分組名單(不足5個人空著就好):

姓名	學號
陳以晴	111550178
蔡昀錚	111550035
莊婷馨	111550057
謝詠晴	111550113
鄭芯薇	111550064

1. **Name of the paper:**
   New Directions in Cryptography
   Invited Paper
   WHITFIELD DIFFIE AND MARTIN E. HELLMAN, MEMBER, IEEE

2. **Summary:**

這份文獻討論了當代密碼學的重要發展趨勢。文章指出，由於當前硬體的發展，使得高級密碼設備的成本大幅下降，進而產生大量的商業應用。此類應用使得加密方式的隱私 (Privacy) 和驗證 (Authentication) 特性被更加關注，希望能夠透過新的加密系統，減少使用安全通道分配密鑰的需求，並同時提供與手寫簽名相等的可驗證性。

作者首先探討了傳統密碼學的範疇，傳統密碼學主要能夠處理兩大類的安全問題，分別是隱私(privacy)和身份驗證(authentication)，隱私系統的作用是防止未經授權的第三方從傳遞的消息中提取重要信息，而身份驗證系統則防止未經授權的消息注入公共通道(public channel)中，確保接收者能驗證訊息是否合法。傳統密碼系統中信息傳輸的基本流程大致如下:發送者先生成未加密的明文消息，通過不安全的通道傳送給接收者，為了預防被竊聽，發送者還會生成密文，而密鑰會通過安全通道傳送給接收者，這樣一來接收者就可以透過密鑰運算密文來解出明文。

作者也提出了傳統密碼學需要改良的原因，由於傳統密碼學系統通常需要通過安全通道傳輸密鑰，但這種方法在現代商業環境中效率低下且成本高昂，因此有了新的加密機制誕生，也就是公鑰，登登登登。？？？

公鑰密碼學系統通過將加密和解密操作分別由不同的密鑰控制來解決密鑰分發問題。加密密鑰(public key)可以公開，解密密鑰(private key)需要保持秘密。每個用戶可以將其加密密鑰公開，允許任何用戶使用其加密並發送給他消息。公鑰分發系統則允許兩個用戶通過不安全的通道交換密鑰，這種系統使用數學困難問題(有限域 finite field 上的計算對數的困難性)來保證安全性。

在文章的第四個部份，作者提到可驗證性 (Authentication) 在商業應用的重要性。One-way authentication 要讓任何人都可以驗證其真實性，但這個簽名只能由合法簽名者生成。文中提出可以使用一對一函數 (one-way function) 來實現，此函數從計算的角度是不可逆的，需要10^{30}以上的計算量才能算出，簡單的應用如網站登入，首先在網站存放$f(PW)$，當使用者再次登入，輸入密碼 PW' 時，比較 $f(PW)$ 和 $f(PW')$ 是否相等，如此就保證沒有其他人知道該密碼，但又能驗證使用者身份。此外，作者也提出了一種基於公鑰系統的數位簽名方案，簽名者用密鑰"解密"自己的訊息並發送，所有人都可以用公鑰"加密"並得到原本的訊息。

最後，作者探討密碼的計算困難性。密碼系統的安全性通常基於解密的計算困難性，作者特別討論了NP完全問題，並提出了一種基於背包問題的一對一函數作為潛在

的密碼學工具。文章還討論了陷門函數 (trap-door one-way function) 和其他高級概念，這些概念有助於設計更安全的密碼系統。

總結而言，這篇文章提出了一些當代新的密碼學方向，並展示了如何利用數學和計算理論來設計更安全和高效的密碼系統，增進密碼的隱私性和可驗證性，使它能商業應用中更加普及。

3. Strength(s) of the paper:

1) 創新理念：提出了公開密鑰密碼系統的概念，這是當時密碼學領域的一個革命性突破，改變了傳統密碼學依賴私密密鑰分配的模式。
2) 提供理論基礎：文章結合信息理論和計算理論，為解決長期存在的密碼學問題提供了理論基礎，並探討了計算複雜度在密碼系統安全性評估中的應用。
3) 考慮實際應用：強調了新型密碼系統在商業應用中的潛力，如遠程取款機和計算機終端，展示了這些技術在現實世界中的實用性和必要性。
4) 歷史回顧全面與清晰：論文提供了對現有密碼學方法的全面回顧，清晰地指出了公開密鑰密碼學旨在填補的空白。這有助於將新的理念更好的在密碼學領域中被使用。
5) 引用和參考文獻豐富：論文引用了大量相關工作，支持作者的論點並為新提出的想法提供了堅實的基礎。

