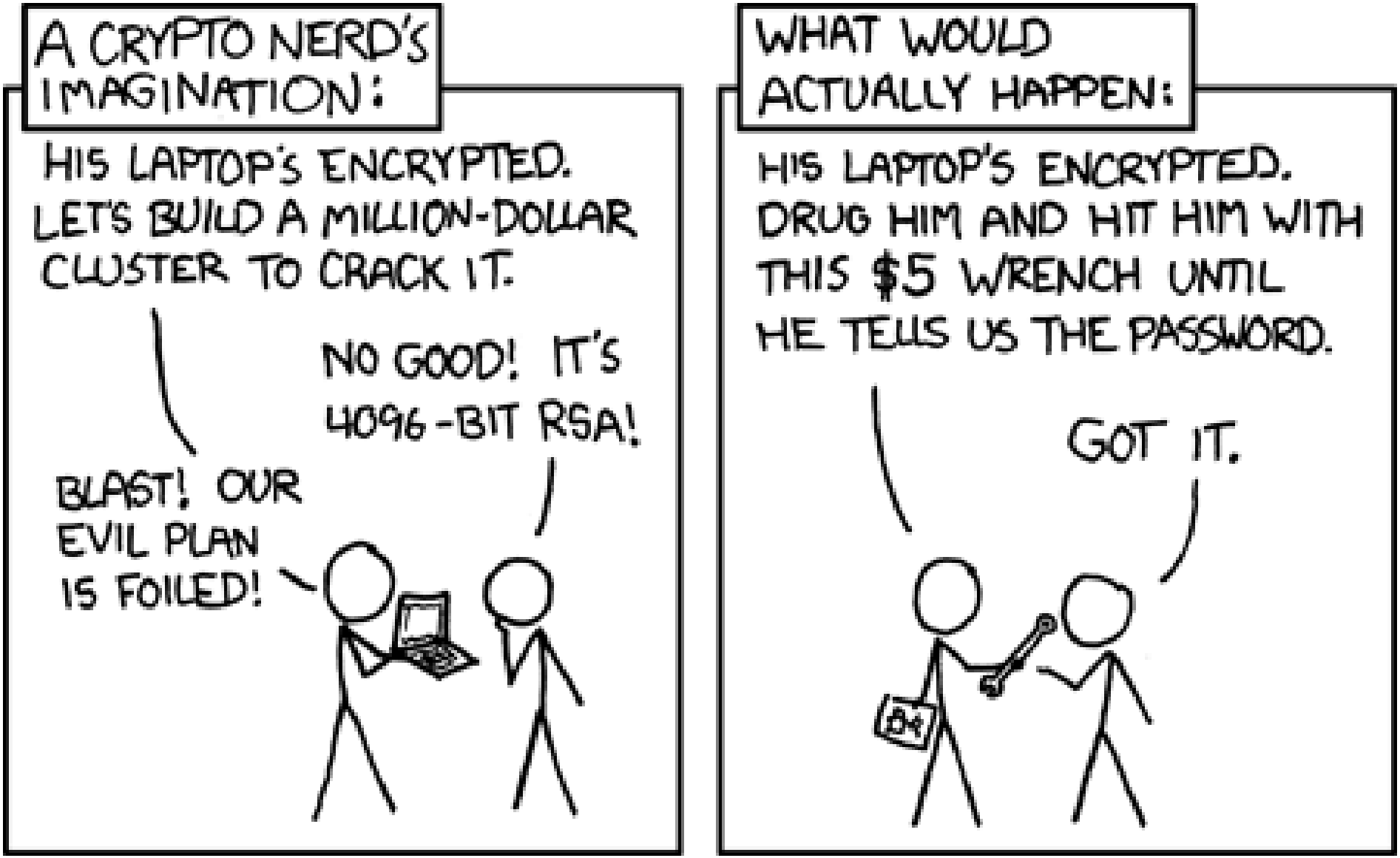
# Cryptanalysis in real life

周立平 等研究團隊

# Do u understand real hackers ?



# 綱目

第一章　緒論

第二章　Mifare Classic Card介紹

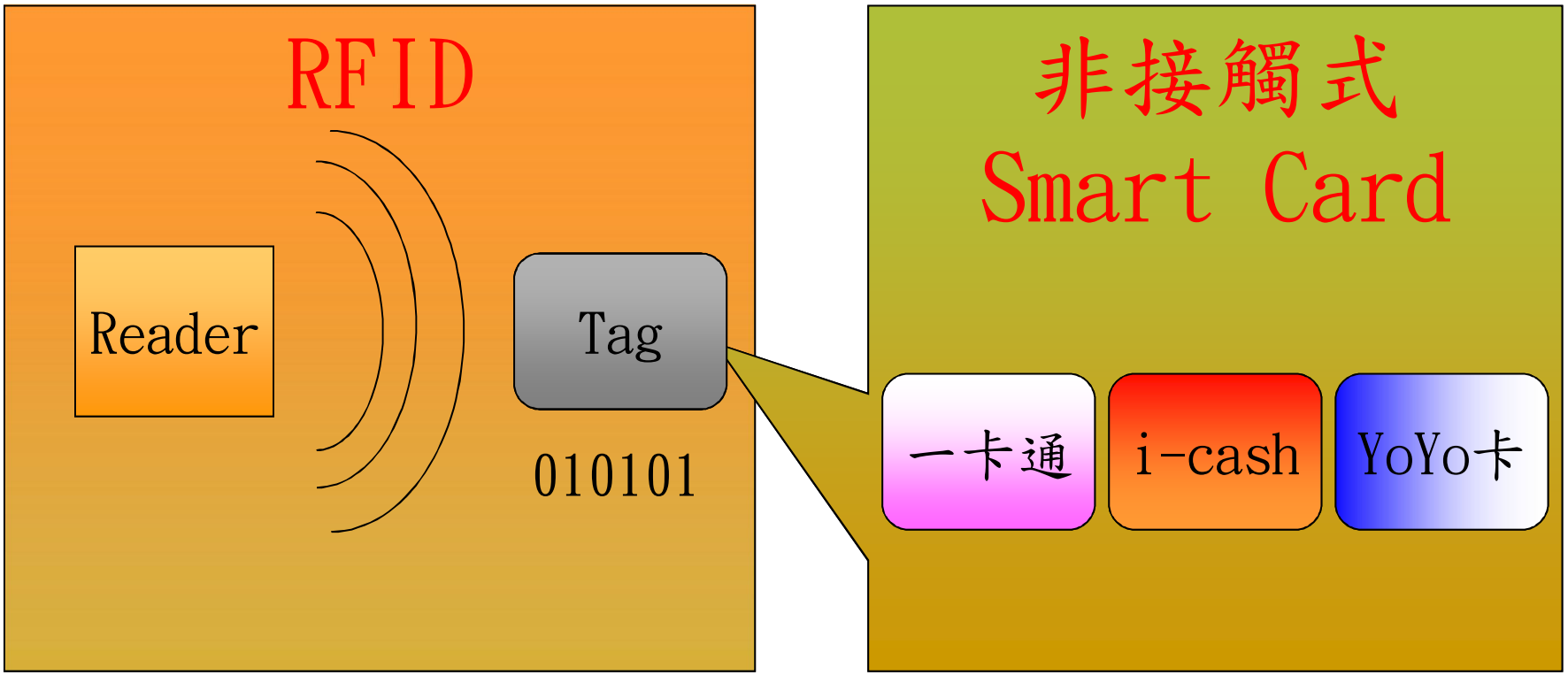
第三章　Mifare Classic Card現有攻擊方式

第四章　二代卡的特性

第五章　自然人憑證

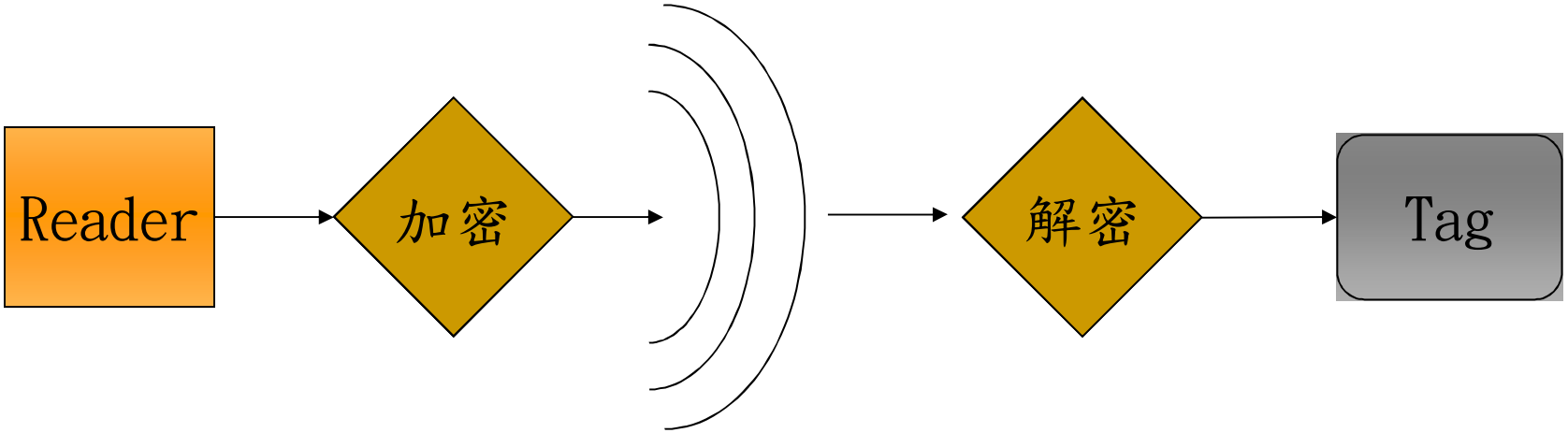
# 第一章　緒論

## 攻擊背景

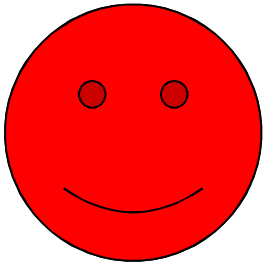


# 第一章　緒論

## 攻擊手段




竊聽



# 第一章　緒論

## 現狀

| 新/舊卡 | 改良方式 |
| --- | --- |
| 新卡 | 用3DES etc |
| 舊卡 | 改善弱點 增加金鑰長度 etc |


0.8美元/張，全球已發行約10億張

# 第二章　Mifare Classic Card介紹

第一節　Mifare Classic Card規格與結構

第二節　Mifare Classic Card認證協定

第三節　Crypto-1加密演算法

第四節　Mifare Classic Card弱點

# 第二章　Mifare Classic Card介紹

## 第一節　Mifare Classic Card規格與結構

### 規格與特性

1. 操作頻率：13.56 MHz。
2. 傳輸速度：106 Kbps。
3. 傳輸距離：10cm。

# 第二章　Mifare Classic Card介紹

## 第一節　Mifare Classic Card規格與結構

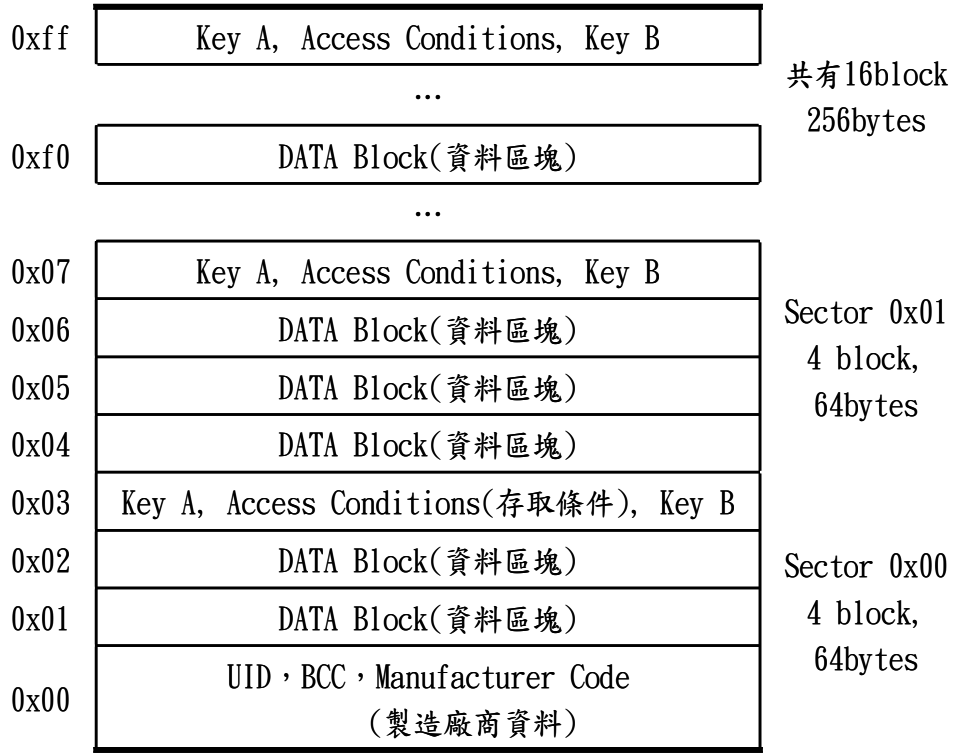
### 資料儲存結構

| Sector | Block 0 | Block 1 | Block 2 | Block 3 |
|---|---|---|---|---|
| 0 | Manufacturer Code | Data Block | Data Block | Key A, Access Conditions, Key B |
| 1 | Data Block | Data Block | Data Block | Key A, Access Conditions, Key B |
| 2 | Data Block | Data Block | Data Block | Key A, Access Conditions, Key B |
| 3 | Data Block | Data Block | Data Block | Key A, Access Conditions, Key B |
| 4 | Data Block | Data Block | Data Block | Key A, Access Conditions, Key B |
| 5 | Data Block | Data Block | Data Block | Key A, Access Conditions, Key B |
| 6 | Data Block | Data Block | Data Block | Key A, Access Conditions, Key B |
| 7 | Data Block | Data Block | Data Block | Key A, Access Conditions, Key B |
| 8 | Data Block | Data Block | Data Block | Key A, Access Conditions, Key B |
