AES 규격

김동현(wlswudpdlf31@kookmin.ac.kr)

March 8, 2025

Contents

1 AES 규격 개요

AES-128, AES-192 또는 AES-256을 실행하는 일반적인 함수는 Cipher로 나타내며, 그 역함수는 InvCipher로 표시됩니다.

Cipher및 InvCipher알고리즘의 핵심은 상태(state)에 대한 일정한 변환 과정인 라운드(round)의 연속적인 수행입니다. 각 라운드는 라운드 키(round key)라고 하는 추가 입력을 필요로 하며, 라운드 키는 일반적으로 네 개의 워드(word)로 구성된 블록, 즉 16바이트로 표현됩니다.

KeyExpansion이라고 하는 확장 루틴(expansion routine)은 블록 암호화 키를 입력으로 받아 라운드 키를 생성합니다. 구체적으로, **KEYEXPANSION()**의 입력은 단어 배열(key)로 표현되며, 출력은 확장된 단어 배열(w)로 나타납니다. 이 확장된 키 배열을 **키 스케줄(key schedule)**이라고 합니다.

AES-128, AES-192 및 AES-256 블록 암호는 세 가지 측면에서 차이가 있습니다:

- 키 길이
- 라운드 수 (이는 필요한 키 스케줄의 크기를 결정함)
- KeyExpansion내에서의 재귀(recursion) 규격

각 알고리즘에서 라운드 수는 N_r, 키 길이의 워드 수는 N_k로 표시되고, 블록의 워드 수는 N_b로 나타낸다. N_b, N_k, N_r 값은 표 3에 제시되어 있다.

	블록 길이 N_b	키 길이 N_k	라운드 수 N_r
AES-128	4 (128 bits)	4 (128 bits)	10
AES-192	4 (128 bits)	6 (192 bits)	12
AES-256	4 (128 bits)	8 (256 bits)	14

Cipher함수 규격은 1 절을 참고한다. InvCipher함수 규격은 2 절을 참고한다. KeyExpansion 함수 규격은 3 절을 참고한다.

2 Cipher

Algorithm 1 CIPHER

Require: in, N_r, w ▷ $w = \text{KEYEXPANSION}(key)$
Ensure: $state$

```
1: procedure CIPHER(in, N_r, w)
2:     state ← in
3:     state ← ADDROUNDKEY(state, w_[0:16])
4:     for i = 1 to N_r − 1 do
5:         state ← SUBBYTES(state)
6:         state ← SHIFTROWS(state)
7:         state ← MIXCOLUMNS(state)
8:         state ← ADDROUNDKEY(state, w_[16i:16(i+1)])
9:     end for
10:    state ← SUBBYTES(state)
11:    state ← SHIFTROWS(state)
12:    state ← ADDROUNDKEY(state, w_[16N_r:16(N_r+1)])
13:    return state
14: end procedure
```

CIPHER의 입력은 다음과 같다:

- 데이터 입력 in : 16 바이트 선형 배열로 표현되는 블록
- 라운드 수 : 해당 AES 인스턴스에 대한 라운드 수
- 라운드 키

예를 들어, AES-128의 CIPHER함수는 다음과 같이 표현된다.

$$\text{CIPHER}(in, 10, \text{KEYEXPANSION}(key)).$$

CIPHER에서 라운드는 SUBBYTES, SHIFTROWS, MIXCOLUMNS, ADDROUNDKEY 네 가지 바이트 단위 변환을 포함한다. 이 네 가지 변환 규격은 하위 절에서 설명한다.

첫 번째 단계(line 2)는 입력을 상태 배열(state array)에 복사하는 것이며, 이는 섹션 3.4에서 정의된 규칙을 따릅니다. 초기 라운드 키 추가(3행) 후, 상태 배열은 Nr번의 라운드 함수(round function) 변환(412행)을 거칩니다. 마지막 라운드(1012행)는 MIXCOLUMNS() 변환이 생략된다는 점에서 이전 라운드들과 다릅니다. 최종 상태(final state)는 **출력(13행)**으로 반환되며, 이에 대한 설명은 섹션 3.4에 나와 있습니다.

2.1 SubBytes

2.2 Shiftrows

2.3 Mixcolumns

2.4 AddRoundKey

3 InvCipher

4 KeyExpansion