0 \geq \exp(n\log_2 e/2)$ such that $\forall\lambda > \lambda_0$, $\lambda^{\frac{2}{\log_2 e}\log\lambda} > \lambda^n$, proving that $2^{(\log\lambda)^2}$ is a higher order infinity than any polynomial.
- $\frac{1}{(\log\lambda)^2}$ is obviously not negligible, because $\log(\lambda)^2 < \lambda^2$ for λ large enough.

- $\frac{1}{\lambda^{1/\lambda}}$ is obviously not negligible, because $\lambda^{1/\lambda} < \lambda$ for $\lambda > 1$.
- $\frac{1}{\sqrt{\lambda}}$ is obviously not negligible, because $\sqrt{\lambda} < \lambda$ for $\lambda > 1$.
- $\frac{1}{2^{\sqrt{\lambda}}}$ is negligible, because for any $k \in \mathbb{N}$,

$$\lim_{\lambda \to +\infty} \frac{\lambda^k}{2^{\sqrt{\lambda}}} = \lim_{\lambda \to +\infty} \exp\left(k \log \lambda - \sqrt{\lambda} \log 2\right) = \exp(-\infty) = 0.$$

So $2^{\sqrt{\lambda}}$ is a higher order infinity than any polynomial.

(c) (1) Since $2\max(f, g) > f + g$, $P(\lambda) \times (2\max(f, g)) > P(\lambda)(f + g)$. And there is

$$\lim_{\lambda \to +\infty} P(\lambda) \times (2\max(f, g)) = 2 \lim_{\lambda \to +\infty} P(\lambda) \max(f, g) = 0.$$

So

$$\lim_{\lambda \to +\infty} P(\lambda)(f + g) = 0.$$

In other words, $f + g$ is negligible.

(2) By definition,

$$\lim_{\lambda \to +\infty} P(\lambda) \times (f \times g) = \lim_{\lambda \to +\infty} (P(\lambda) \times f) \times g = 0 \times 0 = 0.$$

shows that $f \times g$ is negligible.

(3) For example, $f(\lambda) = \exp(-\lambda)$ and $g(\lambda) = \lambda \exp(-\lambda)$ are both negligible, but $f(\lambda)/g(\lambda) = 1/\lambda$ is not negligible.