2023/01/20, 2023 겨울방학 보안기초 세미나

Network Security Essentials

- Chapter_2 대칭 암호와 메시지 기밀성(2) -

손 우 영(wooyoung@pel.sejong.ac.kr)

세종대학교 프로토콜공학연구실

목 차

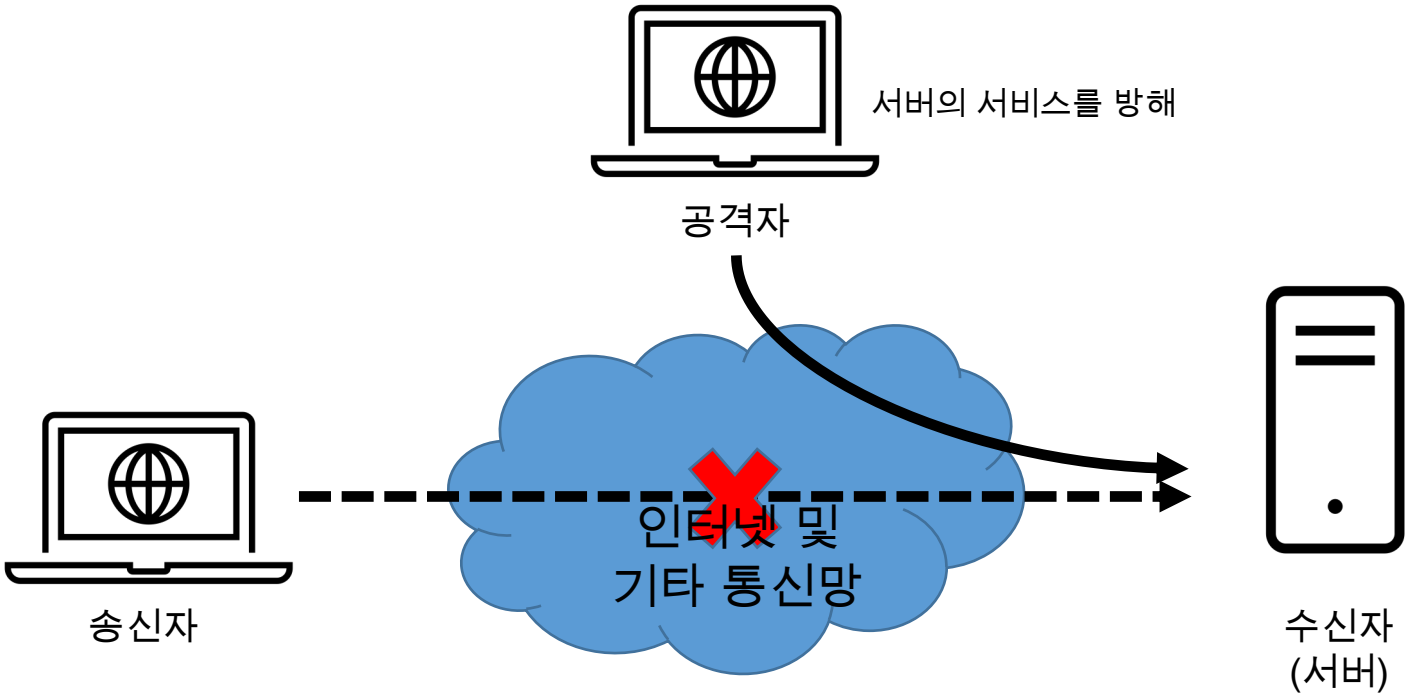
- 보충
- 스트림 암호와 RC4
- 암호 블록 운용 모드

목 차

- 보충
- 스트림 암호와 RC4
- 암호 블록 운용 모드

보충

- 보안 공격의 분류(2/2)
  - 적극적 공격(4/4)
    - 서비스 거부(Denial of Service)
      - 정보 시스템의 데이터를 정당한 사용자가 적절한 시간 내에 사용하는 것을 방해하는 행위
        - e.g., SYN 플러딩 공격, ICMP 플러딩 공격



보충

- 보안 서비스의 분류(2/3)
  - 접근 통제(Access Control)
    - 적절한 권한을 가진 인가자만 특정 정보에 접근할 수 있도록 통제하는 서비스
      - e.g., 접근 권한 여부 증명
  - 부인 봉쇄(Nonrepudiation)
    - 통신의 주체가 통신에 참여했던 사실을 부인하는 것을 방지
      - 수신자는 송신자로부터 송신된 메시지임을 확신함
      - 송신자는 메시지가 수신자에 의해 수신되었음을 확신함
  - 가용성 서비스(Availability Service)
    - 시스템의 가용성을 보장하기 위해 시스템을 보호하는 서비스

보충

- 네트워크 접근 보안 모델
  - 소프트웨어 공격
    - 공격 방법
      - 공격자가 유용한 소프트웨어에다 악성 로직을 잠복시켜 놓고 그 소프트웨어를 사용자가 시스템에 설치하면 시스템이 감염됨
    - 유형
      - 바이러스(Virus)
        - 정의
          - 숙주가 되는 컴퓨터 프로그램을 변형시키고 자신을 복제해 감염시키며 컴퓨터 시스템을 파괴하는 프로그램으로서 자가 전파 불가능
      - 웜(Worm)
        - 정의
          - 스스로 증식하며 네트워크를 통해 연결된 다른 컴퓨터에 침투해 감염시키는 프로그램으로서 자가 전파 가능