| 9 | Data Block | Data Block | Data Block | Key A, Access Conditions, Key B |
| 10 | Data Block | Data Block | Data Block | Key A, Access Conditions, Key B |
| 11 | Data Block | Data Block | Data Block | Key A, Access Conditions, Key B |
| 12 | Data Block | Data Block | Data Block | Key A, Access Conditions, Key B |
| 13 | Data Block | Data Block | Data Block | Key A, Access Conditions, Key B |
| 14 | Data Block | Data Block | Data Block | Key A, Access Conditions, Key B |
| 15 | Data Block | Data Block | Data Block | Key A, Access Conditions, Key B |

# 第二章　Mifare Classic Card介紹

## 第一節　Mifare Classic Card規格與結構

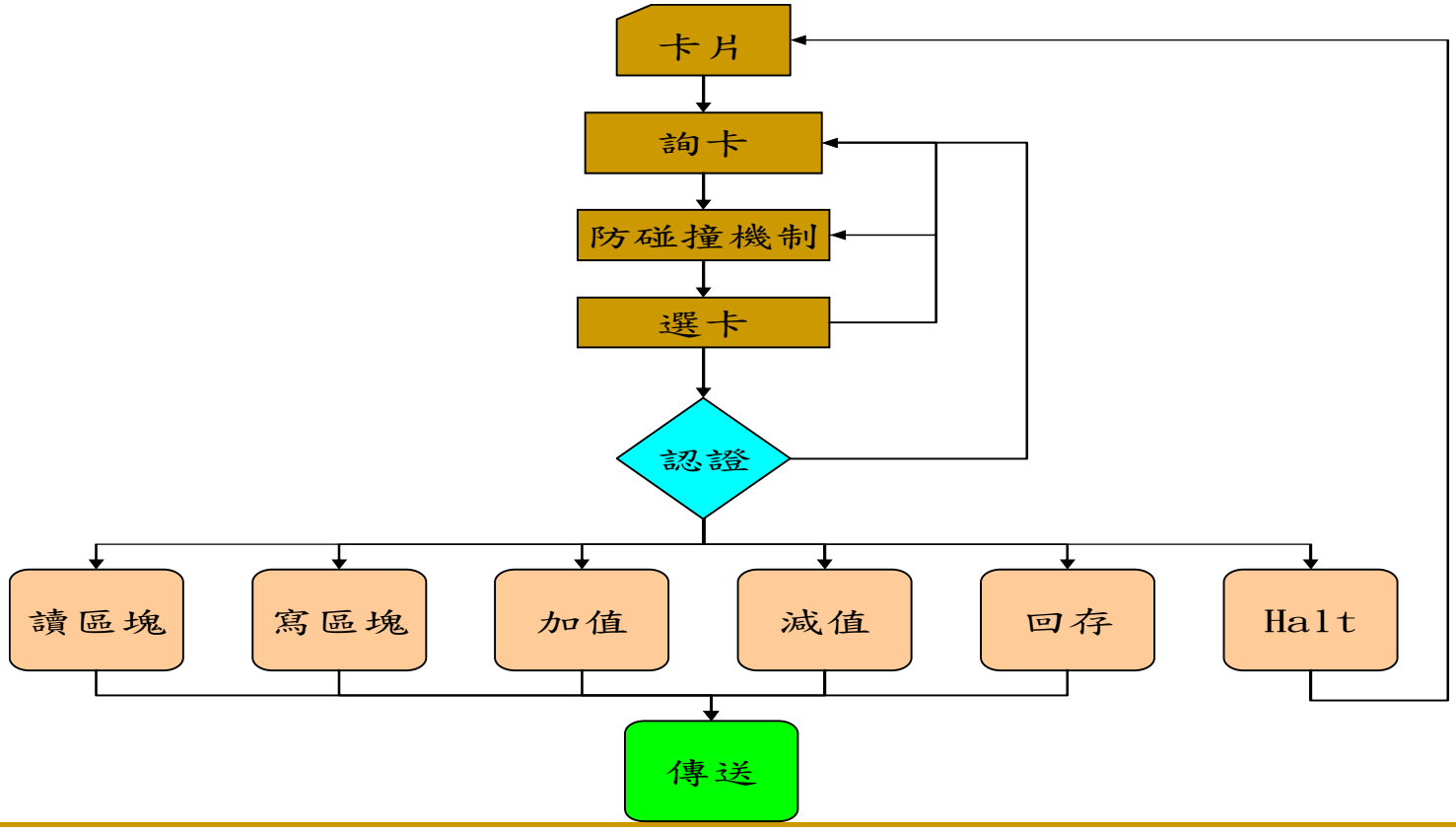
### 記憶體儲存結構



# 第二章　Mifare Classic Card介紹

## 第一節　Mifare Classic Card規格與結構

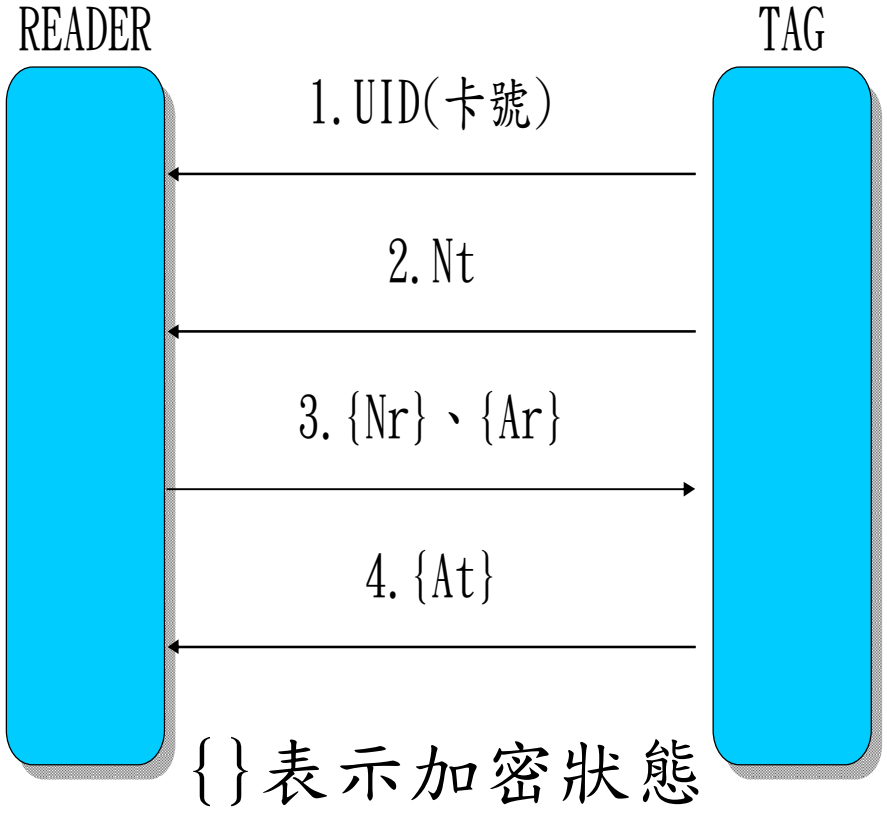
### 通訊流程



# 第二章　Mifare Classic Card介紹

## 第二節　Mifare Classic Card認證協定

### 一、Mifare Classic卡認證流程



# 第二章　Mifare Classic Card介紹
# 第二節　Mifare Classic Card認證協定

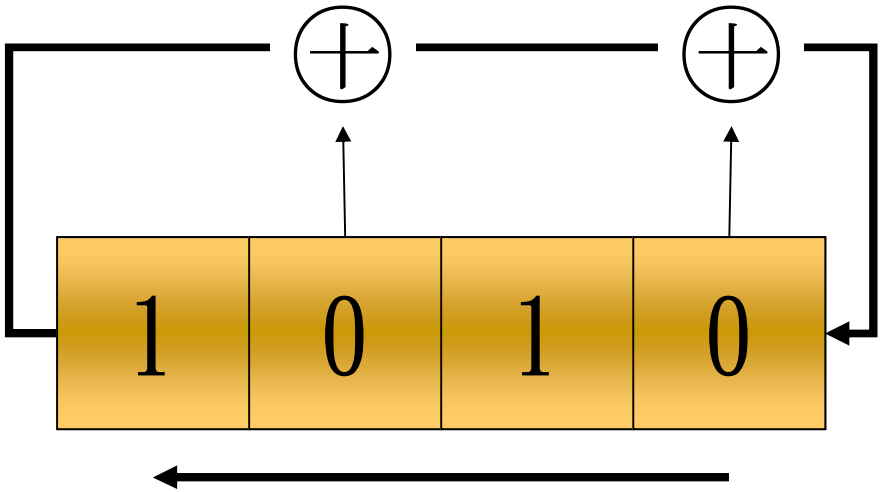
## 二、通訊範例

| Step | 發送者 | Hex(16進位內容) | ISO 14443 指令 | 註解 |
|---|---|---|---|---|
| 0 | RD | 26 | REQUEST | Hi, I am Reader, Is any card here ? |
| 1 | TAG | 04 00 | AWAKE | Hello, I am here. |
| 2 | RD | 93 20 | Polling | Who are you ? |
| 3 | TAG | 9C 59 9B 32 6C | UID | I am 9C 59 9B 32 6C |
| 4 | RD | 93 70 9C 59 9B 32 6C 6B 30 | ANTI COLL | OK, I want to talk to you 9C 59 9B 32 6C |
| 5 | TAG | 08 B6 DD | TAG TYPE | Ok. My card type is Mifare Classic 1K |
| 6 | RD | 60 00 F5 7B | AUTH | 開始認證，請問 00 Block |
| 7 | TAG | 82 A4 16 6C | Nt | 明文 Nt |
| 8 | RD | EF EA 1C DA 8D 65 73 4B | Nr + Nt' | 密文 {Nr} + {Ar} |
| 9 | TAG | 9A 42 7B 20 | Nt" | 密文 {At} |

