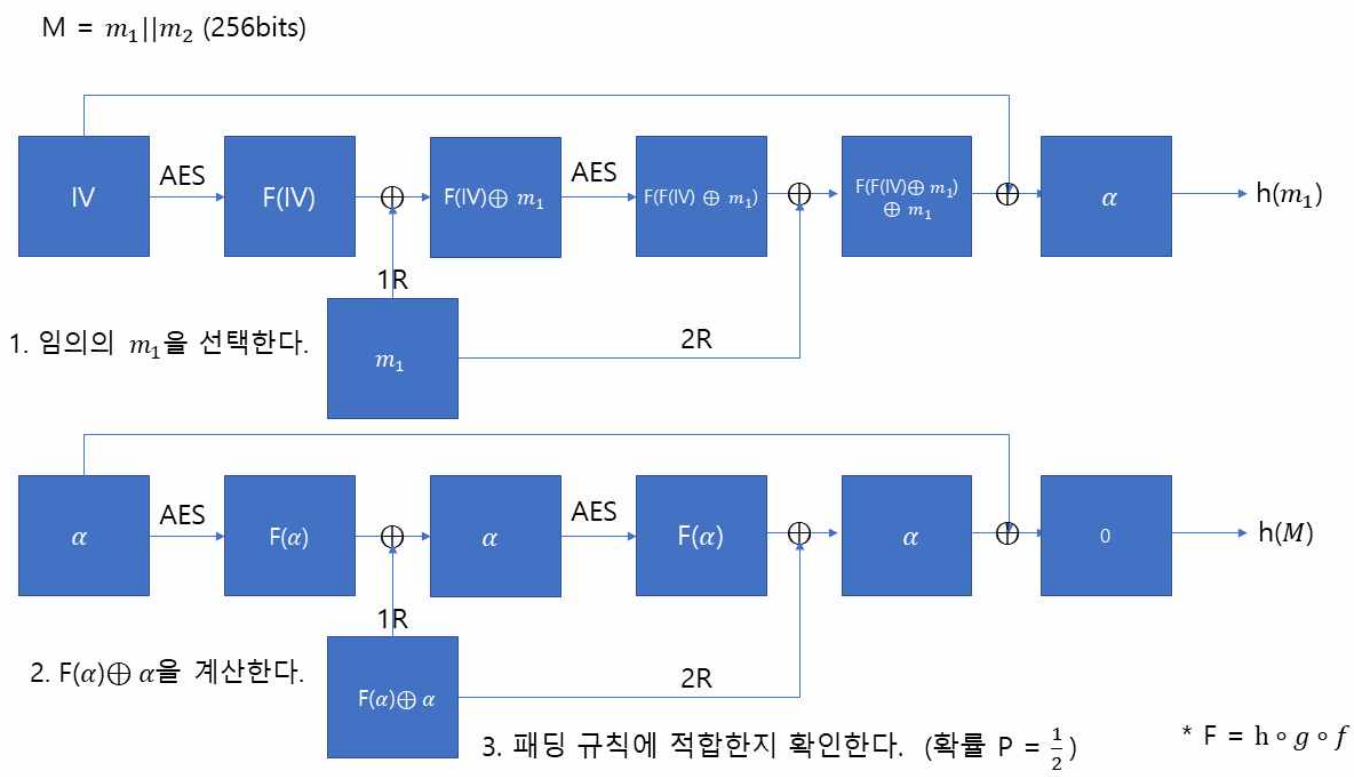
2018 암호분석경진대회 : 1번 문제 답안

1) 알고리즘 제시

(2Round 예시)



메시지 $M = m_1 || m_2$을 선택하는 방법은 다음과 같다.