목 차

- 보충
- 스트림 암호와 RC4
- 암호 블록 운용 모드

스트림 암호와 RC4

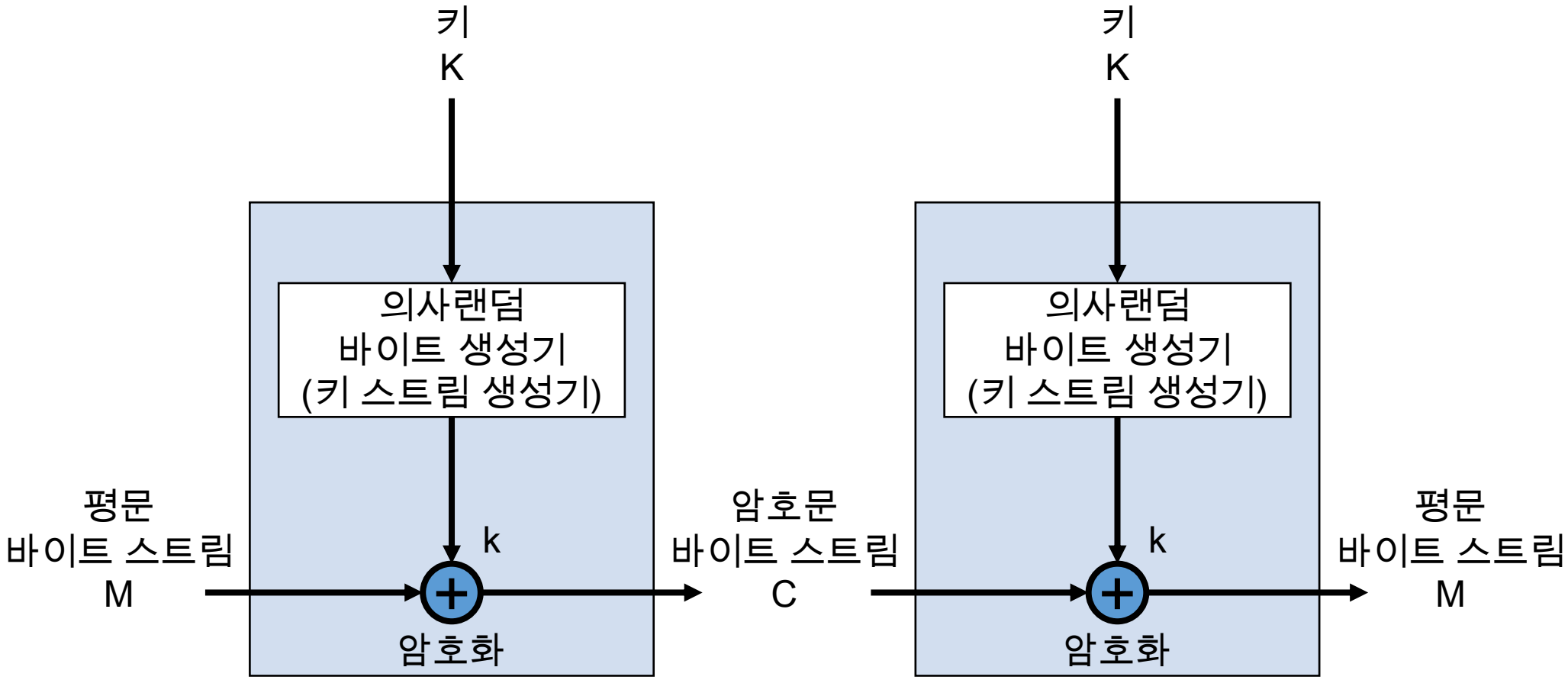
- 스트림 암호(Stream Cipher)
  - 정의
    - 입력되는 요소를 연속적으로 처리하는 대칭키 암호 구조
  - 구조
    - 의사난수 비트 생성기에 키를 입력하여 키 스트림이 출력
    - 이진 평문 스트림과 이진 키 스트림의 XOR 연산으로 암호문 생성
    - 암호화에 사용된 키 스트림과 암호문을 XOR 연산하여 복호화

스트림 암호와 RC4

- 스트림 암호(Stream Cipher)
  - 특징
    - 블록 암호보다 속도가 빠름
    - 블록 암호와 달리 실시간 처리 가능
    - 두 개 이상의 평문을 동일한 키를 사용해서 암호화하는 경우, 암호 해독이 단순해짐
  - 종류
    - RC4

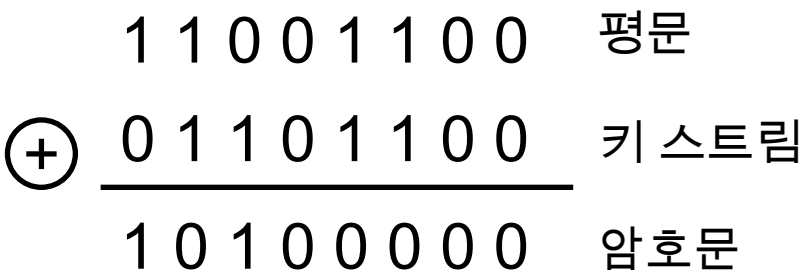
스트림 암호와 RC4

- 스트림 암호(Stream Cipher)
  - 구조

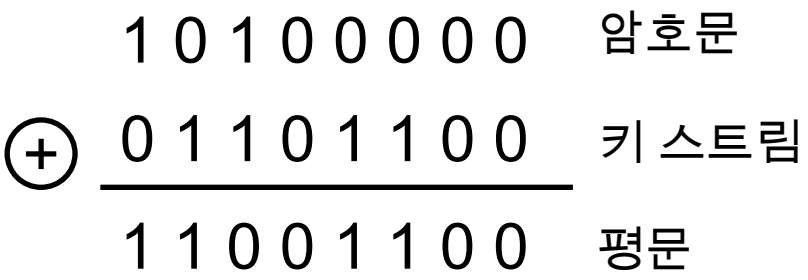


스트림 암호와 RC4

- 스트림 암호(Stream Cipher)
  - 구조
    - 키 스트림을 이용한 XOR 연산



<암호화>



<복호화>

스트림 암호와 RC4

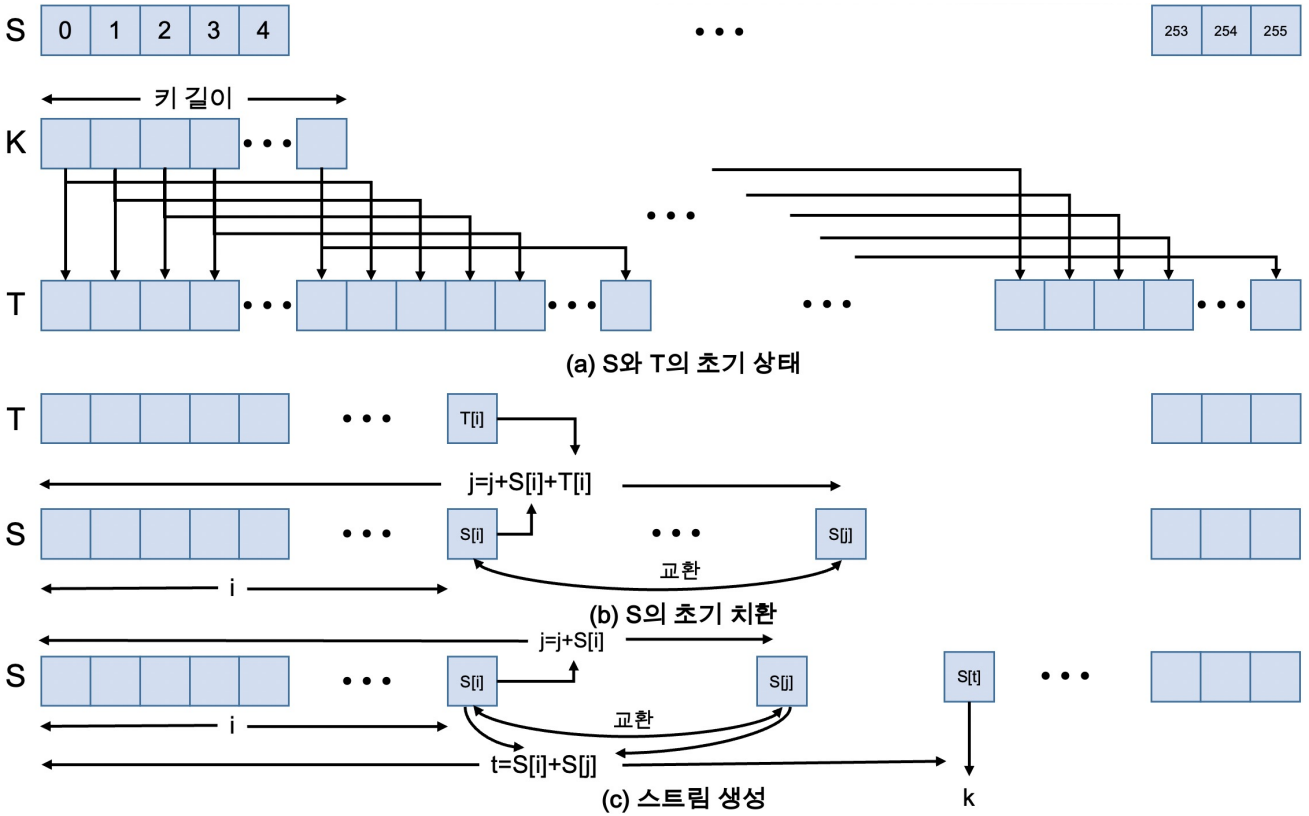
- 스트림 암호(Stream Cipher)
  - 설계 시 고려사항
    - 암호열의 주기가 커야 함
    - 키 스트림은 진성 난수 스트림의 특성에 근사해야 함
    - 키의 길이가 충분히 길어야 함
      - 최소 128비트 이상