# 第二章　Mifare Classic Card介紹

## 第三節　Crypto-1加密演算法

### 一、LFSR（Linear Feedback Shift Registers）線性反饋位移暫存器

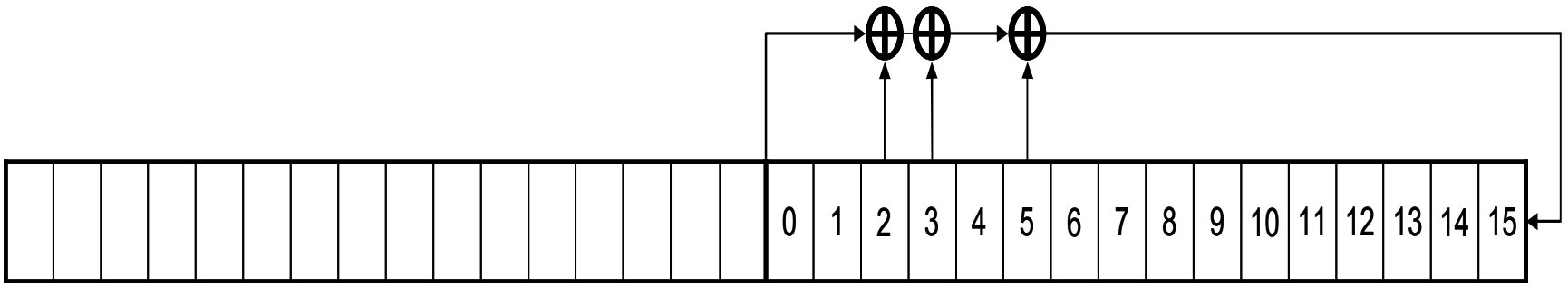



n個位元的LFSR，變化周期最大為$2^n-1$種

# 第二章　Mifare Classic Card介紹

## 第三節　Crypto-1加密演算法

### 二、PRNG(Pseudo-random Number Generator)偽亂數產生器



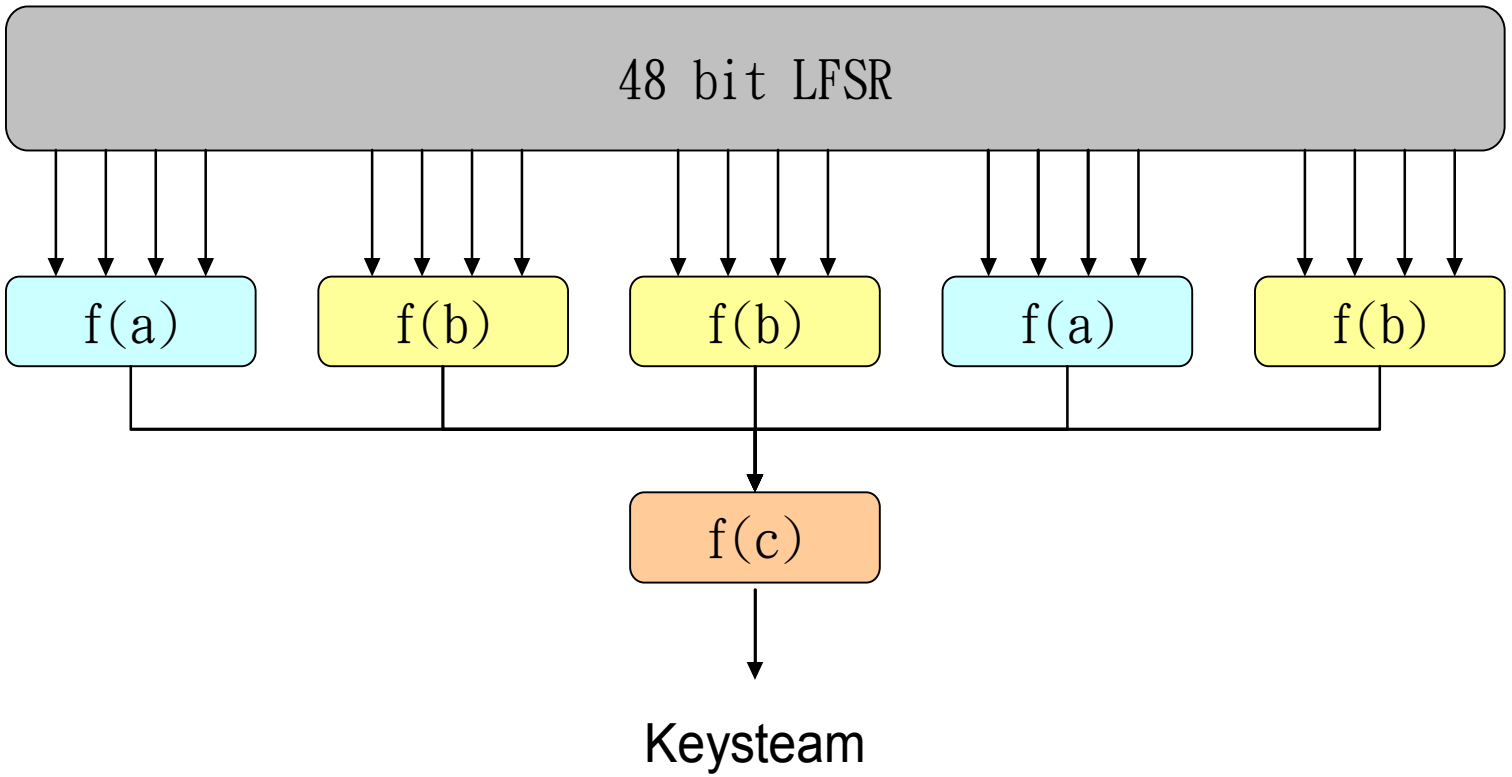

$$L\ (x_0x_1...x_{15}) := x_0 \oplus x_2 \oplus x_3 \oplus x_5$$

$$suc(x_0x_1...x_{31}) := x_1x_2...x_{31}L\ (x_{16}x_{17}...x_{31})$$

# 第二章　Mifare Classic Card介紹

## 第三節　Crypto-1加密演算法

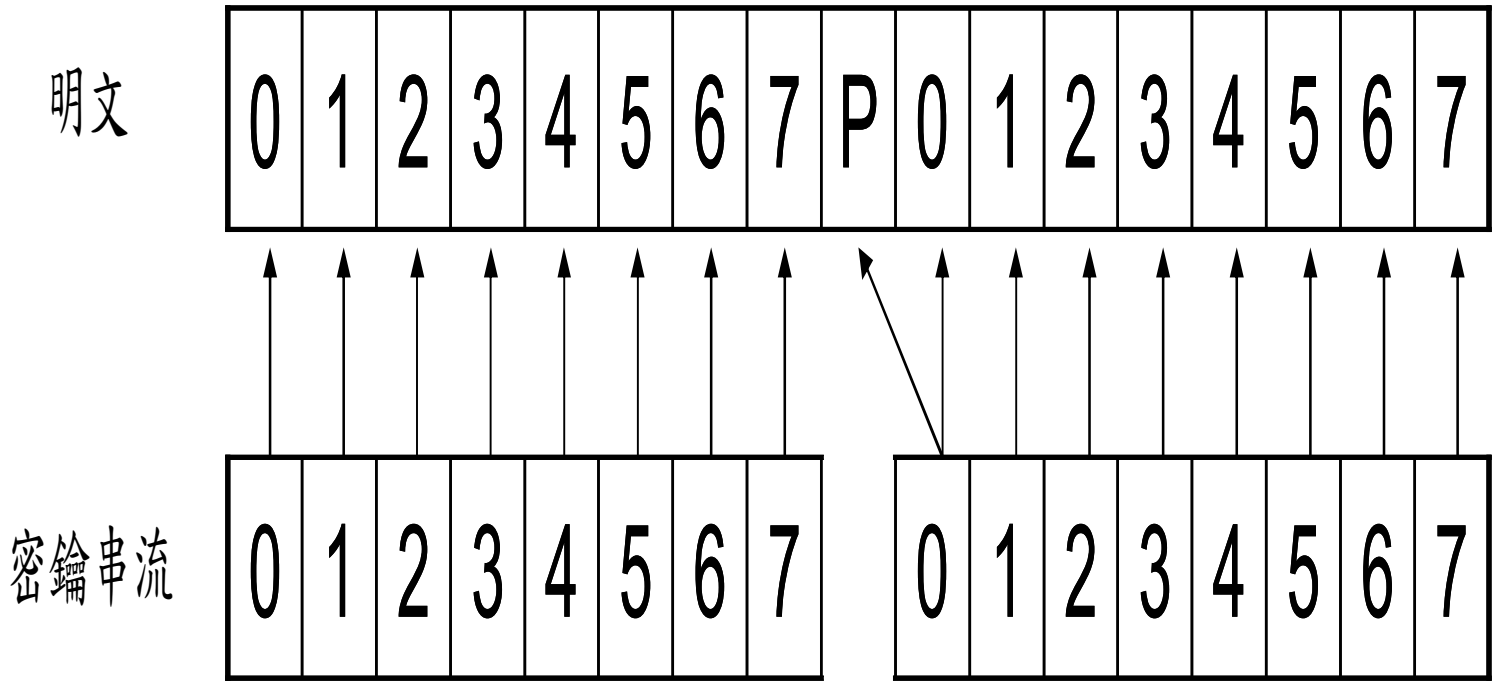
### 三、Filter Function



# 第二章　Mifare Classic Card介紹

## 第三節　Crypto-1加密演算法

### 四、同位元(Parity Bit)

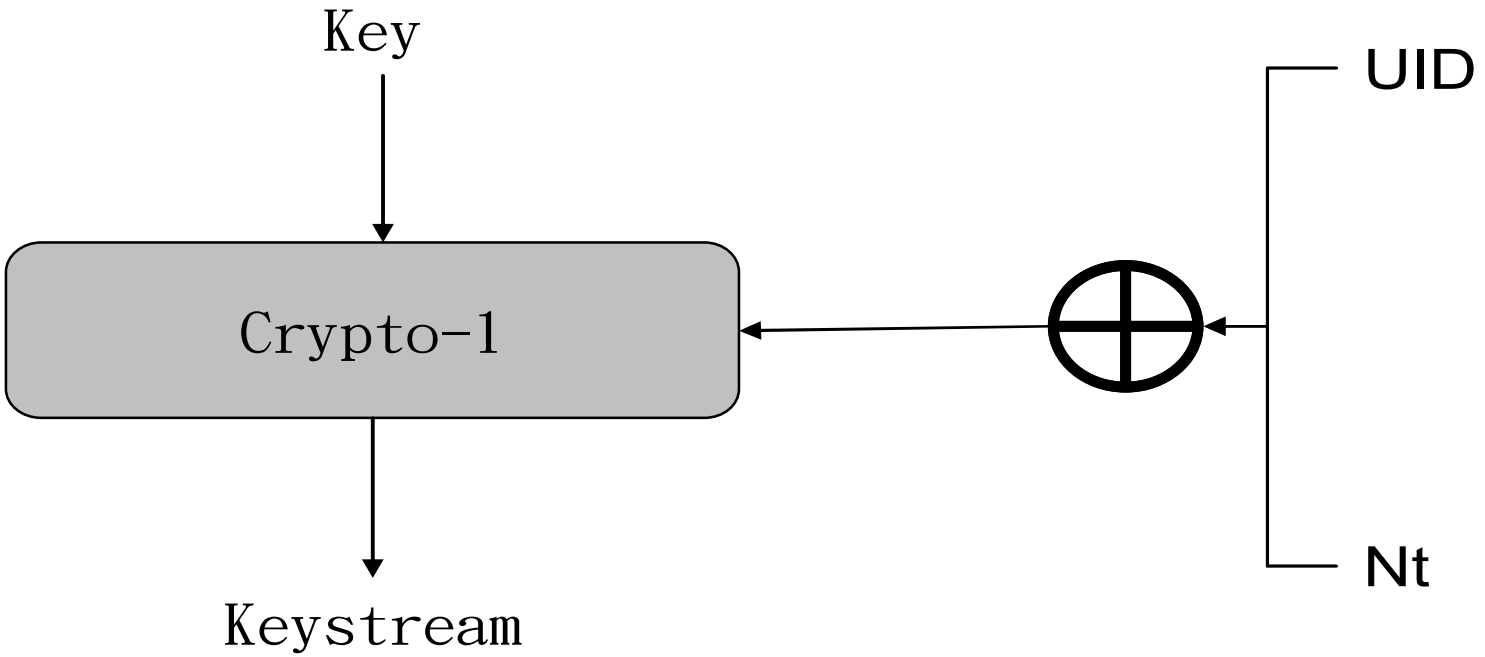


