Network Security Homework2

109064518 高聖哲(Kao Sheng-Che)

- **Develop Environment**

Operation System	Ubuntu 16.04
Compiler	GNU C/C++

- **Package introduction**

This homework2 include six different file and it represent individual work

- plainterx_file - 原始檔案 - key_file - 解密檔案 - sbox_file-加密使用16進位宣告檔案

- inv_sbox_file-解密使用16進位宣告檔案 - aes.c - encrypt和decrypt function撰寫 - main.c-主要執行檔案

- 加密,步驟如下

共有 round 0 ~ round N ((N 根據 Key 長度不同而不同 AES-128 (N = 10)

a. round0
   1. AddRoundKey()

b. round1 ~ round (N - 1)
   1. SubBytes()
   2. ShiftRows()
   3. MixColumns()
   4. AddRoundKey()

c. round N
   1. SubBytes()
   2. ShiftRows()
   3. AddRoundKey()

- 解密,步驟如下

round(回合數)以相反方向開始

a. round0
   1. AddRoundKey()
   2. ShiftRows()
   3. SubBytes()

b. round (N - 1)~ round 1
   1. AddRoundKey()
   2. MixColumns()
   3. ShiftRows()
   4. SubBytes()

   c. round 0
   1. AddRoundKey()

- Sbox_file&inv_sbox_file—多項式進行快速加法與乘法處理

1.S-box 製作

-明文區段輸入固定為 128 bits，金鑰長度則可以是 128，192 或 256 bits，加密過程中使用的金鑰是由 Rijndael 金鑰生成方案產生。

-輸入 128bit 依序切成 b0, b1, ... b15 (16 個 8bit 小區塊)，然後重新排列概念圖如下:

[b0, b1, …., b15] -> [b0 b4 b8 b12
b1 b5 b9 b13
b2 b6 b10 b14
b3 b7 b11 b15] 以 column major 來排列

2. Key Expansion function:擴充鑰匙函數產生所有鑰匙

-Input: Key[](主鑰匙), Nr(round), Nb, Nb_k(AES-128(4 block), AES-192(6), AES-256(8))

-Output: Roundkey[], 產生所有子鑰匙 - AES-128(44), 192(52), 256(60),

- key_file- 指定特定的加密鑰匙,鑰匙必須是 128-bit 且是 16 進位的格式

- plaintext_file – 儲需要輸入後的訊息,到時執行 decrpyt.cpp 會呼叫 plaintext 來進行加密

AES 主要有 4 大加密函數

1. Add round key - 輸入資料區塊(128 bit) ⊕ 回合金鑰
2. SubBytes - 透過 S-Box 將 每個 Byte 做轉換
3. ShiftRows - 每 row 中 4 個小區塊進行 circular shift
4. MixColumns - 每 column 的 4 個小區塊進行 linear transform

註:Input、output(plaintext、ciphertext)格式 =>ASCII 格式，AES 每個小區塊為 8 bit，故以軟體實作用 char 的資料型態

Compile and Run the Program

進入109064518_hw2資料夾,並執行以下指令(ubuntu)

手動

```
$ gcc main.c aes.c -o main
$ ./main key_file plaintext_file sbox_file inv_sbox_file
```

自動

```
$ make
```

- The compile result of the Ubuntu terminal

```
jerry86064@jerry86064-VirtualBox:~/Desktop/109064518_hw2$ make
gcc main.c aes.c -o main
./main key_file plaintext_file sbox_file inv_sbox_file
[109064518 Sheng-Che Kao Assignment#2]
==============================
 128-bit AES Decryption Tool
 ==============================

Key Schedule:
78686f74,ab206d65,203e756e,6720d67c,
ce9e7ff1,65be1294,458067fa,22a0b186,
2c563b62,49e829f6,c684ec,2ec8ff8a,
c0404553,89a86ca5,85c022a9,ab8dd23,
f8816331,7129f94,f4e92d3d,5fe1f01e,
10d11fe,61241e6a,95cd3357,ca2cc349,
41232a8a,20734e0,b5ca7b7,7fe6c4fe,
8f3f9158,af38a5b8,1af2a2f,651466f1,
f5c3015,5a3495ad,40c637a2,25d25153,
5bdddd2a,1e94887,412f7f25,64fd2e76,
39ece569,385adee,792ad2cb,1dd7fcbd,


ENCRYPTION PROCESS
------------------

Plain Text:
6e 33 54 77   30 34 6b 5f   35 65 43 75   72 31 54 79

Inital Round (Only AddRoundkey):
-----------
16 5b 3b 03   9b 14 06 3a   15 5b 36 1b   15 11 82 05

Round 1
-----------
3b 52 75 11   6e 96 94 74   08 7b f8 6a   1b 95 e5 7e

Round 2
-----------
ea bf c5 73   54 c7 a3 ab   1d 22 d8 d4   4b 03 12 46

Round 3
-----------
bf a9 28 d2   21 7a dc 9a   9f 01 0c 47   8c fd aa 7f

Round 4
-----------
b1 ed 3b b2   0d 76 09 29   29 7e 19 40   df 09 ab bc

Round 5
-----------
62 b7 fd 9b   8f 9f a7 f7   80 f2 bc 11   1a 12 11 b6

Round 6
-----------
53 c5 43 4d   d0 f4 ed 56   48 1a 13 b6   1e 6b ea e9
```

```
Round 7
-----------
f7 2e 1b 8a   d6 66 5c d0   94 c1 73 e5   6b 80 ca 08

Round 8
-----------
cf b8 3e 71   2f d0 71 5c   97 ff 0d 46   1b c8 fb 00

Round 9
-----------
70 b6 37 ce   bd 62 ab fc   91 eb be 68   6c 2e 5e e7

CipherText (Last Round):
68 46 4b fd   42 ec f5 65   f8 1b 48 7b   4d 99 9e f8
```

Fig.1 Ubuntu terminal result(Encrypt)

```
DECRYPTION PROCESS
------------------

CipherText (Last Round):
68 46 4b fd   42 ec f5 65   f8 1b 48 7b   4d 99 9e f8

Round 8
-----------
8a 70 d7 63   15 16 0f a3   88 e8 b2 4a   af 6c a3 5a

Round 7
-----------
68 33 8f 30   f6 78 74 7e   22 cd af 70   7f 31 4a d9

Round 6
-----------
ed bf 7d 1e   70 a2 87 e3   52 7f 1a b1   72 a6 55 4e

Round 5
-----------
aa db 65 4e   73 89 82 14   cd c9 54 68   a2 a9 5c 82

Round 4
-----------
c8 38 d4 65   d7 f3 62 37   a5 01 e2 a5   9e 55 01 09

Round 3
-----------
08 da fe d2   fd 7c ac b5   db 54 34 b8   64 d3 86 a0

Round 2
-----------
87 c6 61 5a   20 93 c9 8f   a4 7b a6 62   b3 08 0a 48

Round 1
-----------
e2 90 41 f3   9f 21 d9 82   30 2a 9d 92   af 00 22 02

Inital Round (Only AddRoundkey):
-----------
47 fa 05 6b   14 39 13 7b   59 82 e2 80   59 39 6f af

Plain Text:
6e 33 54 77   30 34 6b 5f   35 65 43 75   72 31 54 79

END of Decryption
-----------------

Assertion Passed: State is same as original message

Program ended Successfully
```

Fig.2 Ubuntu terminal result(Decrypt)