스트림 암호와 RC4

- 스트림 암호(Stream Cipher)
  - RC4 알고리즘
    - 정의
      - 바이트 단위로 작동되도록 만들어진 다양한 크기의 키를 사용하는 스트림 암호
    - 특징
      - 사용되는 알고리즘은 랜덤 치환에 기초하여 만들어짐
      - 블록 암호보다 빠름
      - WEP(Wired Equivalent Privacy) 프로토콜과 WPA(WiFi Protected Access) 프로토콜에서 사용

스트림 암호와 RC4

- 스트림 암호(Stream Cipher)
  - RC4 알고리즘
    - 과정
      1. S와 T의 초기화
      2. S의 초기 치환
      3. 스트림 생성

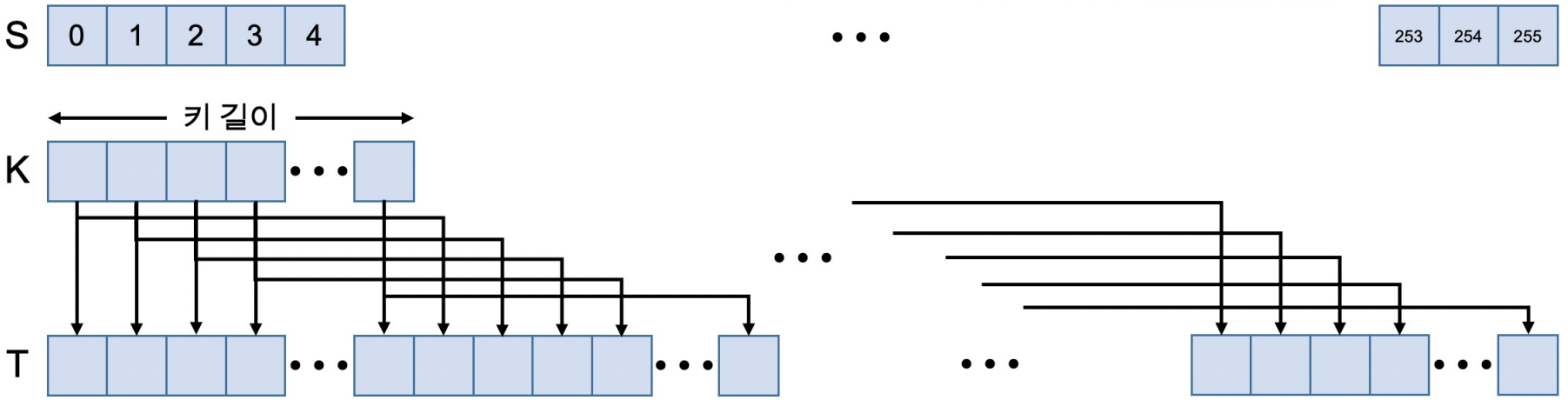


스트림 암호와 RC4

- 스트림 암호(Stream Cipher)
  - RC4 알고리즘
    - 과정(1/3)
      - S와 T의 초기화
        - S는 0부터 255까지 오름차순 정렬
        - T가 채워질 때까지 K값 저장

```
/*Initialization*/
for i = 0 to 255 do
    S[i] = i;
    T[i] = K[i mod keylen];
```

T[i] = K[i % strlen(K)]

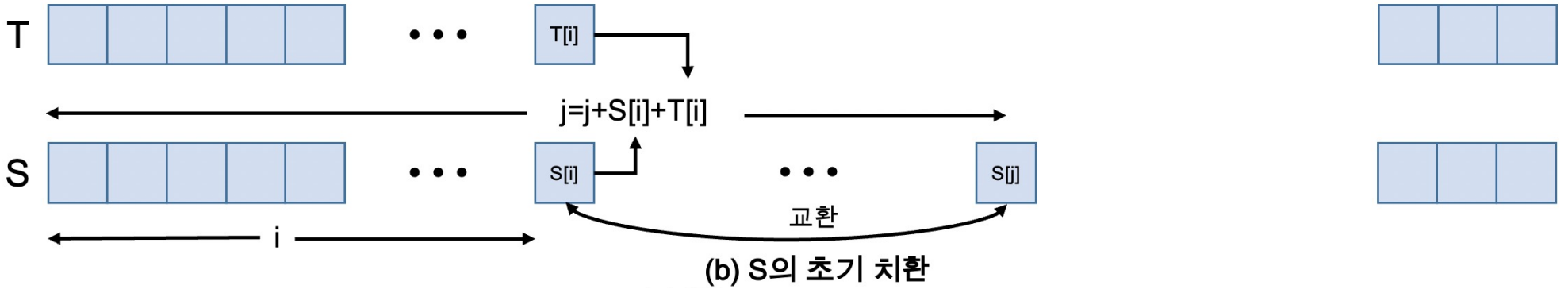


(a) S와 T의 초기 상태

스트림 암호와 RC4

- 스트림 암호(Stream Cipher)
  - RC4 알고리즘
    - 과정(2/3)
      - S의 초기 치환
        - T[i]를 이용한 구조에 따라 S[i]를 S의 다른 바이트와 교환

```
/*Initial Permutation of S*/
j = 0;
for i = 0 to 255 do
    j = (j + S[i] + T[i]) mod 256;
    Swap(S[i], S[j]);
```