# 第二章　Mifare Classic Card介紹

## 第三節　Crypto-1加密演算法

### 五、Crypto-1演算法加密過程

(一)Crypto-1初始狀態設定

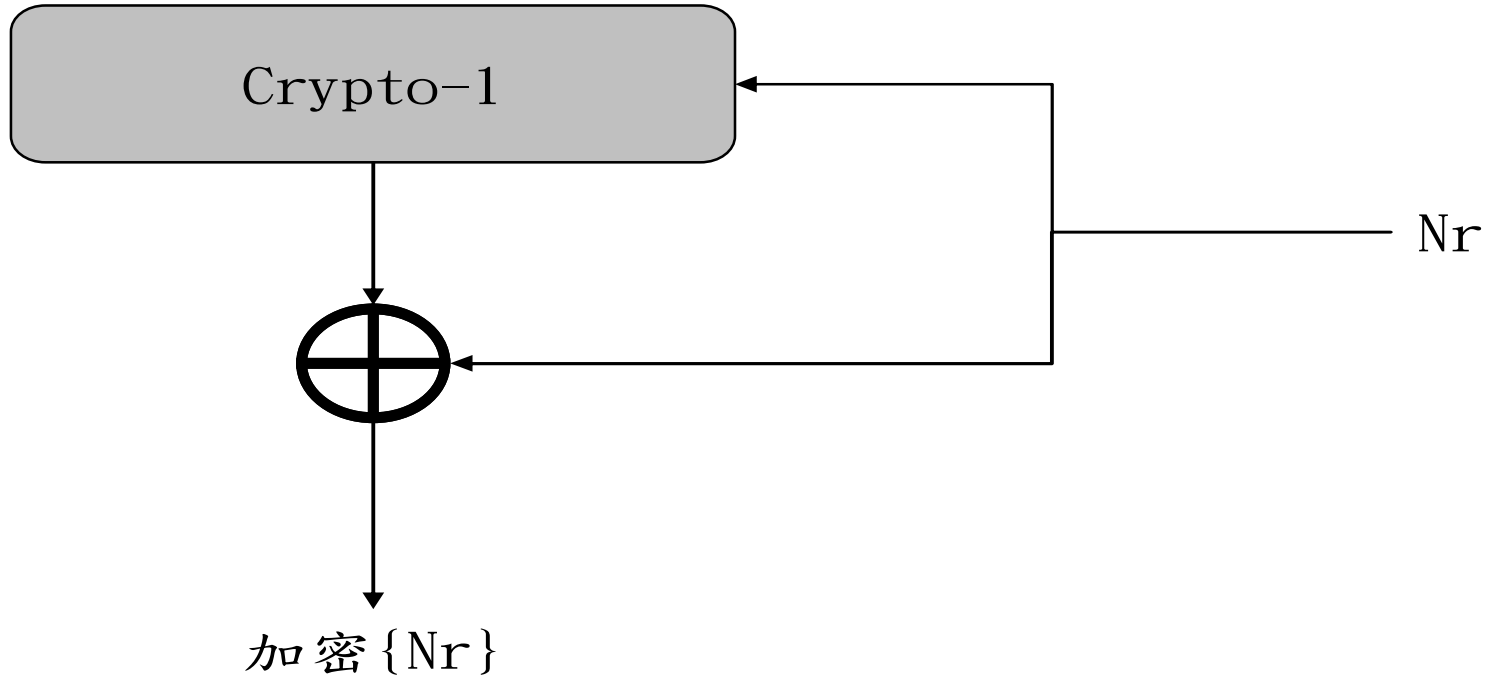


# 第二章　Mifare Classic Card介紹

## 第三節　Crypto-1加密演算法

### 五、Crypto-1演算法加密過程

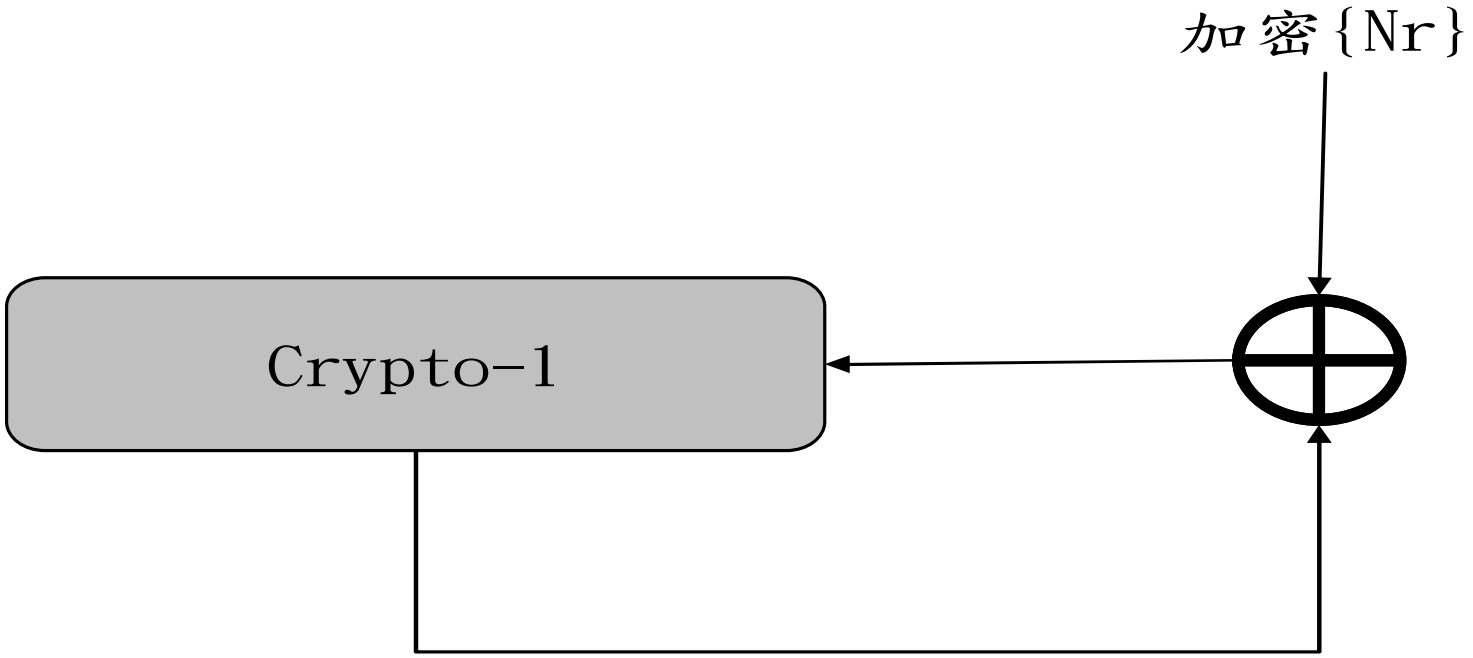
(二) Nr加密



# 第二章　Mifare Classic Card介紹
## 第三節　Crypto-1加密演算法

### 五、Crypto-1演算法加密過程

(三)Tag產生Ar、At前的Crypto-1狀態

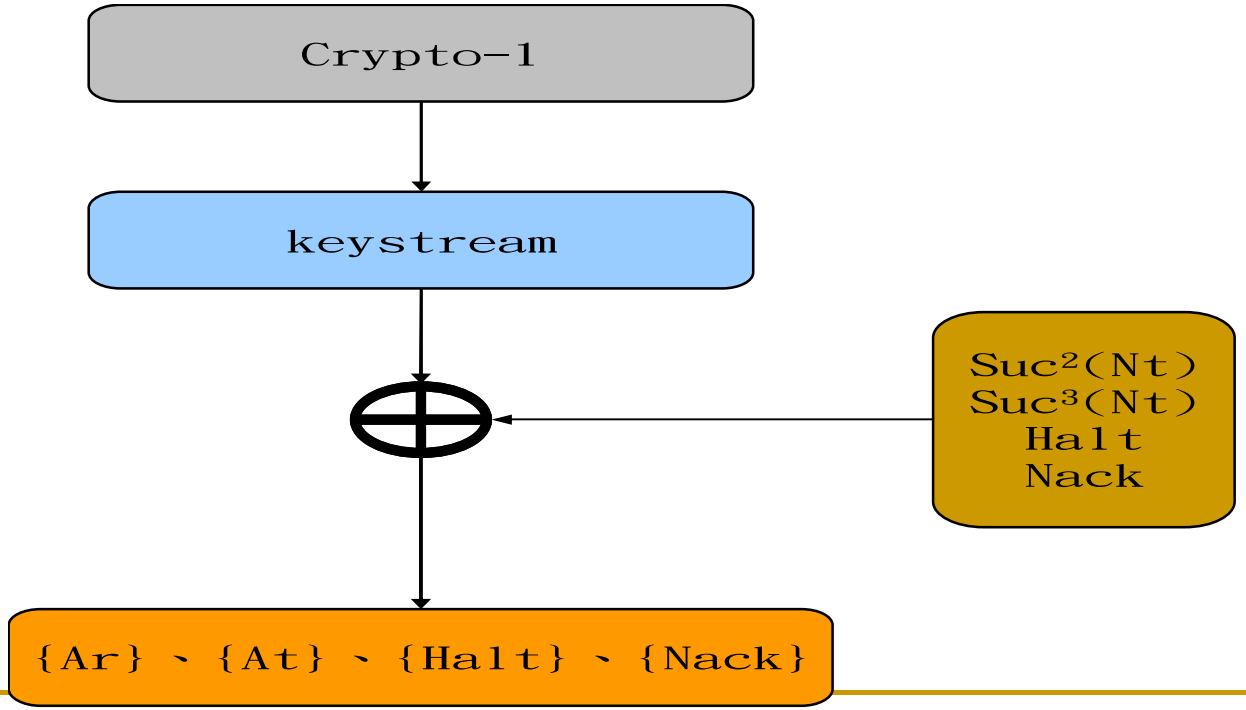


# 第二章　Mifare Classic Card介紹

## 第三節　Crypto-1加密演算法

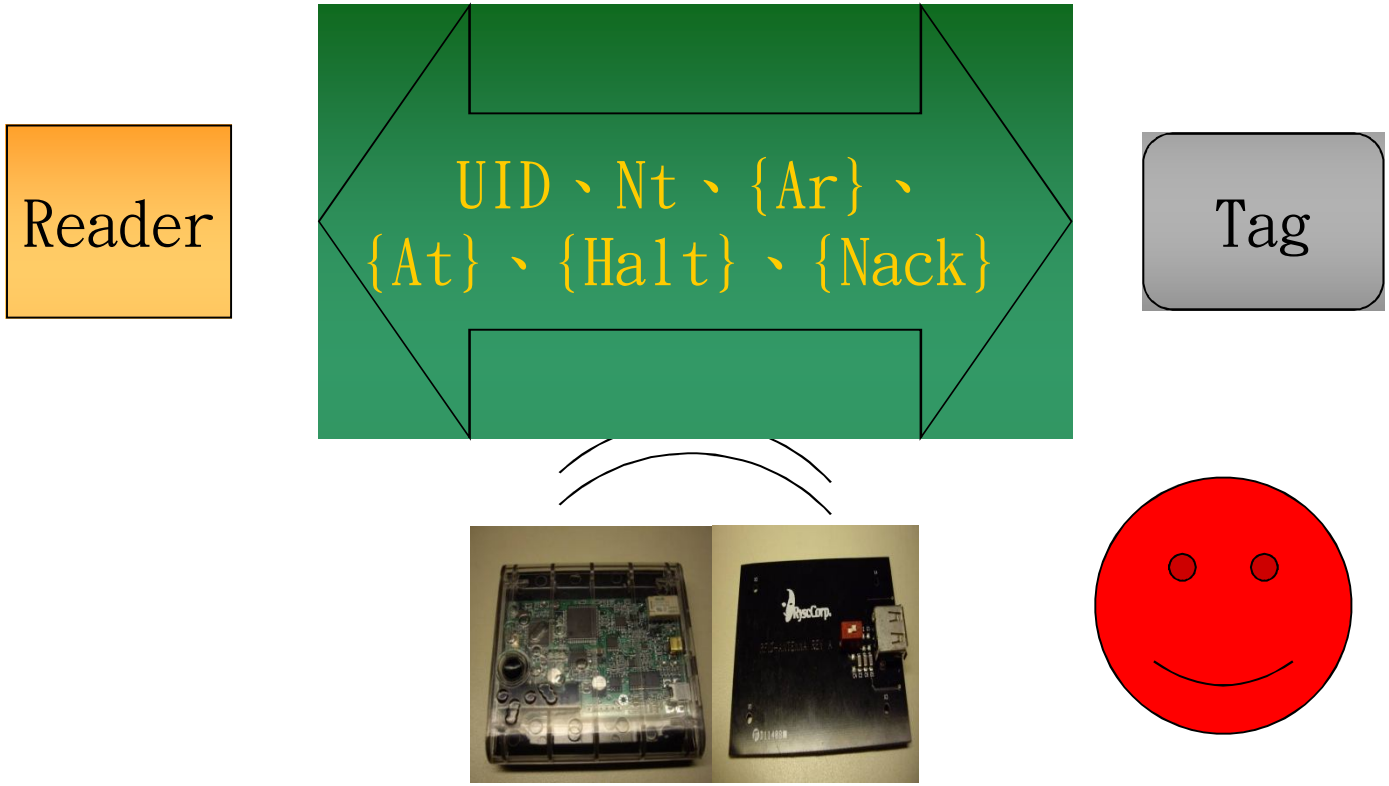
### 五、Crypto-1演算法加密過程

(四)Reader產生Ar、Tag產生At的密文



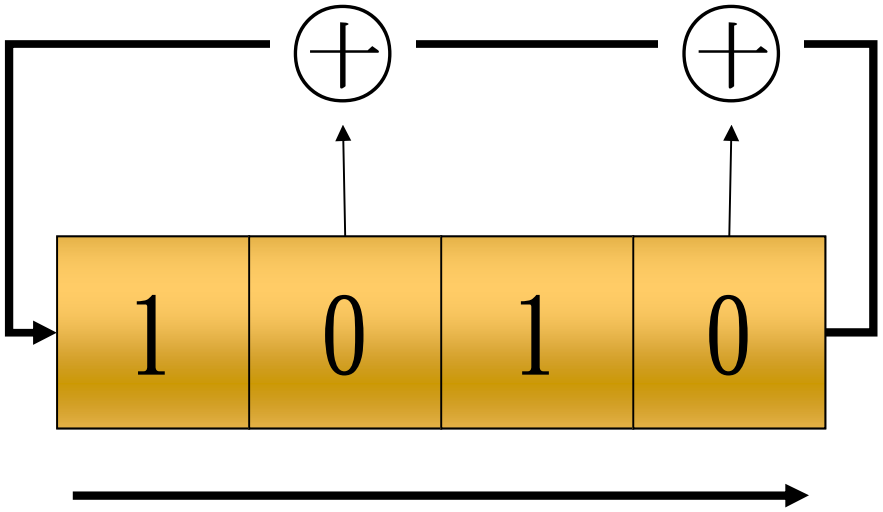
# 第二章　Mifare Classic Card介紹
# 第四節　Mifare Classic Card弱點

