암호학 Project (XTS-AES)

컴퓨터공학과
2011037064
박거성

1. Plaintext 의 길이가 32(block 크기의 배수)일때

```
#include <stdio.h>
#include <stdint.h>
#include <string.h>
#include "XTS_AES.h"

typedef unsigned char BYTE;

uint8_t iv[]  = {
    0x33, 0x33, 0x33, 0x33, 0x33, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
};

uint8_t key[] = {
    0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11,
    0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22, 0x22
};

uint8_t plain[] = {
    0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44,
    0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44, 0x44
};

uint8_t cipher[] = {
    0xc4, 0x54, 0x18, 0x5e, 0x6a, 0x16, 0x93, 0x6e, 0x39, 0x33, 0x40, 0x38, 0xac, 0xef, 0x83, 0x8b,
    0xfb, 0x18, 0x6f, 0xff, 0x74, 0x80, 0xad, 0xc4, 0x28, 0x93, 0x82, 0xec, 0xd6, 0xd3, 0x94, 0xf0
};

int main(){
    BYTE tmp[64];

    // 암호화 테스트
    XTS_AES128(plain, tmp, 32, ENC, iv, key); // 파라미터 iv, key 추가
    printf("XTS_AES Encryption: %s\n", 0 == strncmp((char*) cipher, (char*) tmp, 32) ? "SUCCESS!" : "FAILURE!");

    // 복호화 테스트
    XTS_AES128(tmp, cipher, 32, DEC, iv, key); // 파라미터 iv, key 추가
    printf("XTS_AES Decryption: %s\n", 0 == strncmp((char*) tmp, (char*) plain, 32) ? "SUCCESS!" : "FAILURE!");

    return 0;
}
```

➔ 결과화면

```
Appleui-MacBook-Air-4:XTS-AES Skeleton apple$ gcc -o XTS_AES XTS_AES.c test_XTS_
AES.c AES128.c
Appleui-MacBook-Air-4:XTS-AES Skeleton apple$ ./XTS_AES
XTS_AES Encryption: SUCCESS!
XTS_AES Decryption: SUCCESS!
Appleui-MacBook-Air-4:XTS-AES Skeleton apple$
```

2. Plaintext 의 길이가 17 일때

```c
#include <stdio.h>
#include <stdint.h>
#include <string.h>
#include "XTS_AES.h"

typedef unsigned char BYTE;

uint8_t iv[]  = {
    0x9a, 0x78, 0x56, 0x34, 0x12, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
    0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
};

uint8_t key[] = { 0xff, 0xfe, 0xfd, 0xfc, 0xfb, 0xfa, 0xf9, 0xf8, 0xf7, 0xf6, 0xf5, 0xf4, 0xf3, 0xf2, 0xf1, 0xf0,
    0xbf, 0xbe, 0xbd, 0xbc, 0xbb, 0xba, 0xb9, 0xb8, 0xb7, 0xb6, 0xb5, 0xb4, 0xb3, 0xb2, 0xb1, 0xb0
};

uint8_t plain[] = { 0x00, 0x01, 0x02, 0x03, 0x04 ,0x05, 0x06, 0x07, 0x08, 0x09, 0x0a, 0x0b, 0x0c, 0x0d, 0x0e, 0x0f,
    0x10
};

uint8_t cipher[] = { 0x6c, 0x16, 0x25, 0xdb, 0x46, 0x71, 0x52, 0x2d, 0x3d, 0x75, 0x99, 0x60, 0x1d, 0xe7, 0xca, 0x09
    ,
    0xed
};

int main(){
    BYTE tmp[64];

    // 암호화 테스트
    XTS_AES128(plain, tmp, 17, ENC, iv, key); // 파라미터 iv, key 추가
    printf("XTS_AES Encryption: %s\n", 0 == strncmp((char*) cipher, (char*) tmp, 17) ? "SUCCESS!" : "FAILURE!");

    // 복호화 테스트
    XTS_AES128(tmp, cipher, 17, DEC, iv, key); // 파라미터 iv, key 추가
    printf("XTS_AES Decryption: %s\n", 0 == strncmp((char*) tmp, (char*) plain, 17) ? "SUCCESS!" : "FAILURE!");

    return 0;
}
```

➔ 결과화면

```
Appleui-MacBook-Air-4:XTS-AES Skeleton apple$ gcc -o XTS_AES XTS_AES.c test_XTS_
AES.c AES128.c
Appleui-MacBook-Air-4:XTS-AES Skeleton apple$ ./XTS_AES
XTS_AES Encryption: SUCCESS!
XTS_AES Decryption: SUCCESS!
Appleui-MacBook-Air-4:XTS-AES Skeleton apple$
```