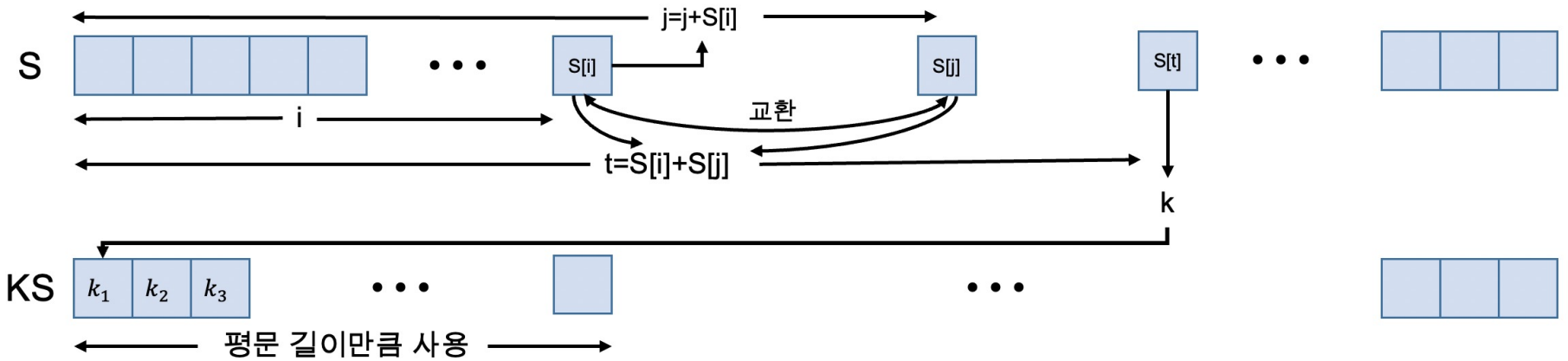


(b) S의 초기 치환

스트림 암호와 RC4

- 스트림 암호(Stream Cipher)
  - RC4 알고리즘
    - 과정(3/3)
      - 스트림 생성
        - i 값에 따른 j 값 계산
        - S[i]와 S[j]의 위치 교환
        - t 값에 따른 키 스트림 생성

```
/*Stream Generation*/
i, j = 0
while(true)
     i = (i + 1) mod 256;
     j = (j + S[i]) mod 256;
     Swap(S[i], S[j]);
     t = (S[i] + S[j]) mod 256;
     k = S[t];
```



(c) 스트림 생성

스트림 암호와 RC4

- 스트림 암호(Stream Cipher)
  - RC4 알고리즘
    - 강도
      - WEP 프로토콜의 취약점
        - RC4에 입력으로 사용되는 키의 생성 방법의 문제
      - 바이트 추측 공격 가능
        - 내부 상태의 바이트를 계속해서 업데이트하기 때문에 내부 상태의 값이 공격자에게 노출되면 바이트 추측 공격을 통해 다음 바이트 값 예측 가능

목 차

- 보충
- 스트림 암호와 RC4
- 암호 블록 운용 모드

암호 블록 운용 모드

- 정의
  - 하나의 키를 사용하여 블록 암호를 반복적으로 이용하는 절차
- 종류
  - 전자 코드북 모드(ECB, Electronic Codebook Mode)
  - 암호 블록 체인 모드(CBC, Cipher Block Chaining Mode)
  - 암호 피드백 모드(CFB, Cipher Feedback Mode)
  - 출력 피드백 모드(OFB, Output Feedback Mode)
  - 카운터 모드(CTR, Counter Mode)

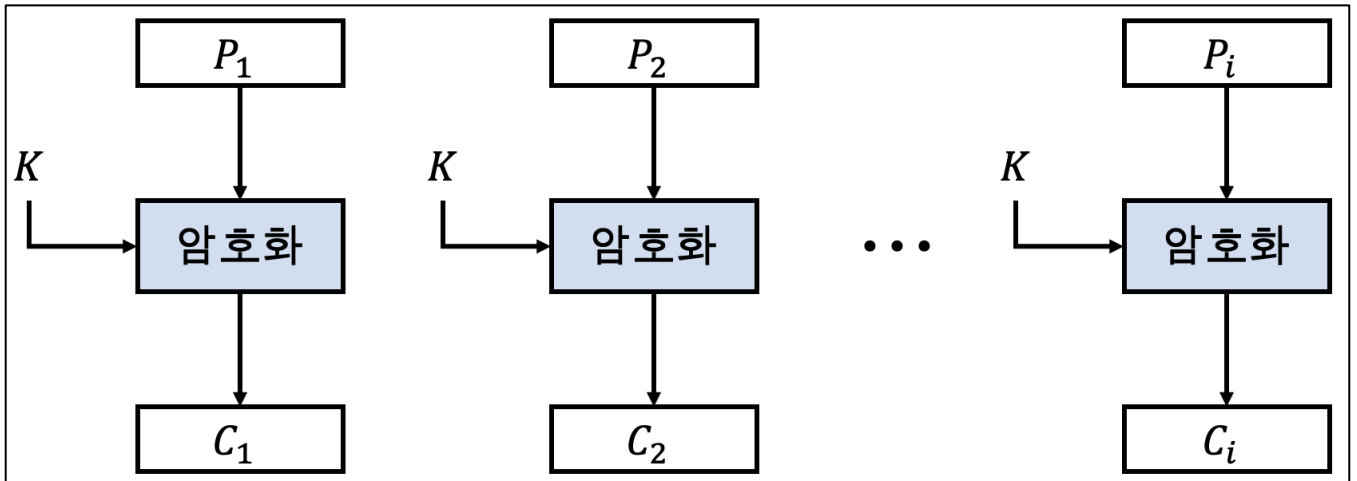
암호 블록 운용 모드

- 종류(1/5)
  - 전자 코드북 모드(ECB, Electronic Codebook Mode)
    - 정의
      - 평문을 일정한 크기의 블록으로 나누고 각 블록을 동일한 키로 암호화하는 방식
    - 특징
      - 운용 모드 중에서 가장 간단한 모드
      - 패딩이 필요함

암호 블록 운용 모드

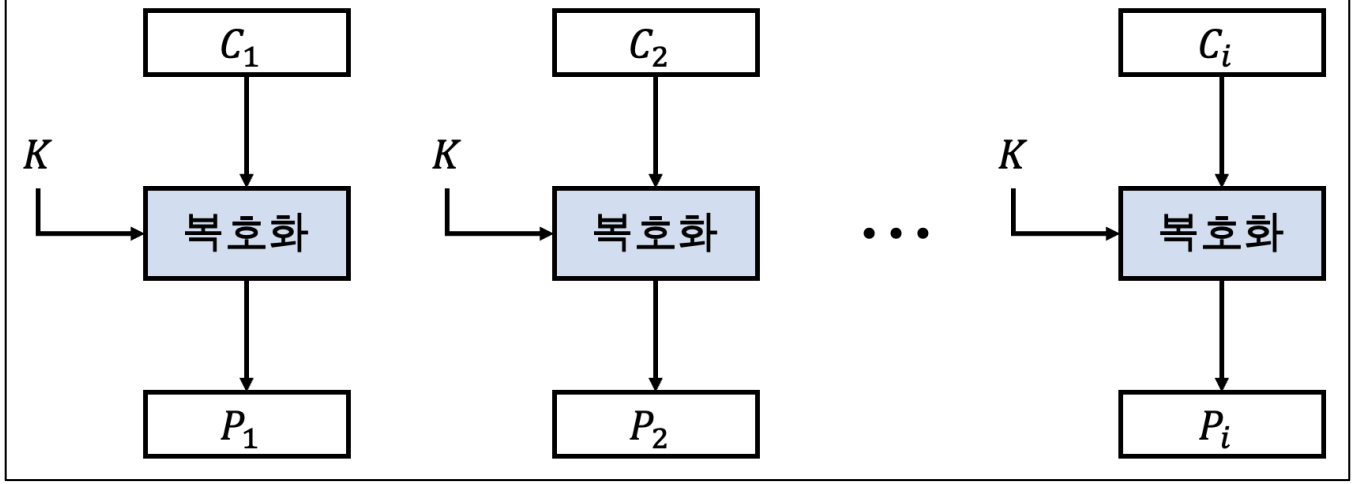
- 종류(1/5)
  - 전자 코드북 모드(ECB, Electronic Codebook Mode)
    - 암호화