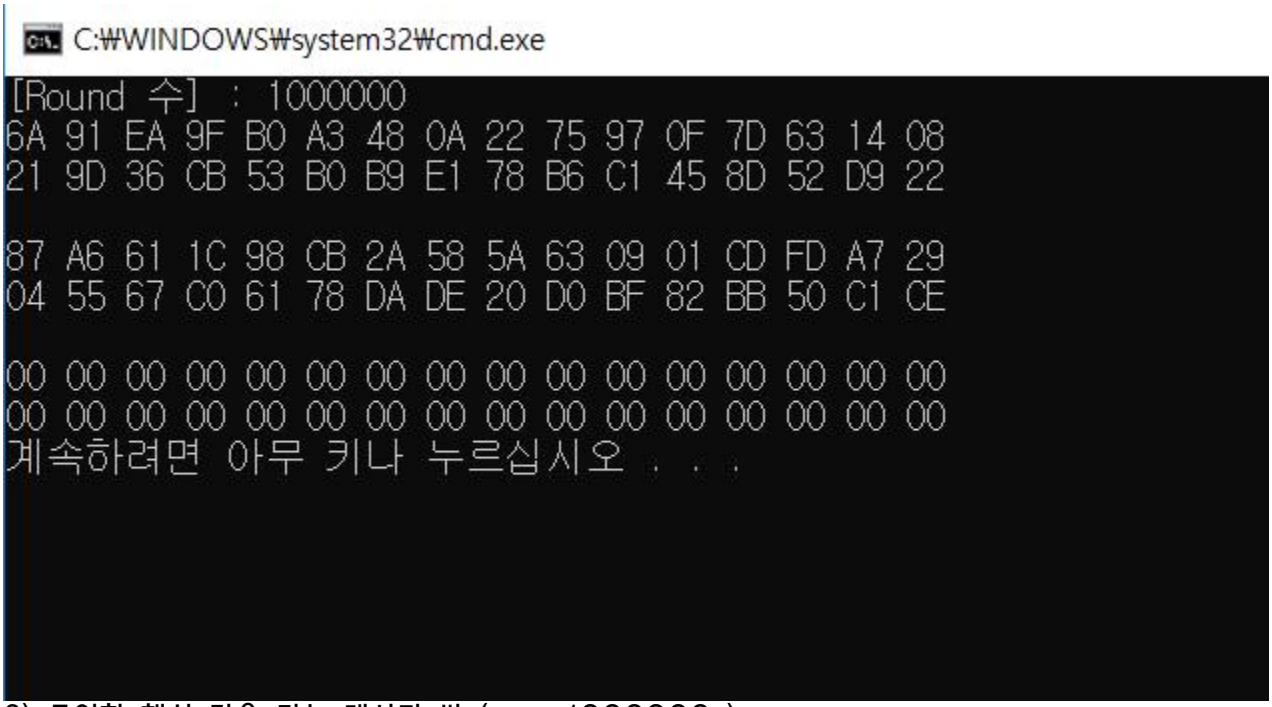
1. m_1을 임의로 선택하여 hash(m_1)을 계산한다.
2. $m_2 = h(g(f(hash(m_1)))) \bigoplus hash(m_1)$을 계산한다. f,g,h는 각각 문제에 주어진 SB, SR, MC을 의미한다.
3. m_2의 마지막 비트가 0인지 확인한다. 만약 마지막 비트가 1일 경우 1번으로 돌아간다.

위와 같은 방법으로 패딩을 포함한 메시지 M을 선택하게 되면, $hash(M)$의 값은 0이 된다.
두 번째 메시지 또한 위와 같은 방법으로 생성하면, $M_1 \neq M_2$이지만 $hash(M_1) = hash(M_2) = 0$을 만족하는 충돌 쌍을 찾을 수 있다.

위의 과정은 임의의 r 라운드(r≥1)에 대해서도 성립한다.

2018 암호분석경진대회 : 1번 문제 답안

[실행 결과]



2) 동일한 해쉬 값을 갖는 메시지 쌍 (r = 1000000)

M1 = 0x6A91EA9FB0A3480A22759700F7D6314080 219D36CB53B0B9E178B6C1458D52D9222

M2 = 0x87A6611C98CB2A585A63090101CDFDA729045567C06178DADE20D0BF82BB50C1CE

M1의 마지막 비트열 10과 M2의 마지막 비트열 10은 패딩이다.