## 一、密鑰串流之取得



# 第二章　Mifare Classic Card介紹
# 第四節　Mifare Classic Card弱點

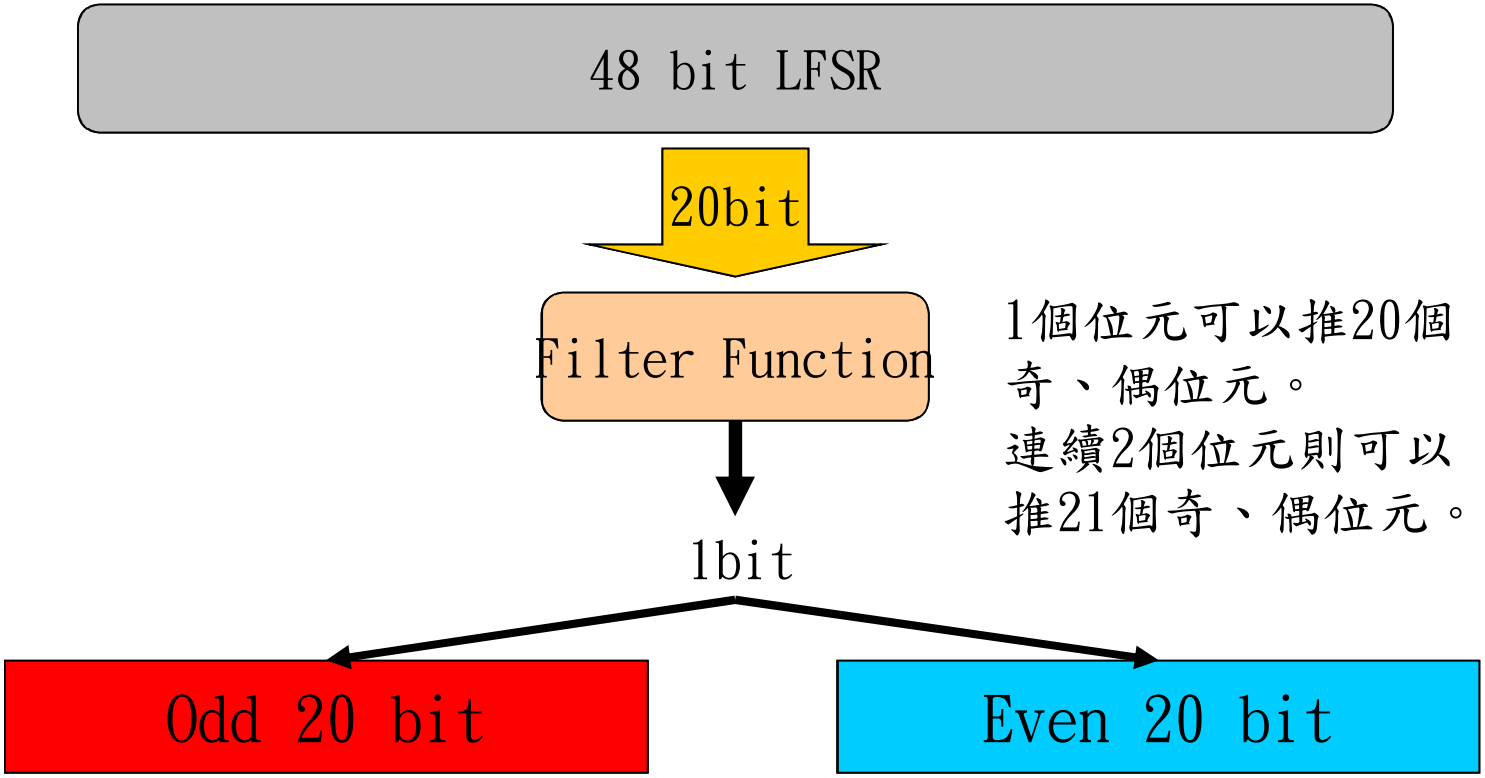
## 二、LFSR Rollback




$$R(x_1x_2...x_{47}\ L(x_0x_1...x_{47})) = x_0$$

# 第二章　Mifare Classic Card介紹
# 第四節　Mifare Classic Card弱點

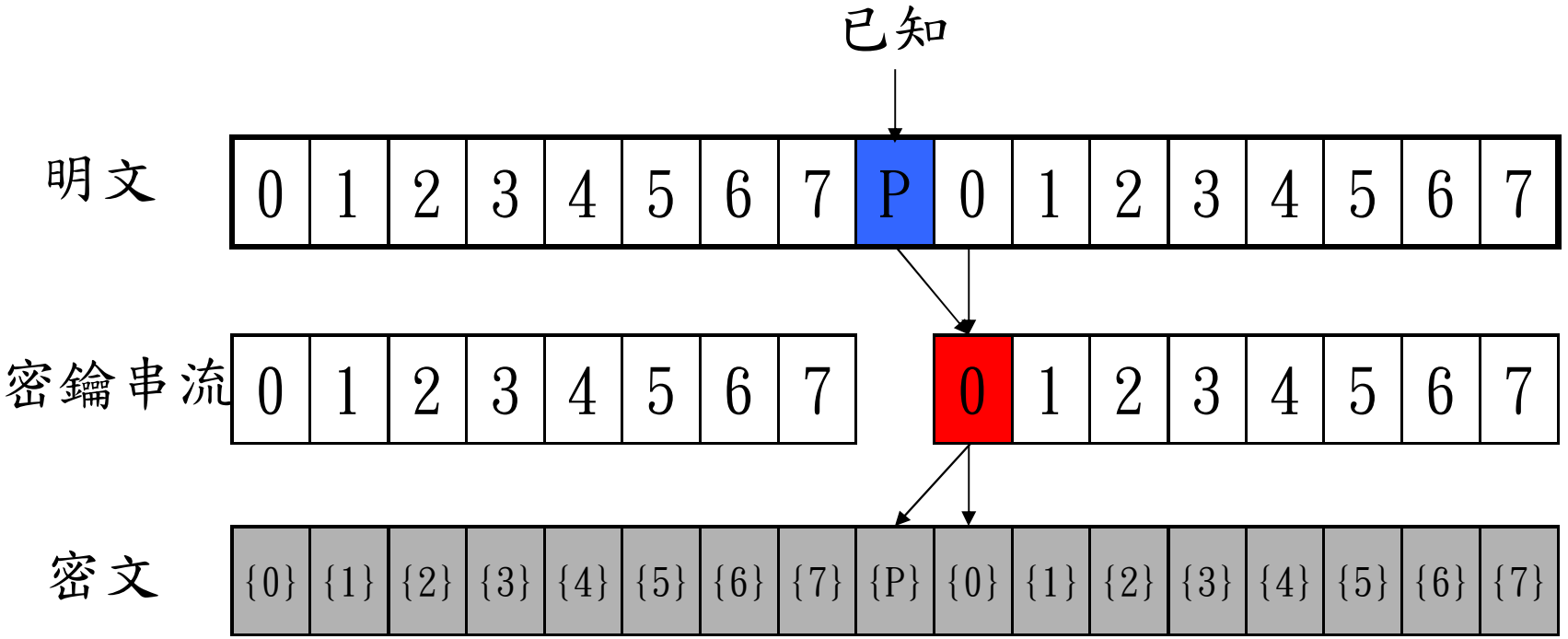
## 三、Inputs to Filter Function




1個位元可以推20個奇、偶位元。
連續2個位元則可以推21個奇、偶位元。

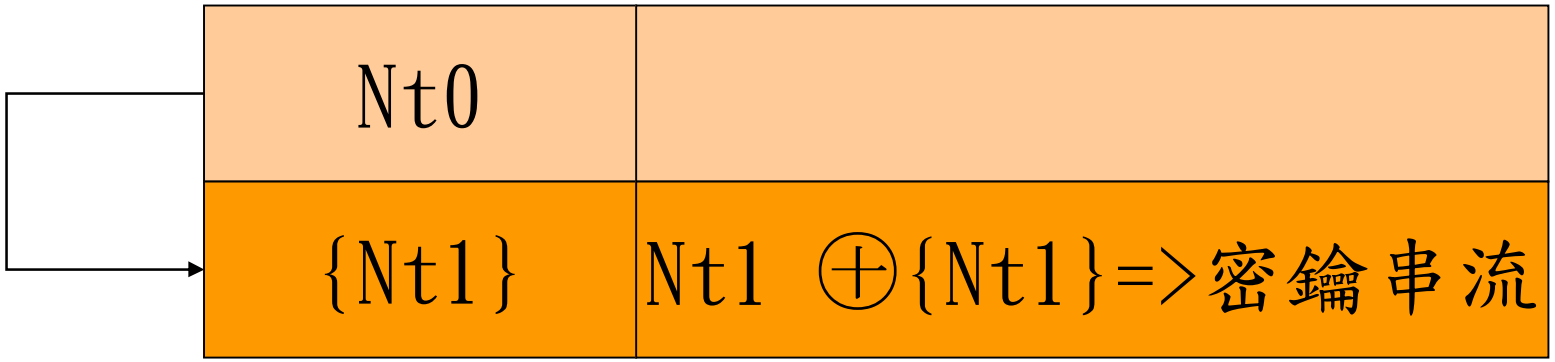
# 第二章　Mifare Classic Card介紹
## 第四節　Mifare Classic Card弱點

### 四、Parity Bit



# 第二章　Mifare Classic Card介紹
# 第四節　Mifare Classic Card弱點

## 五、Nested Authentications




Nt：$2^{16}$-1種的變化