$C_i = E(K, P_i)$



    - 복호화

$P_i = D(K, C_i)$



암호 블록 운용 모드

- 종류(1/5)
  - 전자 코드북 모드(ECB, Electronic Codebook Mode)
    - 장점
      - 암호문 블록의 독립성
        - 병렬처리 가능
    - 단점
      - 블록 단위의 패턴이 유지됨
        - 평문에서 같은 값을 갖는 블록은 대응되는 암호문 블록도 같은 값을 가짐
      - 블록 간의 독립성은 키를 알지 못해도 특정 암호문을 변조할 수 있는 기회 제공
        - 암호문의 순서가 변조되어도 동일한 키를 사용하므로 복호화가 가능하기 때문

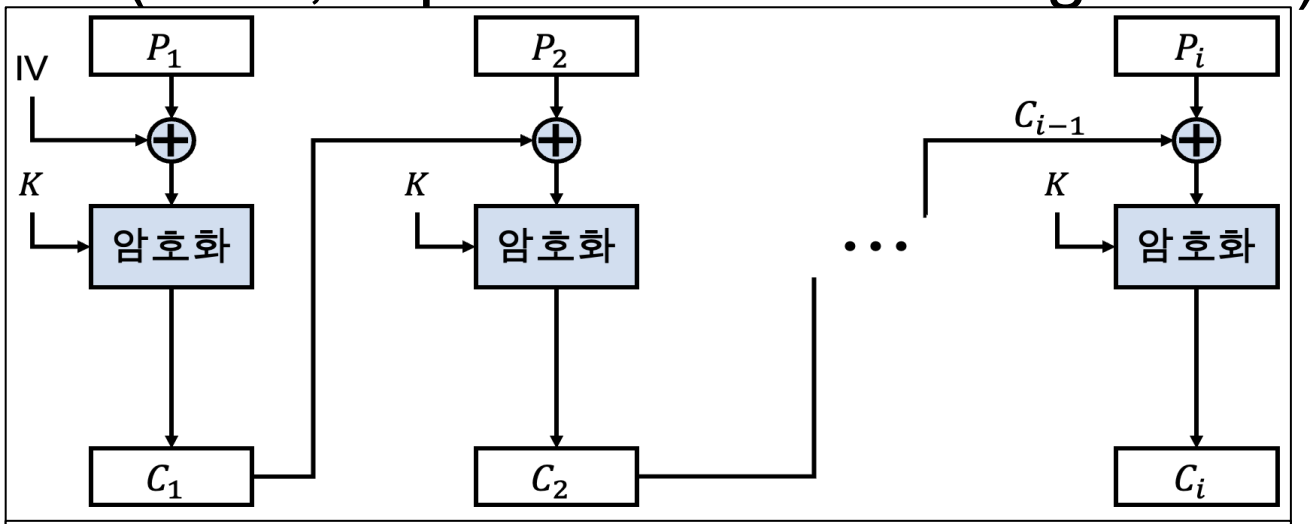
암호 블록 운용 모드

- 종류(2/5)
  - 암호 블록 체인 모드(CBC, Cipher Block Chaining Mode)
    - 정의
      - 현재의 평문 블록과 바로 직전의 암호 블록을 XOR 한 결과를 알고리즘의 입력으로 사용하여 암호화하는 방식
    - 특징
      - 초기화 벡터(IV, Initialization Vector) 사용
      - 체인 구조를 이루며 각 블록이 이전의 암호화 블록의 영향을 받음

암호 블록 운용 모드

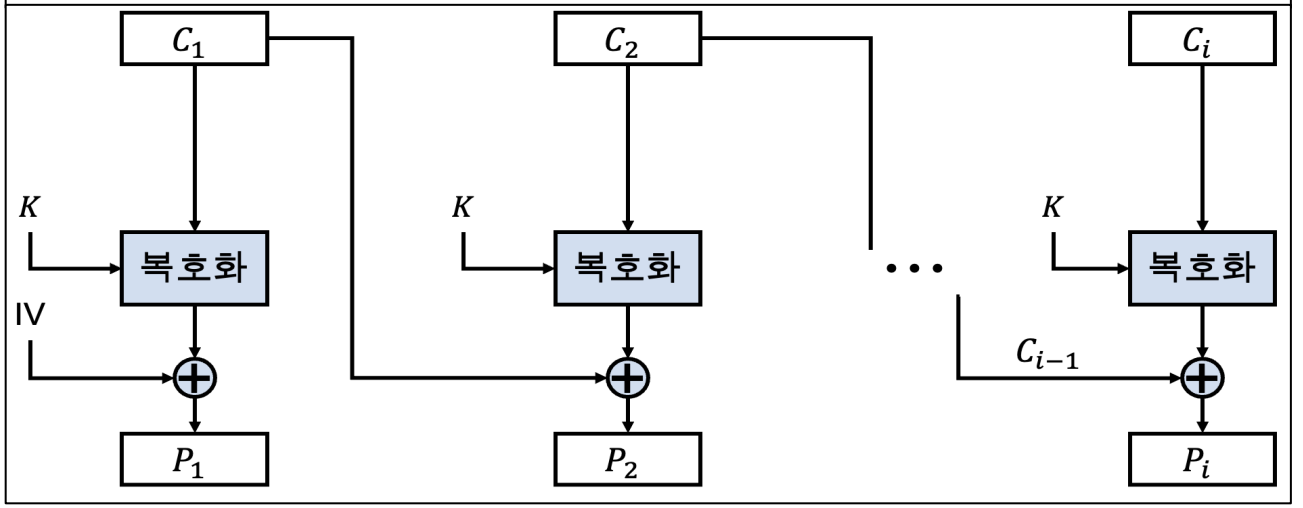
- 종류(2/5)
  - 암호 블록 체인 모드(CBC, Cipher Block Chaining Mode)
    - 암호화