3) 해시값

h(M1) = h(M2) = 0

4) 최대 라운드 수 r

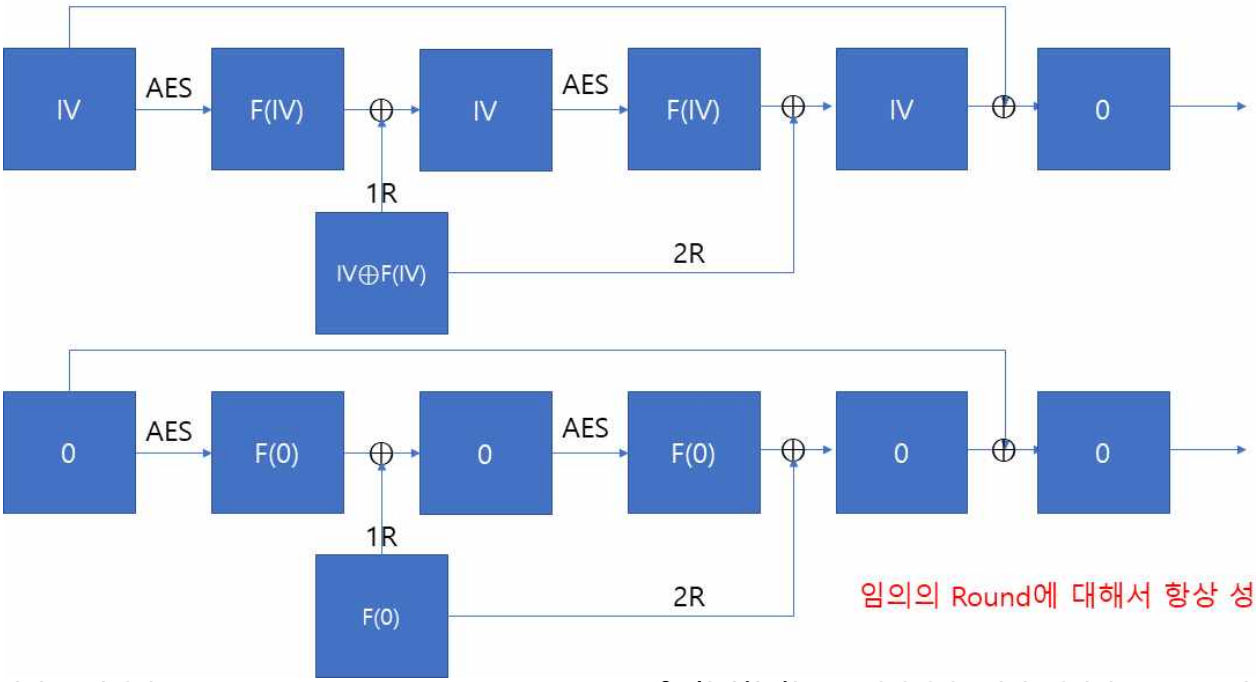
위의 알고리즘을 사용하면 계산 가능한 모든 라운드 r에 대해서 동일한 해쉬 값 0x0을 갖는 메시지 쌍을 찾을 수 있다.

2018 암호분석경진대회 : 1번 문제 답안

알고리즘에 대한 부가 설명

일반적인 해쉬 알고리즘은 일반적으로 임의의 값 x에 대해서 hash(M) = x 를 만족하는 M의 값은 찾기 어렵지만, 이 문제의 경우 x =0일 때 Xor(exclusive OR)을 이용하면 만족하는 M의 값을 찾기가 쉽다. 가장 쉬운 예로는 다음과 같다.

$M_1 = IV \oplus F(IV) \,||\, F(0)$ $\quad h(M_1) = 0$
$M_2 = IV \oplus F(IV)$ $\quad h(M_2) = 0$



위의 그림에서 F는 SubByte, ShiftRow, MixColumn을 합성한 함수를 나타낸다. 위의 메시지 M1, M2의 경우 임의의 Round에 대해서 항상 hash 값을 0으로 출력하는 모습을 보여준다.

하지만 M1 ={{0x2e, 0x1d, 0x98, 0x1, 0x50, 0x3a, 0xe4, 0xb2, 0x10, 0x1b, 0x18, 0x13, 0x84, 0x10, 0x23, 0x15},
{ 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63, 0x63 } }
이므로 $IV \oplus F(IV)$ 와 $F(0)$ 둘 다 마지막 비트가 1이다. 따라서 패딩규칙에 맞지 않는 것을 알 수 있다.

패딩규칙은 $M = m_1 \| m_2$에서 m_2의 마지막 비트가 0이기만 하면 패딩규칙을 따르는 메시지가 된다. 따라서 m_1은 임의로 선택하여 hash(m_1)$\oplus$F(hash(m_1))을 계산하여 마지막 비트가 0인 경우, 그 값을 m_2로 정의하면 패딩 규칙을 따르며 hash(M)=0을 만족하는 메시지 M을 생성할 수 있다.