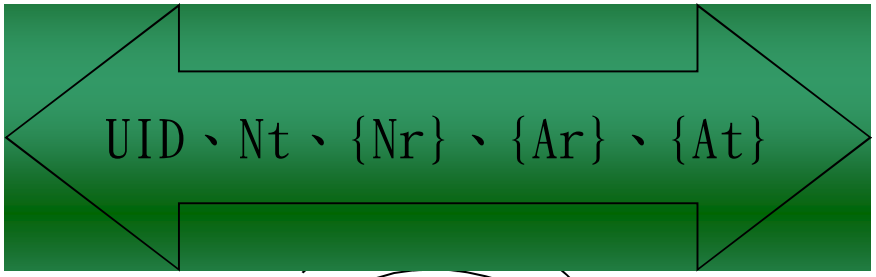
# 第三章　Mifare Classic Card
## 現有攻擊方式

竊聽攻擊法

{Nr}差分攻擊法

# 第三章　Mifare Classic Card現有攻擊

## 竊聽攻擊法




Reader

真的讀卡機

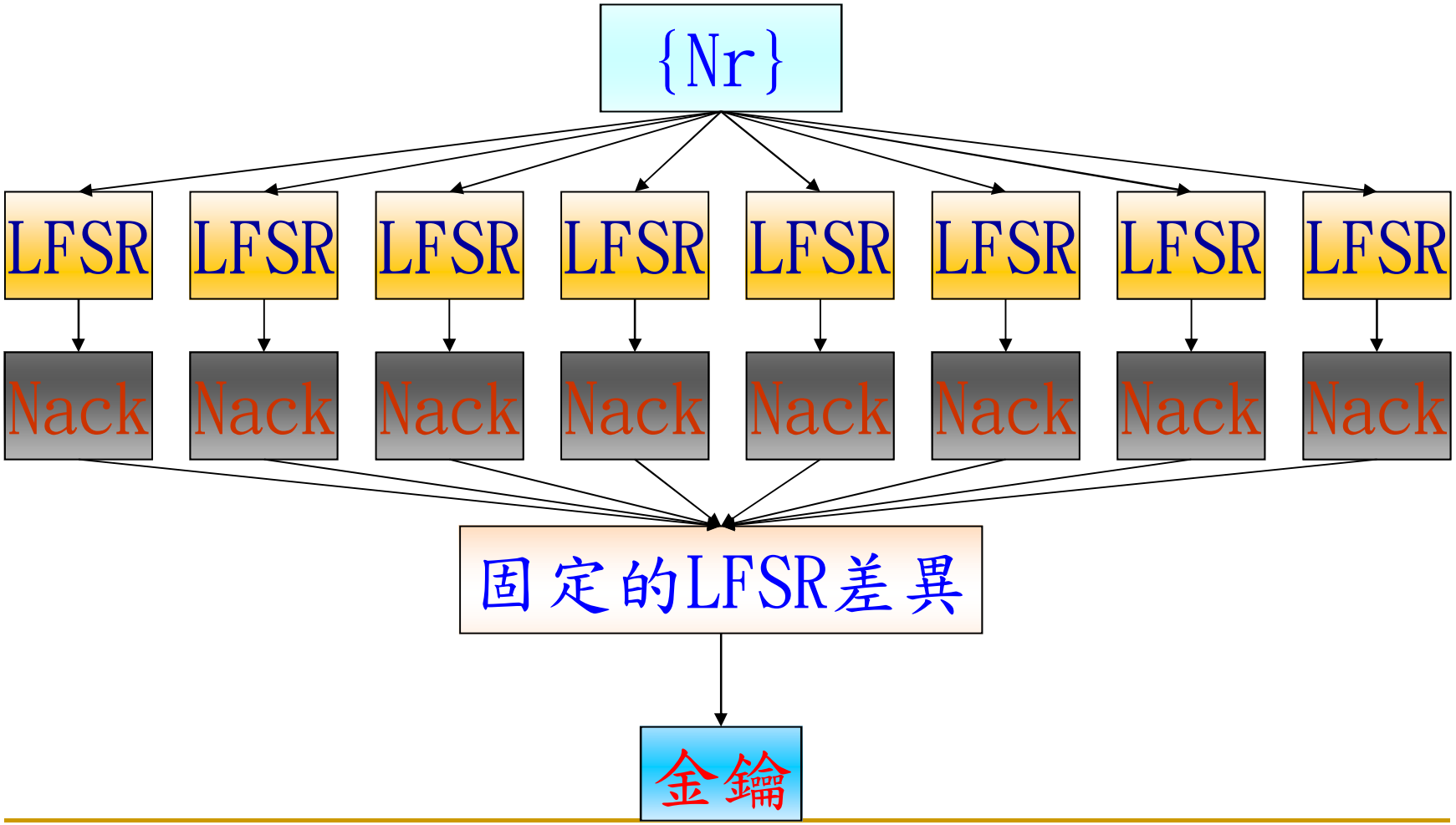
Tag

真的卡片

1. 使用Nt、 {At}、{Halt}、{Ar}計算ks3、ks2
2. 使用{Nr}來計算ks1、Nr。
3. 使用Nr來回推state。
4. 使用UID、Nt來回推Key。

# 第三章　Mifare Classic Card現有攻擊

## {Nr}差分攻擊法



# 第三章　Mifare Classic Card現有攻擊

## 攻擊工具

1.CRYPTO1 lib

主要是針對CRYPTO1加密演算法寫的程式，經由此程式，我們可以解出Key的內容。

UID= 0x9c599b32;

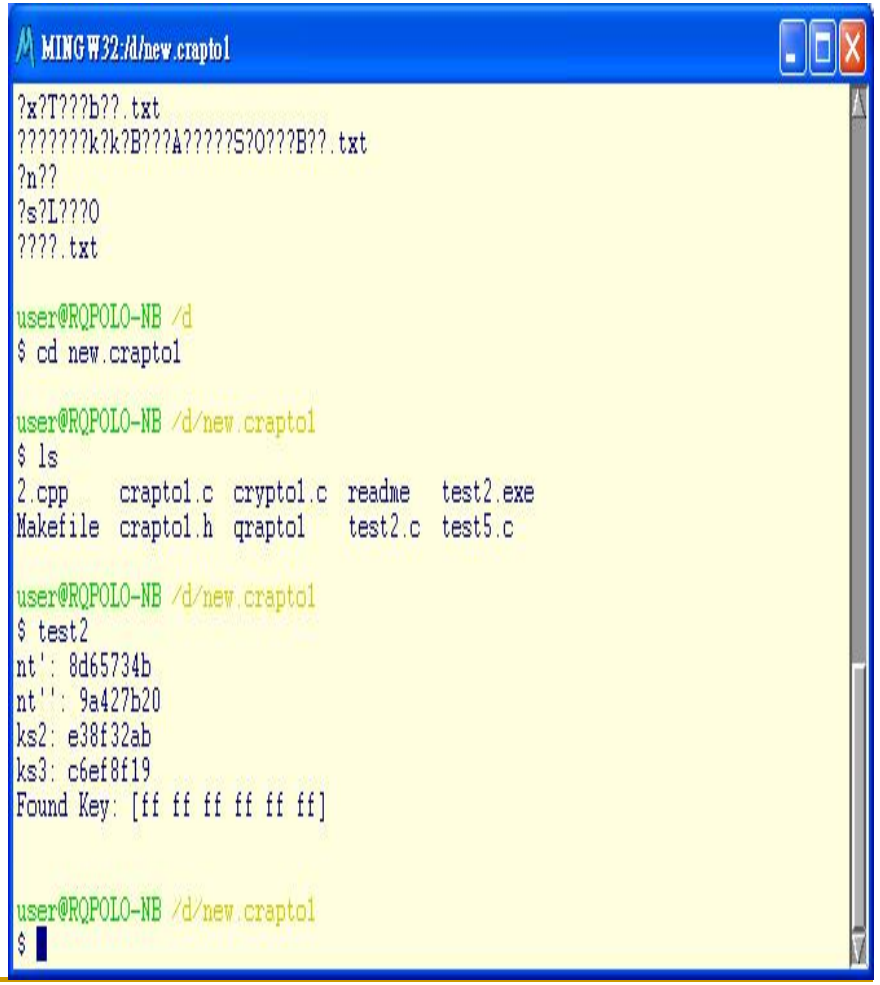
Nt= 0x82a4166c;

{Nr}= 0xa1e458ce;

{Ar}= 0x6eea41e0;

{At}= 0x5cadf439;