$$C_0 = IV$$
$$C_i = E(K, [C_{i-1} \oplus P_i])$$



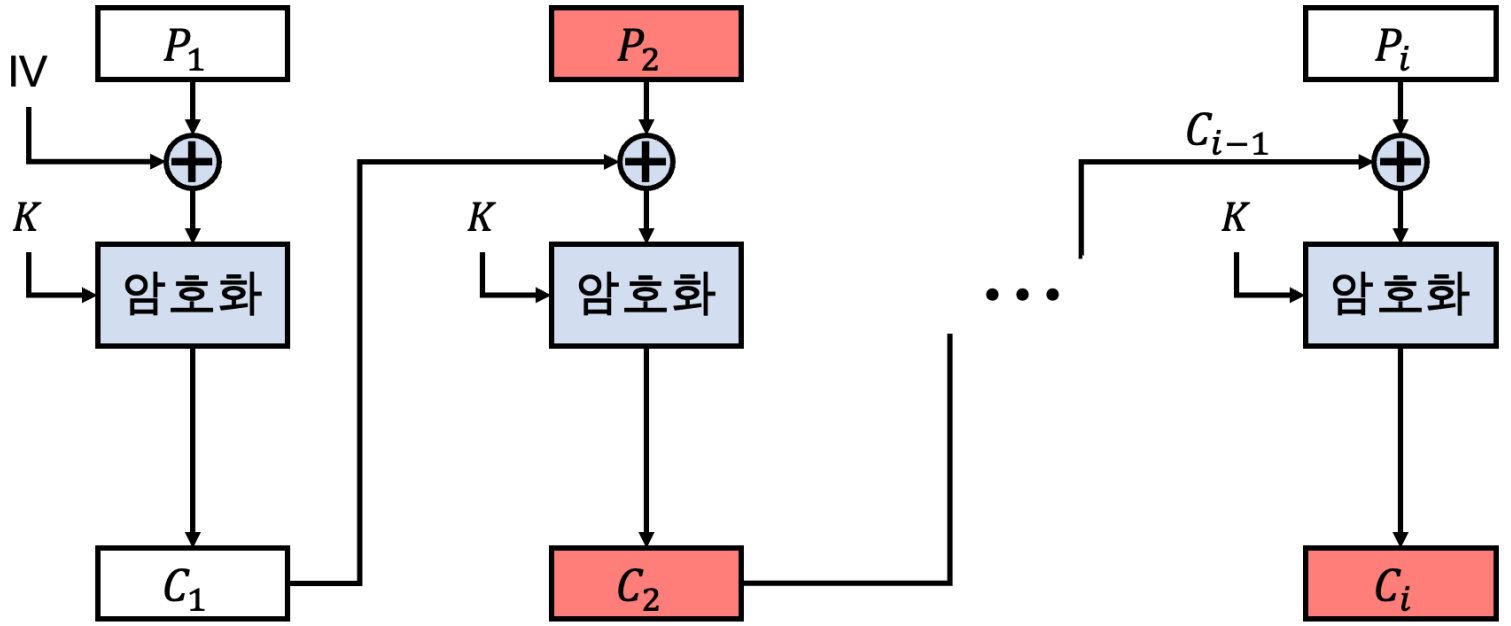
    - 복호화

$$C_0 = IV$$
$$P_i = D(K, C_i) \oplus C_{i-1}$$



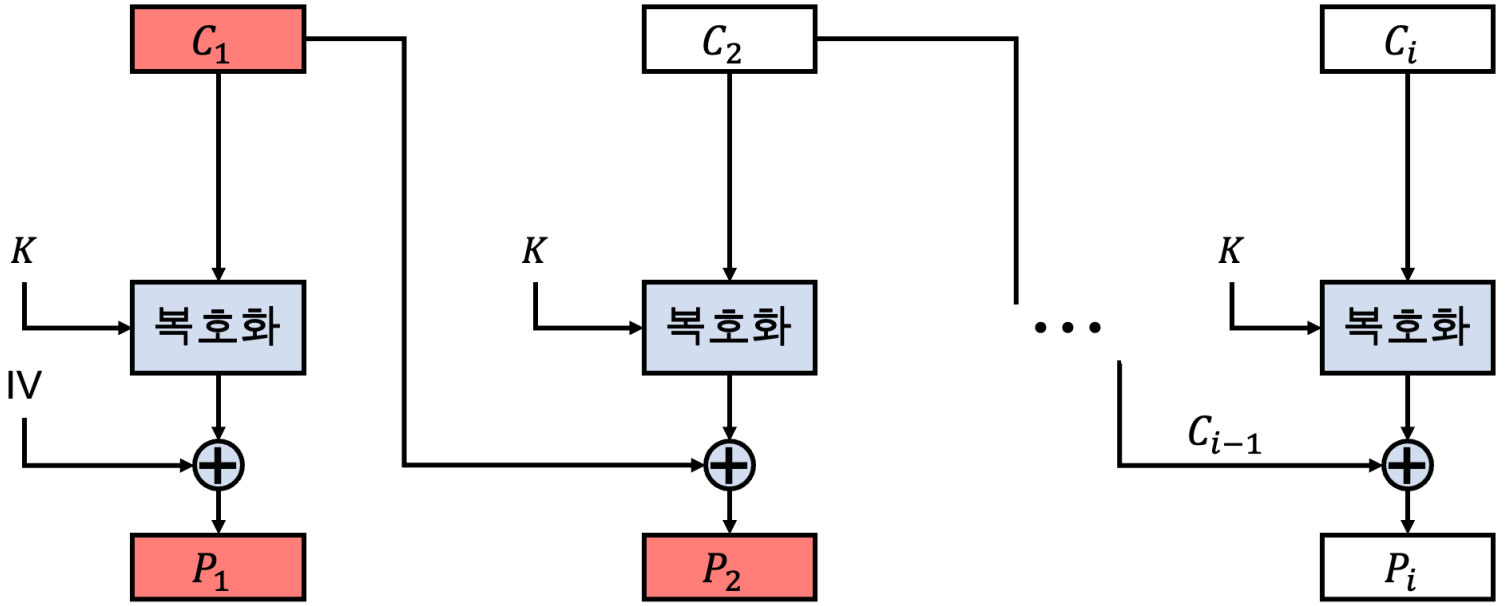
암호 블록 운용 모드

- 종류(2/5)
  - 암호 블록 체인 모드(CBC, Cipher Block Chaining Mode)
    - 오류 확산
      - 암호화 과정
        - 한 평문 블록에 오류가 발생하면 현재 암호문 블록 이후 전체 암호문 블록에 영향



암호 블록 운용 모드

- 종류(2/5)
  - 암호 블록 체인 모드(CBC, Cipher Block Chaining Mode)
    - 오류 확산
      - 복호화 과정
        - 한 암호문 블록에 오류가 있을 때, 현재 평문 블록과 다음 평문 블록에만 영향



암호 블록 운용 모드

- 종류(2/5)
  - 암호 블록 체인 모드(CBC, Cipher Block Chaining Mode)
    - 장점
      - 한 메시지 내에서 블록 단위의 패턴이 유지되지 않음
        - 동일한 평문 블록들은 서로 다른 암호화 블록으로 암호화 됨
    - 단점
      - 동일한 초기 벡터를 사용하는 환경에서 두 개의 메시지가 동일한 경우 대응되는 암호문은 동일함
      - 블록 간의 연관성 존재
        - 병렬 처리 불가능
        - 랜덤하게 선택된 파일을 암호화하거나 복호화할 때 사용 불가능