4. Weakness(es) of the paper:

1) 對非專業人士的複雜性：雖然對有些背景知識的人來說容易接近，但對於那些沒有密碼學和算法概念先驗理解的人來說，數學和理論的性質可能是一個挑戰。
2) 實施細節：這篇文章主要是理論性的，並未深入探討實際實施問題或在部署公開密鑰基礎設施時可能出現的實際挑戰。
3) 安全考量：在撰寫時，公開密鑰加密的某些安全影響，如對某些類型攻擊的脆弱性，未能完全探討，在這方面可能過於理想化。
4) 論文發佈年代久遠：由於此篇論文大約於五十年前發表，並非所有在論文中做出的預測和假設都經得起時間的考驗，隨著新的密碼技術和更複雜的攻擊方法的發展，許多理論已經無法保證安全性以及實用性，導致其難以在現代應用。

5. Your own reflection, which can include but not limited to:

A. What did you learn from this paper?

1) 公開密鑰密碼學的概念：了解公開密鑰密碼學的基本原理及其在信息安全中的重要性，這種密碼學技術不僅解決了傳統密碼學中的密鑰分配問題，還提供了一種新的數據加密和認證方法。
2) 計算複雜度在密碼學中的應用：理解計算複雜度理論如何應用於密碼系統的安全性評估，特別是NP問題和NP完全問題在構建安全系統中的作用。
3) 理論與實踐的結合：認識理論發展如何推動實踐應用的進步，並看到理論研究在設計安全和高效密碼系統中的關鍵角色。

B. How would you improve or extend the work if you were the author?

1) 太過理論化以及太多文字敘述:本文獻
2) 提供實現細節和實驗數據:增加具體的實現細節和實驗數據, 以驗證公開密鑰密碼系統的實用性和安全性。這將有助於從理論走向實踐, 讓讀者更清楚地了解該系統的運作。
3) 討論擴展性和應用場景:深入討論公開密鑰系統在大規模應用中的可擴展性問題, 提供具體的解決方案來應對大規模網絡中的密鑰管理和分發挑戰。
4) 簡化技術術語:減少過難術語的使用, 增加更多解釋和示例, 使密碼學背景較淺的讀者也能理解文章內容。

C. What are the unsolved questions that you want to investigate?

1) 公開密鑰系統的實用性:此嶄新的系統在不同實際應用場景中的表現如何？它在處理大量用戶時的效率和安全性如何？(當然由於此論文年代久遠, 我們現在已經知道表現十分優異)
2) 計算複雜度的進一步研究:計算複雜度理論能否提供更強有力的證據來證明某些加密方法的安全性？是否存在其他未被發現的NP完全問題可以用於密碼學？
3) 抗量子計算攻擊:公開密鑰系統能否抵抗量子計算的攻擊？如何設計能夠抵抗量子計算的加密算法？
4) 數位簽名和身份驗證問題:目前現有的電子認證技術還無法提供遠端數位簽名系統的身份驗證保障, 特別是在防止傳輸上仍有隱私洩露的疑慮。

D. What are the broader impacts of this proposed technology?

1) 信息安全的增強:透過引入公鑰分配系統, 兩個不認識的用戶可以在不安全的通道上安全地交換密碼, 為信息安全領域帶來了革命性變革, 特別是在數字通信和互聯網安全方面。
2) 商業應用的普及:論文中提及數位簽名的方法, 該技術對於電子商務、網絡銀行和其他需要可靠身份驗證的應用場景的發展十分重要, 為數據加密提供了可靠的技術支持。
3) 後續技術的發展:論文發表隔年RSA算法的提出是公開密鑰密碼學的一個重要實現, 進一步推動了該領域的發展和應用。
4) 降低密鑰分發的成本和時間:傳統的密鑰分發需要依賴於物理渠道(如快遞或掛號郵件), 這不僅耗時且成本高昂, 而公鑰加密技術能夠消除這一瓶頸, 使密鑰分發變得更加便捷和經濟高效。
5) 法律和政策影響:公開密鑰密碼學在法律和政策上的影響, 包括隱私保護、數據安全法律法規的制定和實施。
6) 遠程通信的安全性提升:隨著通信網絡發展迅速, 這項技術能夠確保跨越全球的通信安全, 防止竊聽和不合法消息的注入, 從而促進更廣泛的數據和信息交流。