計算出Key為ffffffffffff。

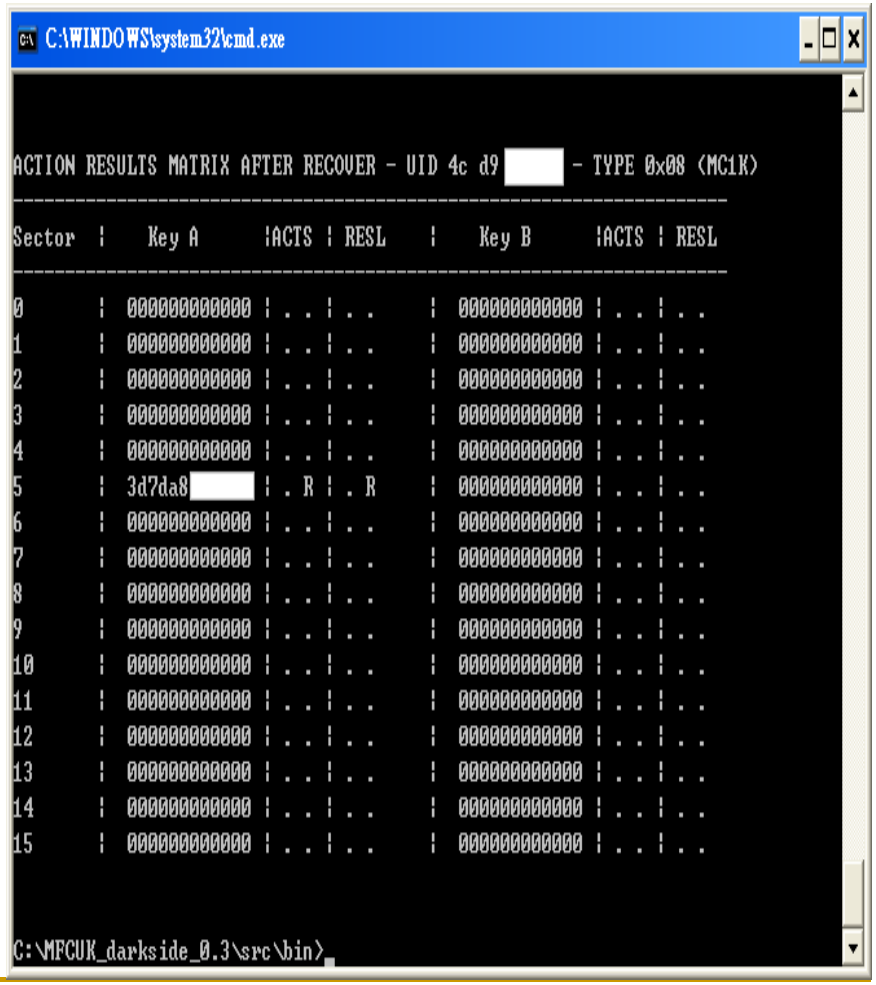


# 第三章　Mifare Classic Card現有攻擊

## 攻擊工具

2.MFCUK：Mifare Classic Universal toolkit

此軟體工具包含相關的範例及各種使用在Libnfc和Crapto-1的工具，該軟體是針對Mifare Classic卡的弱點並參考Wirelessly Pickpocketing a Mifare Classic Card及THE DARK SIDE OF SECURITY BY OBSCURITY這二篇論文的攻擊方式去寫的程式。

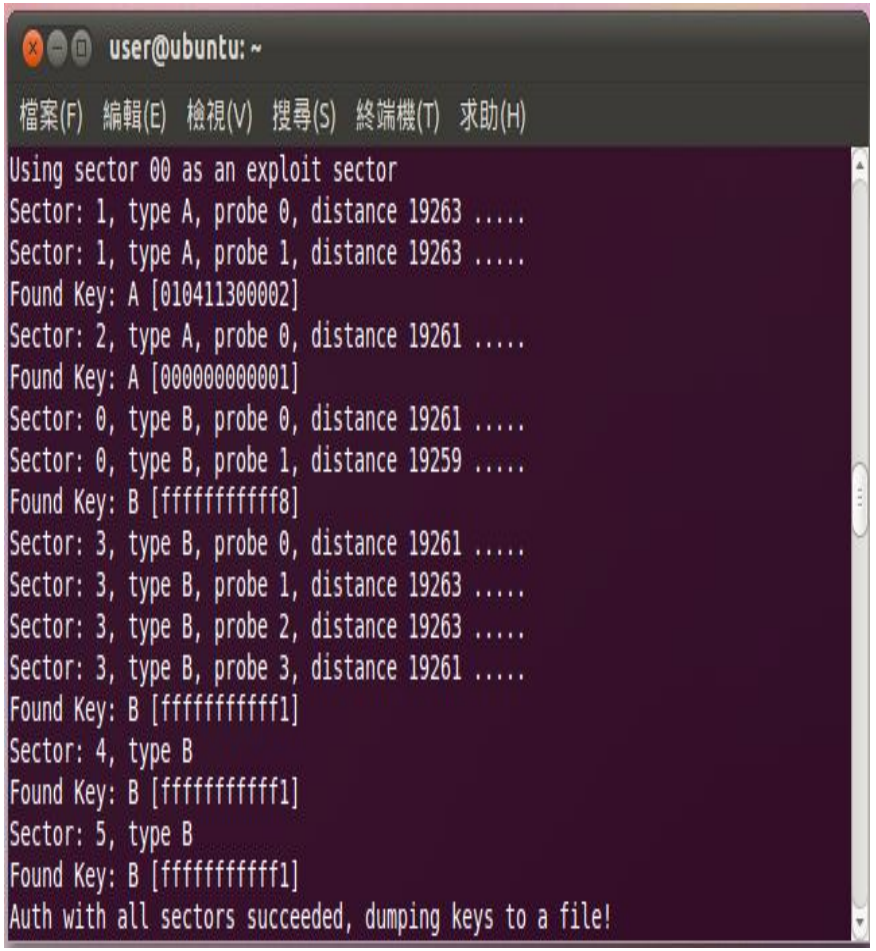


# 第三章　Mifare Classic Card現有攻擊

## 攻擊工具

3.MfOC(Mifare Classic Offline Cracker )

此軟體是NFC的工具軟體之一，它可以還原Mifare Classic卡片的密鑰，利用的原理是Nested Authentication來還原密鑰，只要能知道某一個Sector的密鑰及Nt，它就可以把其他Sector的密鑰還原。



# 第三章　Mifare Classic Card現有攻擊

## 攻擊工具

4.Libnfc[NFC10]

此軟體讓NFC的設備可以做相關的模擬，它可以模擬Mifare Classic卡片認證的過程，軟體為一自由軟體，開放程式碼，可以人員依其需求做修改。

# 第三章　Mifare Classic Card現有攻擊

## 攻擊工具

5.Proxmark 3

可以使用在任何類型的低頻率（125 KHz）或高頻率（13.56 MHz）的RFID設備。它可以模擬成卡片或讀卡機。它也可以竊聽讀卡機和卡片之間的通訊過程。它可以分析接收在空中的訊號作。

# 第三章　Mifare Classic Card現有攻擊

## 攻擊工具

使用ISO 14443的標準的讀卡機。

# 第三章　Mifare Classic Card現有攻擊

## Mifare Classic模擬

### 卡片記憶體說明

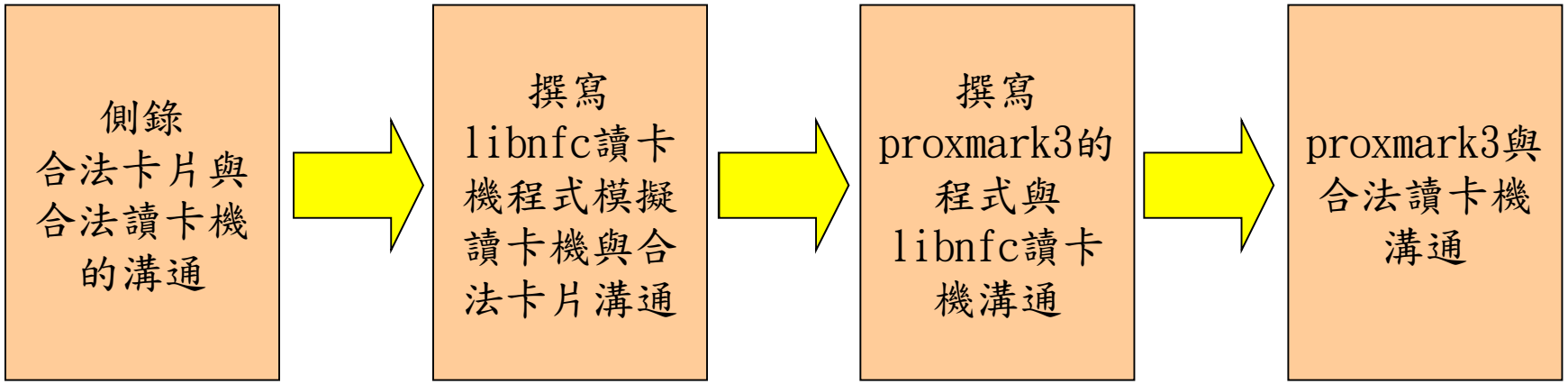
1. Sector 0：卡片基本資料(卡片UID)。
2. Sector 1：卡片認證及管理的資料。
3. Sector 2：卡片的餘額。
4. Sector 3：最近2次交易的紀錄。
5. Sector 4：最近6次使用紀錄1-3。
6. Sector 5：最近6次使用紀錄4-6。
7. Sector 6：公車使用紀錄。
8. Sector 7：最後進出站紀錄。
9. Sector 14：大學學生資料。
10. Sector 15：小額付款每日花費最大金額。
11. Sector 8-13：空白(保留)。

| Sector | Block | Data |
|---|---|---|
| 15 | 60-63 | 小額付款每日花費最大金額 |
| 14 | 56-59 | XX 大學學生資料 |
| 13 | 52-55 | 空白 |
| 12 | 48-51 | 空白 |
| 11 | 44-47 | 空白 |
| 10 | 40-43 | 空白 |
| 9 | 36-39 | 空白 |
| 8 | 32-35 | 空白 |
| 7 | 28-31 | 最後進出站紀錄 |
| 6 | 24-27 | 公車使用紀錄 |
| 5 | 20-23 | 最近6次使用紀錄4-6 |
| 4 | 16-19 | 最近6次使用紀錄1-3 |
| 3 | 12-15 | 最近2次交易的紀錄 |
| 2 | 8-11 | 卡片的餘額 |
| 1 | 4-7 | 卡片認證及管理的資料 |
| 0 | 0-3 | 卡片基本資料(卡片UID) |

