**Cryptography Project #1**

소프트웨어학부

2018044720 석예림

1. aes.c 소스코드**Text

Description automatically generatedText

Description automatically generatedText

Description automatically generatedText

Description automatically generatedText

Description automatically generated**

2. 코드 내 함수 설명

* RotWord(uint32\_t word) : KeyExpansion에 필요한 함수로 왼쪽으로 한 byte씩 로테이션 한다. 변수 rw는 word의 마지막 바이트를 저장하여 word를 한 바이트씩 옮기고 rw를 제일 앞에 넣어준다.
* SubWord(uint32\_t word) : KeyExpansion에 필요한 함수로 word 바이트 값을 다른 바이트로 바꿔준다. 한 바이트 씩 이동시키면서 Word바이트 값을 sbox에 넣어 새로운 값(sw)을 얻어낸다.
* KeyExpansion(const uint8\_t \*key, uint32\_t \*roundKey) : 처음 key 값은 그대로 사용해 준다. 첫번째 라운드부터 keyexpansion에 의해 만들어는 라운드키를 사용하기 때문에 16byte마다 스케줄에 의해 새로운 키 값을 만들어 낸다.
* AddRoundKey (uint8\_t \*state, const uint32\_t \*roundKey): 라운드 키를 XOR연산을 사용하여 state에 더하는 함수로 state와 roundkey를 입력받아 각 바이트씩 xor연산 해 준다.
* SubBytes(uint8\_t \*state, int mode) : SubWord함수와 마찬가지로 바이트를 치환하는데 모드에 따라 순방향, 역방향으로 치환한다. 모드가 달라도 치환하는table만 달라지지 동작원리는 바뀌지 않는다.
* ShiftRows(uint8\_t \*state, int mode) : 바이트의 위치를 변경하는 함수이다. 이것도 마찬가지로 mode를 입력받는데 역방향은 순방향의 반대로 돌리면 된다.
  + 1번째 row는 왼쪽으로 1 바이트씩 위치를 변경
  + 2번째 row는 왼쪽으로 2바이트씩 위치를 변경
  + 3번째 row는 왼쪽으로 3바이트씩 위치를 변경
* MixColumns(uint8\_t \*state, int mode) : 기약다항식 x^8 + x^4 + x^3 + x + 1을 사용한 GF(2^8)에서 행렬 곱셈을 수행하는 함수이다. Mode에 따라 encrypt일때는 matrix in GF(2^8)를 사용하고 decrypt일때는 inverse matrix in GF(2^8)을 사용한다. SubByte함수와 마찬가지로 복호화할 때 행렬만 바뀔뿐 동작원리는 달라지지 않기때문에 Temp에 state값을 임시 저장해 두고 mode에 따라 행렬을 선택한다. 그 후 a\*b mod m(x)를 해주고 그 값들을 xor하여 행렬 곱셈을 완성한다.
* Cipher(uint8\_t \*state, const uint32\_t \*roundKey, int mode): 복호화 할 때 equivalent inverse cipher를 사용하여 복호화 순서를 바꾸어 암호화 과정과 같게 구현하였다. AES 암호화 과정은 먼저 plaintext와 key를 AddRoundKey해 주고 SubBytes – ShiftRows – MixColumns - AddRoundKey 순으로 9라운드를 진행하고 마지막 라운드에서는 SubBytes – ShiftRows – AddRoundKey 순으로 진행하게 된다. 매 라운드에서는 KeyExpansion에서 생성된 roundkey를 사용한다.

equivalent inverse cipher는 암호화 과정과 같지만 roundkey를 거꾸로 사용하며 매 라운드마다 roundkey를 MixColumns해 state를 MixColumns한 것과 AddRoundKey 하게된다.

3. 실행 결과

Calendar

Description automatically generated