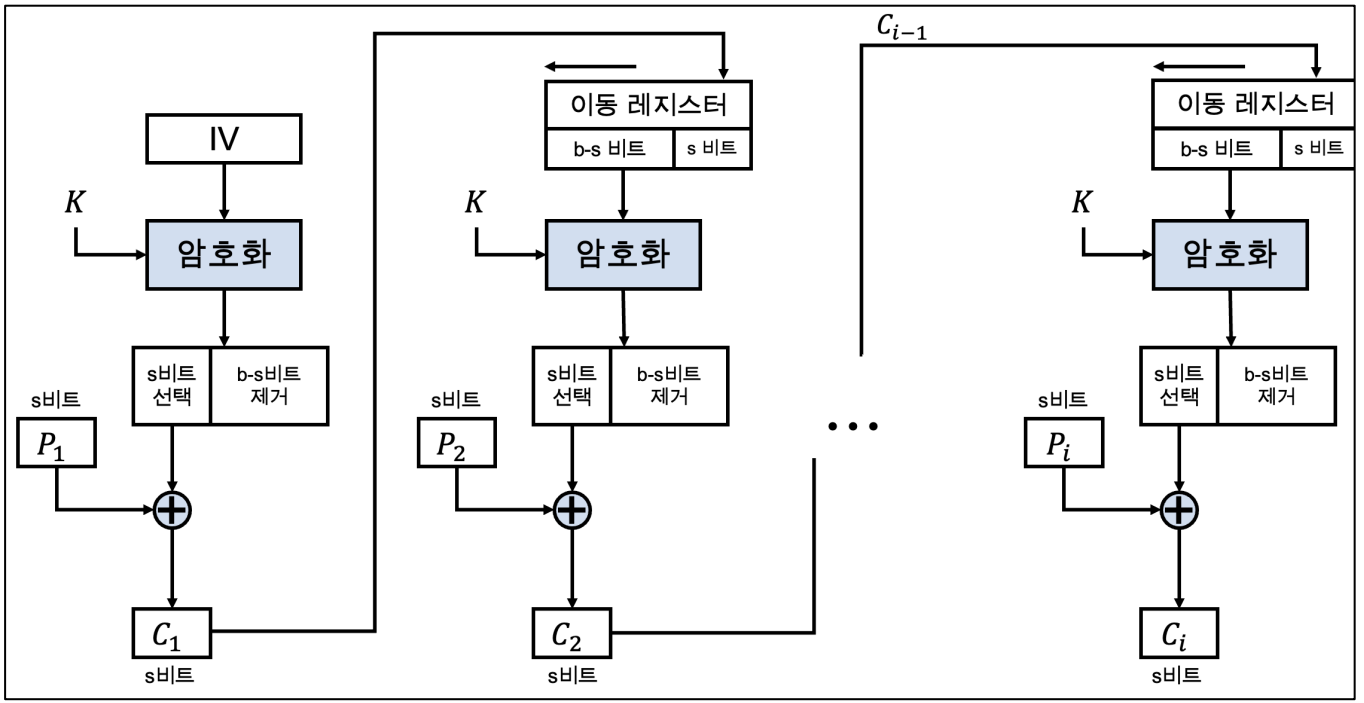
암호 블록 운용 모드

- 종류(3/5)
  - 암호 피드백 모드(CFB, Cipher Feedback Mode)
    - 정의
      - 암호 알고리즘에 이전 암호문 블록을 입력하여 얻어낸 키 스트림을 평문 블록과 XOR하여 암호화하는 방식
    - 특징
      - 초기화 벡터(IV: Initialization Vector) 사용
      - 암호 피드백 모드를 이용하면 블록 암호를 스트림 암호를 바꿀 수 있음
        - 각 문자가 암호화 되는 즉시 전송 가능
      - 한 문자나 한 비트와 같은 작은 크기의 블록을 암호화하는데 사용
      - 암호화 함수만 사용됨