E. Else?

這篇論文在密碼學發展史具有革命性的影響，為後續研究（例如：RSA技術）和實踐應用提供了非常堅實的基本理念，所以現今許多技術都可以在這篇論文中看到相似的理論基礎。且將密碼學轉向科學化，加解密技術背後的理論能夠有邏輯性地被證明是安全的系統，從而提高加密技術的可靠性和可驗證性。

這篇文章的重要性在於它提出了公開密鑰加密的概念，這不僅解決了密鑰分發的問題，也開創了數字簽名的應用。它將加密學從一門依賴於保密機制的藝術推向了一門依賴於計算理論的科學。這些概念和技術對於現代的通信系統安全性有著深遠的影響。

6. Experiment lab

使用 Diffie-Hellman 密鑰交換協議生成一個共享密鑰，並利用這個密鑰通過 AES (高級加密標準) 來加密和解密信息。

1. Diffie-Hellman 密鑰交換:
- 首先定義了兩個函數 `diffie_hellman` 和 `compute_shared_secret` 來實現密鑰交換過程。
- `diffie_hellman` 函數接受一個質數 p，一個基 g 和一個私鑰，計算並返回對應的公鑰 $g^{private\ key} mod\ p$。
- `compute_shared_secret` 函數用來計算共享的秘密，它基於對方的公鑰和自己的私鑰進行計算 $public\ key^{private\ key} mod\ p$。

2. 生成 AES 密鑰:
- 使用 Diffie-Hellman 協議計算出的共享秘密作為輸入，通過 SHA-256 哈希函數生成一個長度更適合的密鑰，然後截取前 16 字節作為 AES 的密鑰。

3. AES 加密和解密:
- 定義了 `encrypt` 和 `decrypt` 函數來進行 AES 的加密和解密操作。
- `encrypt` 函數中，使用 CBC (密碼塊鏈) 模式和一個隨機生成的 16 字節初始化向量 (IV) 來加密資料。
- 在加密前，使用 PKCS#7 填充標準來處理資料，確保資料長度符合 AES 的塊大小要求。
- `decrypt` 函數則進行相反的操作，先解密數據，再去除填充來恢復原始文本。

4. 加解密流程演示:
- Alice 使用共享的 AES 密鑰來加密一條信息 "Hello, Bob!"。
- Bob 使用相同的密鑰來解密這條信息。

這個代碼從密鑰交換到信息加密解密提供了一個完整的加密通信流程，展示了如何在通信雙方間安全地共享和使用密鑰。

Readme

該程式碼展示如何實作 Diffie-Hellman 密鑰交換算法。該算法允許兩個用戶在不安全的通道上生成共享密鑰，而不需要提前交換秘密信息，最後使用生成的共享密鑰進行AES加密和解密。

- 需要用到的套件：
  - cryptography：提供加密算法和模式的實現，包括AES和CBC模式
  - hashlib：提供SHA-256哈希函數
  - random：生成隨機數
  - os：生成隨機初始化向量（IV）

- 如何執行：
  - pip install -r requirements.txt
  - python diffie_hellman.py
- 執行步驟：
  - 首先，定義質數p和原根alpha。(在此例中訂為p = 23, alpha = 5)
  - Alice 和 Bob 分別利用random生成他們的私鑰a,b，並經過 diffie_hellman() 函數計算公鑰A,B。
    “public_key = alpha^private_key mod p”
  - Alice會將其公鑰傳給Bob，Bob將其公鑰傳給Alice，再藉由 compute_shared_secret() 函數計算出共享密鑰。
    “shared_secret = A^b mod p = B^a mod p
  - 使用SHA-256對共享密鑰進行hash，並取前16個字節作為AES密鑰。
  - 當Alice要傳送訊息給Bob，他會使用共享密鑰對消息進行AES-CBC加密，而Bob同樣會使用共享密鑰對加密後的信息解密。
  - encrypt()加密函數：在加密前，使用 PKCS7 將原文填充到AES塊大小(16字節)的倍數，確保資料長度符合 AES 的塊大小要求，接著使用 CBC(密碼塊鏈)模式和一個隨機生成的16字節初始化向量(IV)來加密資料。
  - decrypt()解密函數：進行相反的操作，先解密數據，再去除填充來恢復原始文本。
- 執行結果：

```
Alice's Public Key:  14
Bob's Public Key:  10
Shared Secret (Alice):  7
Shared Secret (Bob):  7
Ciphertext:  b'2~\xf2\x82f\x10\xa7\x1b}b\xb5|\x9ev\xff/lC\x8e\xf4m\xce\xb7\xd0\xfd"!\x9d\x02I\xc6\xda'
Decrypted Message:  Hello, Bob!
```