1. **Discuss why DL-based identification scheme is better than traditional password-based identification scheme? Which one is more efficient in terms of computation and communication complexities? (10 points)**

   Ans:

   DL-based identification scheme 相較於 traditional password-based identification scheme 雖然密碼容易被猜測、竊取和截取，使其容易受到暴力破解、字典攻擊和釣魚等攻擊，然而 DL-based identification scheme 使用數學算法和密碼學技術來確保安全識別，使攻擊者難以破壞系統，提供更強的安全性。 DL-based identification scheme 提供更強的身份驗證能力，因為它們依賴於密碼學密鑰和數字簽名，相較於使用者生成的密碼，更能抵禦攻擊。這些方案還通過引入基於時間或隨機數的挑戰和響應機制，提供對 replay attacks 的抵抗能力。

   在可擴展性和易用性方面，DL-based identification scheme 允許在多個系統中使用單一密鑰對，簡化了用戶體驗，減輕了管理多個密碼的認知負擔。 就效率而言，由於涉及加密、解密和數字簽名操作，離散對數方案的計算複雜性通常較高。然而，硬體的進步和優化的算法提高了這些操作的效率，使其在大多數應用中變得實用。 離散對數方案中的通信複雜性可能涉及密鑰和憑證的交換而增加額外的開銷。然而，在現代網絡環境中，這種開銷通常很小，並且由方案提供的增強安全性所抵銷。 總體而言，離散對數識別方案相較於基於密碼的方案提供更強的安全性、身份驗證能力、對 replay attacks 的抵抗能力和可擴展性。雖然它們可能具有較高的計算和通信複雜性，但它們提供的好處使其成為安全識別的更有效和堅固的選擇。

2. **This problem is about ElGamal encryption and signature schemes. (30 points)**

   (a) Show that ElGamal encryption scheme is not secure against the chosen ciphertext attack.

   Ans:

   If such an Oracle exist

   如果有人想要解密密文 $c = (c_1, c_2)$, with $c = g^k$ and $c_2 = my^k$, 選擇隨機數 k'和 m'。Oracle to decrypts $c' = (c_1 \cdot g^{k'}, mm'y^{k+k'})$. Oracle sends mm', the plaintext of $c' = (g^{k+k'}, mm'y^{k+k'})$ 給入侵者. 入侵者可以輕鬆分解 m' 然後得到明文 m of c。

   (b) Is ElGamal signature scheme secure against the chosen plaintext attack (allowing to ask the signing oracle) if the hash-then-sign paradigm is used.

   Ans:

   當使用 ElGamal signature 時，若沒有搭配 hash function，它是存在可偽造的。ElGamal signature scheme 如果搭配 hash function h，並對 hash value 進行簽名應用在原始訊息上，ElGamal signature scheme 才可以抵禦 chosen plaintext attack。所以要偽造訊息並不簡單，對手必須找到一些有意義的訊

息m'，而$h(m') = m$. 如果h是 collision-resistant hash function，對手的成功機率其實非常低。

(c) Assume that the hash-then-sign paradigm is not used. Can we forage a signature for any given message m by asking the signing oracle. You cannot ask the oracle about the signature of m.

Ans:

We can query the Oracle for any message except of m.

因此設計一個偽造演算法:

1. **Query the Oracle for message m', where $m/m' = u \bmod (p-1)$. (Oracle return $(r = g^k \bmod p, s = k^{-1}(m - rx) \bmod (p-1))$)**
2. **計算$s' = su \bmod (p-1)$ and r' $s.t.$ $r' \equiv ru \bmod (p-1)$ and $r' \equiv r \bmod p$**
3. **檢查驗證步驟: $y'r^{s'} = y^{ru}r^{su} = (y^r r^s)^u = \left(g^{m'}\right)^u = g^m$**
4. **返回 (m, r', s')**

3. Assume that Alice and Bob know the common (p, g), where p is a large prime and g is a generator of Z_p. (30 points)

(a) If they want to exchange a large amount of messages through the Internet securely, what can they do?

Ans:

根據第四章投影片 page34，A 和 B 使用 Diffie-Hellman key change，並共享通用密鑰保護他們的訊息。

A→B: $c = g^a$ $where$ $a \in_R Z_{p-1}$;

B→A: $d = g^b$ $where$ $b \in_R Z_{p-1}$;

A 計算: $k = d^a = g^{ab}$

B 計算: $k = c^b = g^{ab}$

(b) If an attacker wants to break the communication, what can he do?

Ans:

對手 Eve 可以使用 Man-in-the-middle attack 看到訊息:

$$A \xrightarrow{c=g^a} Eve \xrightarrow{c'=g^{a'}} B$$

$$B \xrightarrow{d=g^b} Eve \xrightarrow{d'=g^{b'}} A$$

A 計算: $k_{A,Eve} = d'^a = g^{ab'}$

B 計算: $k_{B,Eve} = c'^b = g^{a'b}$

(c) Assume that p = 107 and g = 2. Show examples for (a) and (b).

Ans:

Choose $a = 10, b = 20$

1. A→B: $61 = 2^{10}\ where\ 10 \in_R Z_{106}$
   A→B: $83 = 2^{20}\ where\ 20 \in_R Z_{106}$
   A 計算: $25 = 83^{10} = 2^{10*20}$
   B 計算: $25 = 61^{20} = 2^{10*20}$
2. Let's Choose $a' = 15, b' = 16$

   $A \xrightarrow{61=2^{10}} Eve \xrightarrow{26=2^{15}} B$

   $B \xrightarrow{83=2^{20}} Eve \xrightarrow{52=2^{16}} A$

   A 計算: $k_{A,Eve} = 52^{10} = 2^{10*16} = 105$
   B 計算: $k_{B,Eve} = 26^{20} = 2^{15*20} = 89$

4. Show that the regular RSA signature scheme is "arbitrarily forgeable" (forging the signature of any challenge message m) if the attacker is allowed to ask the signing oracle. Note that the challenge message m cannot be queries to the signing oracle. (10 points)

Ans:

We forge the RSA signature σ of message m by querying the signing oracle the message $m' = m * r^e \ mod\ N$, where $r \in_R Z_N^*$ is chosen randomly.

The signing oracle will return the signature $\sigma' = m'^d = m^d * r \ mod\ N$.

Then we can compute the signature $\sigma = \sigma'/r = m^d \ mod\ N$

5.We consider the multi-authority secure electronic voting scheme without a trusted center. How do the authorities A1, A2, . . . , An collaboratively construct the public and private keys? (20 points)

Ans:

根據投影片第五章 45 頁:

Assume that $p = kq + 1, g \in_R G_q \backslash \{1\}$ are given.

1. Each A_i selects x_i and a $t-1$-degree polynomial $f_i(x)$ with $f_i(0) = x_i$ and publishes $h_i = g^{x_i}$
2. The publish key is $h = \prod_{i=1}^{N} h_i$
3. Share the secret key $x = \sum_{i=1}^{n} x_i = f(0)$, where $f(x) = \sum_{k=1}^{n} f_k(x)$
   -each A_i sends $s_{i,j} = f_i(j)$ to A_j via secure channel
   -each A_j computes its share $s_j = \sum_{k=1}^{n} s_{k,j} = \sum_{k=1}^{n} f_k(j) = f(j)$

Note: each A_j should check whether the received share $s_{i,j}$ from A_i is valid.