# 第三章　Mifare Classic Card現有攻擊

## Mifare Classic模擬

### 模擬流程



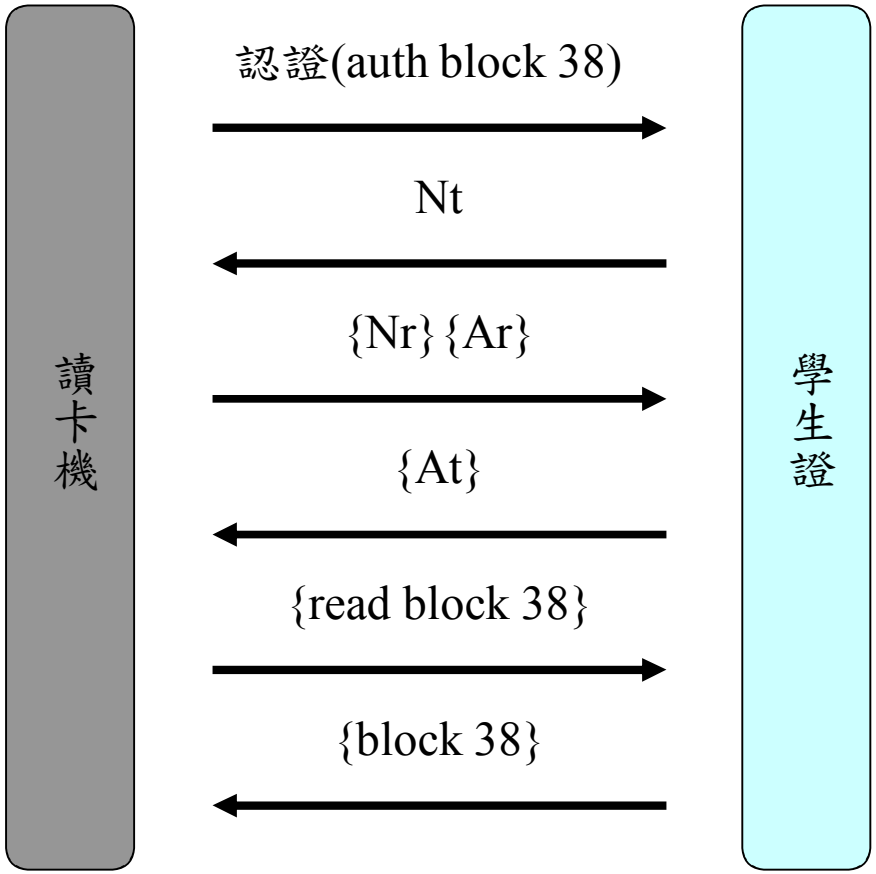
# 第三章　Mifare Classic Card現有攻擊

## Mifare Classic模擬

**學生證模擬**

1. 透過proxmark3設備竊聽(snoop)學生證與學校讀卡機之間溝通的訊息，發現學校門禁讀卡機是讀Sector 14的資料，
2. 用libnfc讀卡機模擬讀卡機與真的卡片測試
3. 再libnfc讀卡機與proxmark3做測試。
4. 最後proxmark3再與學校討論小間讀卡機做實驗。

# 讀卡機與卡片溝通流程(學生證)



# Proxmark3竊聽的內容

| 設備 | Hex | 說明 |
|---|---|---|
| Reader | 26 | |
| TAG | 04 00 | |
| Reader | 93 20 | |
| TAG | 4c d9 ff 7c 16 | |
| Reader | 93 70 4c d9 ff 7c 16 b5 84 | |
| TAG | 08 b6 dd | |
| Reader | 60 38 3e c6 | Auth block 38 |
| TAG | 3c 32 65 f1 | Nt |
| Reader | bb 22 10 d6 c5 f1 be 05 | {Nr}{Ar} |
| TAG | 99! 38 0c! 3d! | {At} |
| Reader | aa 15 31 f6 | {read} |
| TAG | ad 6c! 83! 2d f2 c2! 73 74! 4a c3 7c b2 d9! 48<br>d3 b7 9b 73! | {data} |
| Reader | 14 bc 6b 62 | {halt} |

# Proxmark3運作時設備會亮橘燈

# Proxmark3測試

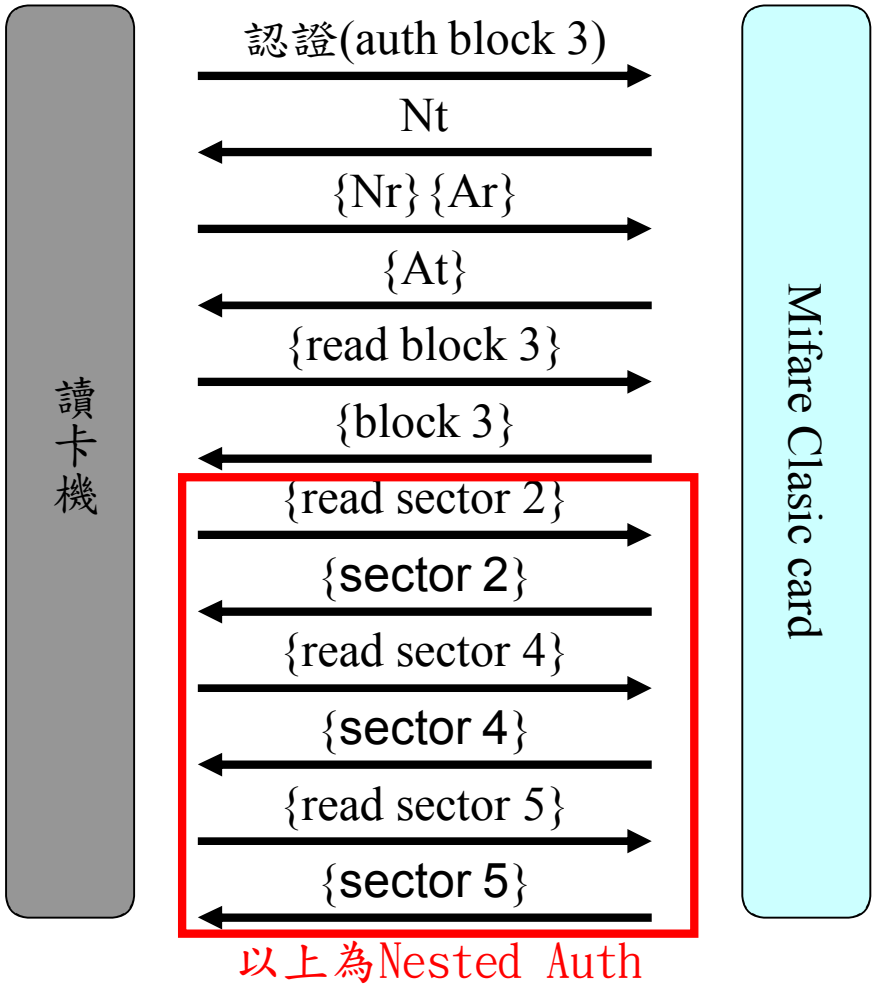
# 第三章　Mifare Classic Card現有攻擊

## Mifare Classic模擬

**四、XXCard模擬(利用餘額查詢機查詢餘額)**

1. 透過proxmark3設備竊聽(snoop)學生證與XX查詢機之間溝通的訊息，發現XX查詢機讀卡機是主要先讀Sector 0的資料(卡片基本資料)，再讀Sector2(餘額)、Sector4(最近6次使用紀錄前3筆)、Sector5(最近6次使用紀錄後3筆) 。
2. 我們寫程式讓libnfc讀卡機可以讀這些Sector的資料。
3. 之後libnfc讀卡機再與proxmark3做測試。
4. 最後proxmark3與XX查詢機做實驗。

# 讀卡機與卡片溝通流程(XXcard)



# 第四章　二代卡的特性

第一節　與一代卡不同處

第二節　目前的攻擊手法

# 第四章　二代卡的特性

## 與一代卡不同處

- Nt 的出現比較不會重覆
  - New PRNG are different from old ones
- Old NACK (0101) ← 舊攻擊的利用重點
  - Parity is correct, Ar is wrong
- Old NACK (0100) ← post authentication
  - Parity is correct, Command is wrong
- New NACK (0000)
  - Always send new NACK (0000 is not sure!)
  - New NACK does not reflect the parity status

# 第四章　二代卡的特性

## 目前的攻擊手法

- 0 sector: 可用舊的 keystream recovery 攻擊
- 其他 sector: 舊攻擊無效
  - {Nr} differential attack
    - Old {NACK}➔{0101}
  - Nested authentication
    - Nt should be predictable

# 第四章　二代卡的特性

## 目前的攻擊手法

- Now we can try following
  - Can we allow those random Nt ?
    - Yes ! But we allow some patterns in pairs
    - We can count the differential Nts based on some little modified states
  - We can construct algebra formulas to represent states and relations
  - Solve them by some existing tools
    - SAT, SMT solver

# 第五章　　自然人憑證

- “「自然人憑證」是可以在網路上作資料交換時，如同網路身分證辨識雙方身分。大家都知道網路很方便。但過去政府無法在網路上為人民服務。主要有以下兩個原因：
  - 在網路上每個人的身分都是很難確認的（假冒者可能會冒名辦土地權狀轉移而造成他人損失）
  - 在網路上傳資料，並不是絕對安全的（例如存在著許多惡意駭客的違法行為）

# 第五章　　自然人憑證

- 就像簽章一樣，只不過是數位的
  - 方法：做出“兩把數學上有相關性的金鑰，具有下列特性：其中一把金鑰可用來做訊息加密，而此加密訊息只有另一把可以解密。就算知道其中一把金鑰要找出另一把金鑰是不可行的。（從計算的角度而言）”
- 怎麼取得我的公開金鑰？
  - “公開金鑰皆須（一般以數位憑證的形式）公開可得”
  - 重點！自然人憑證最重要的目的就是把自然人跟他的公開金鑰綁在一起！

# 第五章　　自然人憑證

- 目前最普遍的方法：RSA（1978年發明）

# 第五章　RSA演算法細節

- (N, e): 公開金鑰
- (N, d): 私密金鑰
- 其中公開金鑰和私密金鑰滿足：
  1. N = pq且p和q皆為質數
  2. de = 1 mod (p - 1)(q - 1)
- 金鑰生成：
  1. 隨機生成差不多大的數，直到生成出質數p,q
  2. e和(p-1)(q-1)互質且e < (p-1)(q-1)
  3. d 可由演算法得到

# 第五章　　RSA加解密

- c: 待簽署文件
- m: 數位簽章
- 簽章

  $m = c^d \bmod N$
- 驗章

  $c = m^e \bmod N$

# 第五章　RSA安全性與質因數分解

- 很明顯的，如果公開金鑰N被分解了，我們就可以找到對應的私密金鑰p和q
- 目前RSA數質因數分解紀錄
  - K. Aoki, J. Franke, A. K. Lenstra, E. Thomé, J. W. Bos, P. Gaudry, A. Kruppa, P. L. Montgomery, D. A. Osvik, H. te Riele, A. Timofeev, and P. Zimmermann. "Factorization of a 768-bit RSA modulus." February 18, 2010.

# 第五章　自然人憑證使用的RSA

- 約226萬把1024-bit RSA公開金鑰
- 約36萬把2048-bit RSA公開金鑰
- 以過去的RSA數質因數分解的進度估計，在2020年左右，利用一台超級電腦應該可以在一年內分解一把1024-bit RSA公開金鑰
  - 好像還蠻安全的

# But!

- A. K. Lenstra, J. P. Hughes, M. Augier, J. W. Bos, T. Kleinjung, and C. Wachter. "Ron was wrong, Whit is right." February 17, 2012.
- "We performed a sanity check of public keys collected on the web."
- "… two out of every one thousand RSA moduli that we collected offer no security."

# 哪裡出錯了？

- 金鑰生成：
  1. 隨機生成差不多大的數，直到生成出質數p, q
  2. N=pq
  3. e和(p-1)(q-1)互質，接著求出d
- $N_1=p_1q$，$N_2=p_2q$ → $q = \gcd(N_1, N_2)$
  - 蝦密，原來國中學的最大公因數真的有用喔！

# 檢驗自然人憑證

- 約226萬把1024-bit RSA公開金鑰
  - 約有100多把被分解
  - 分解比例比Lenstra的結果低
    - 所用的智慧卡品質尚可，不算太差
- 約36萬把2048-bit RSA公開金鑰
  - 全部安全
    - 這是應該的，好嗎？

# 修補方法

- 廢止已被分解之自然人憑證
  - 別懷疑！如果你被通知去換卡，那就是你啦！
- 建立資料庫，檢驗新產生之公開金鑰
- 加強資訊安全教育
  - 如亂數產生器之重要性等等

```
int getRandomNumber()
{
    return 4;  // chosen by fair dice roll.
               // guaranteed to be random.
}
```

# 結語

We must know

We will know

*David Hilbert*