Quiz 3

110550108 施柏江

Problem 1

† Data compression is often used in data storage and transmission. Suppose you want to use data compression in conjunction with encryption. Does it make more sense to:

Ans: ABC

如果先壓縮再加密，會減少原始資料的大小，從而減少需要加密的資料量，可能有助於提高加密效率。如果先加密再壓縮，加密後的資料對壓縮演算法來說只是一個位元組序列，壓縮後也不會受到加密的影響。

Problem 2

† Let G: {0, 1}^s → {0, 1}^n be a secure PRG. Which of the following is a secure PRG:

Ans: BFGH

G'(k) = G(k) ‖ G(k): 密文重複會被找出規律，輸出並不是隨機的

G'(k) = G(k⊕1^s): 把 keyword 取 inverse 後不會影響輸出的隨機性

G'(k) = G(0): 能得知 keyword=0，輸出並不是隨機的

G'(k) = G(1): 能得知 keyword=1，輸出並不是隨機的

G'(k) = G(k) ‖ 0: 最後一個 bit 必為 0，輸出並不是隨機的

G'(k1, k2) = G(k1) ‖ G(k2): 兩個隨機的密文接在一起輸出依然是隨機的

G'(k) = reverse(G(k)): 把密文反過來輸出依然是隨機的

G'(k) = rotation(G(k)): 把密文做位移輸出依然是隨機的

要符合 secure PRG，輸出必須是隨機的，因此答案選 BFGH

Problem 3

Let (E, D) be a (one-time) semantically secure cipher with key space K = {0, 1}^k . A bank wishes to split a decryption key k ∈ {0, 1}^k into two pieces p1 and p2 so that both are needed for decryption. The piece p1 can be given to one executive and p2 to another so that both must contribute their pieces for decryption to proceed. The bank generates random k1 in {0, 1}^k and sets k1' ← k⊕k1. Note that k1⊕k1' = k. The bank can give k1 to one executive and k1' to another. Both must be present for decryption to proceed since, by itself, each piece contains no information about the secret key k (note that each piece is a one-time pad encryption of k). Now, suppose the bank wants to split k into three pieces p1, p2, p3 so that any two of the pieces enable decryption using k. This ensures that even if one executive is out sick, decryption can still succeed. To do so the bank generates two random pairs (k1, k1') and (k2, k2') as in the previous paragraph so that k1⊕k1' = k2⊕k2' = k. How should the bank assign pieces so that any two pieces enable decryption using k, but no single piece can decrypt?

Ans: C

當 p1 = (k1, k2), p2 = (k1', k2), p3 = (k2')，p1 和 p2 可以用 k1 和 k1'解密，p2 和 p3 可以用 k2 和 k2'解密，p1 和 p3 可以用 k2 和 k2'解密。如果只有 p1 或 p2 或 p3，則無法同時獲得 k1 和 k1'或 k2 和 k2'，因此無法解密。

Problem 4

Let M = C = K = { 0, 1, 2, . . . , 255 } and consider the following cipher defined over (K, M, C): E(k, m) = m + k (mod 256); D(k, c) = c - k (mod 256) Does this cipher has perfect secrecy?

Ans: Yes

因為它是 one-time pad，每個密鑰都是隨機且只使用一次，所以即使攻擊者擁有所有密文，也無法推斷出任何有關明文的信息，具有 perfect secrecy。

Problem 5

† Let (E, D) be a (one-time) semantically secure cipher where the message and ciphertext space is {0, 1}^n . Which of the following encryption schemes are (one-time) semantically secure?

Ans: BDFG

E’(k, m) = E(0^n , m): 可以利用已知的密鑰 0^n 來區分出明文 0 和 1 之間的差異，從而破壞 semantic security。

E’((k, k’), m) = E(k, m) ‖ E(k’, m): 由於兩個密文的產生是獨立的，即第一個密文 E(k, m)不會洩漏關於第二個密文 E(k', m)的任何訊息，反之亦然。因此無法通過分析一個密文來獲得有關另一個密文的訊息，具有 semantic security。

E’(k, m) = E(k, m) ‖ MSB(m): 可以考慮明文 0^n 和明文 10^(n-1)的情況。對於明文 0^n，其 MSB 是 0；而對於明文 10^(n-1)，其 MSB 是 1。由於 E'將明文的 MSB 直接串聯在加密後的密文後面，可以通過觀察密文的最後一位來判斷明文的 MSB 是 0 還是 1，破壞 semantic security。

E’(k, m) = 0 ‖ E(k, m): 由於加密的第一部分始終是固定值 0，與明文無關，因此對於任何明文 m，第一部分都是相同的。加密的第二部分是使用密鑰 k 對明文 m 進行加密的結果，這個加密過程本身是安全的，無法從密文中獲得任何有關明文 m 的信息。所以將兩部分串接在一起依然保持 semantic security。

E’(k, m) = E(k, m) ‖ k: 密鑰就透漏在了密文之中，破壞了 semantic security。

E’(k, m) = reverse(E(k, m)): 因為將加密後的結果進行反轉不會洩漏任何有關明文的信息，仍具有 semantic security。

E’(k, m) = rotation(E(k, m)): 因為將加密後的結果進行位移不會洩漏任何有關明文的信息，仍具有 semantic security。

Problem 6

Suppose you are told that the one time pad encryption of the message ”attack at dawn” is 6c73d5240a948c86981bc294814d (the plaintext letters are encoded as 8-bit ASCII and the given ciphertext is written in hex). What would be the one time pad encryption of the message ”defend at noon” under the same OTP key?

Ans: 6962c720079b8c86981bc89a994d

先把 attack at dawn 的明文和密文 xor 起來獲得 OTP key，再把 defend at noon 的明文和 key xor 起來獲得密文。

```python
def str_to_hex(s):
    return int(s.encode().hex(), 16)

ciphertext = 0x6c73d5240a948c86981bc294814d
key = str_to_hex('attack at dawn') ^ ciphertext
ans = hex(str_to_hex('defend at noon') ^ key)
print(ans)
```

Output: `0x6962c720079b8c86981bc89a994d`

Problem 7

As shown below, consider a tree with n = 16 leaves. Suppose the leaf node labeled 25 corresponds to an exposed DVD player key. Check the set of keys below under which to encrypt the key k so that every player other than player 25 can decrypt the DVD. Only four keys are needed.

Ans: DEFG (26, 6, 1, 11)

因為 key 25 位於 key 0 的右側，所以可以包含所有 key 1 底下。因為 key 25 位於 key 2 的左側，所以可以包含所有 key 6 底下。因為 key 25 位於 key 5 的右側，所以可以包含所有 key 11 底下。因為 key 25 位於 key 12 的左側，所以可以包含 key 26。因此答案是 26, 6, 1, 11。

Extra Credit

Did SHA-256 and SHA-512-truncated-to-256-bits have the same security properties? Which one is better? Please explain in detail.

Ans: 他們有一些相同的 security properties，像是找到兩個不同的輸入讓它們產生相同 hash 的難度非常高。還有即便知道 hash，也很難推斷出 input。兩者並不一定誰比較好，需要根據特定的需求來選擇。SHA-256 的運行速度通常比 SHA-512-truncated-to-256-bits 快，因為它處理的資料量更少。而 SHA-512-truncated-to-256-bits 在當 hash function 被用於比 hash 更厲害的屬性，例如構建 MAC 或 pseudorandom values 能比 SHA-256 更安全。