암호 블록 운용 모드

- 종류(3/5)
  - 암호 피드백 모드(CFB, Cipher Feedback Mode)
    - 암호화

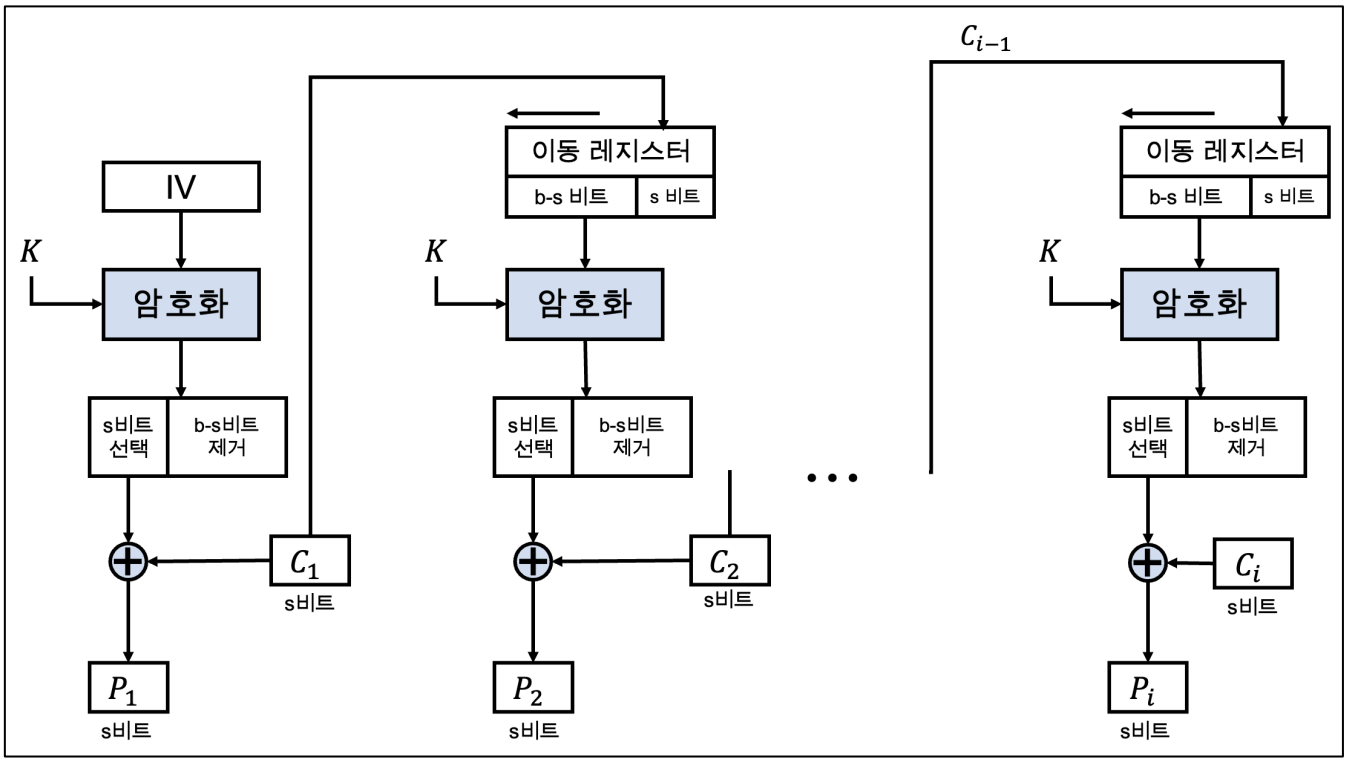
$$C_1 = P_1 \oplus S_S[E(K, IV)]$$



암호 블록 운용 모드

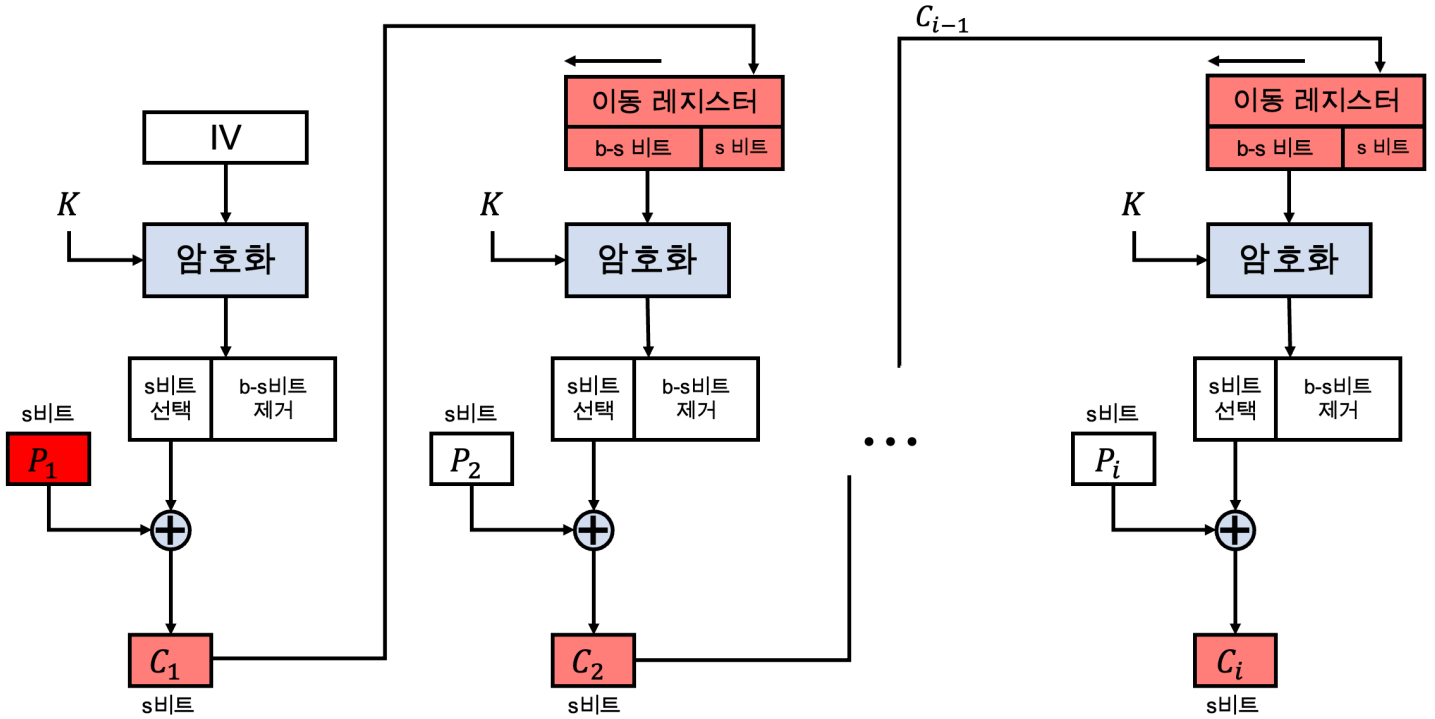
- 종류(3/5)
  - 암호 피드백 모드(CFB, Cipher Feedback Mode)
    - 복호화

$$P_1 = C_1 \oplus S_S[E(K, IV)]$$



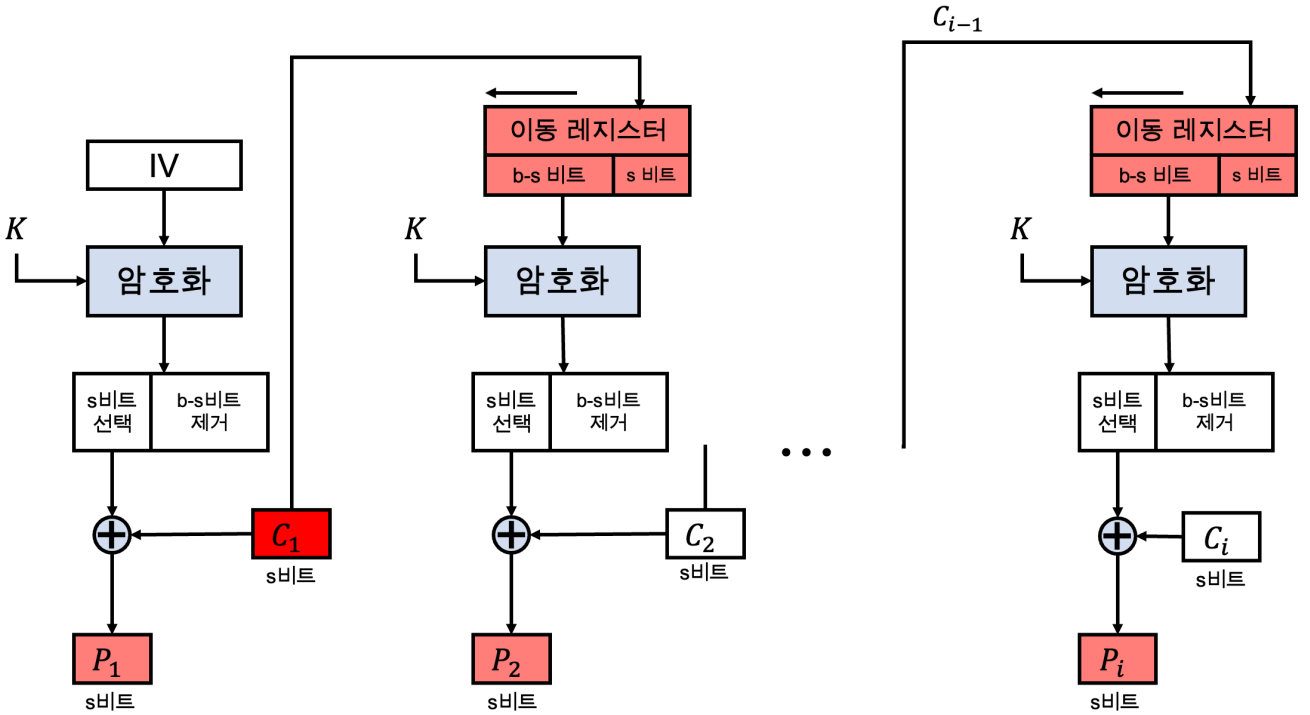
암호 블록 운용 모드

- 종류(3/5)
  - 암호 피드백 모드(CFB, Cipher Feedback Mode)
    - 오류 확산
      - 암호화 과정
        - 한 평문 블록에 오류가 있다면, 현재 암호문 블록 이후 Shift register에서 오류가 소멸될 때까지 암호문에 영향



암호 블록 운용 모드

- 종류(3/5)
  - 암호 피드백 모드(CFB, Cipher Feedback Mode)
    - 오류 확산
      - 복호화 과정
        - 한 암호문 블록에 오류가 있을 때, 현재 평문 블록 이후 Shift register에서 오류가 소멸될 때까지 평문 블록에 영향



암호 블록 운용 모드

- 종류(3/5)
  - 암호 피드백 모드(CFB, Cipher Feedback Mode)
    - 장점
      - 한 메시지 내에서 블록 단위의 패턴이 유지되지 않음
        - 동일한 평문 블록들은 서로 다른 암호화 블록으로 암호화 됨
    - 단점
      - 동일한 초기 벡터를 사용하는 환경에서 두 개의 메시지가 동일한 경우 대응되는 암호문은 동일함
      - 블록 간의 연관성 존재
        - 병렬 처리 불가능
        - 랜덤하게 선택된 파일을 암호화하거나 복호화할 때 사용 불가능

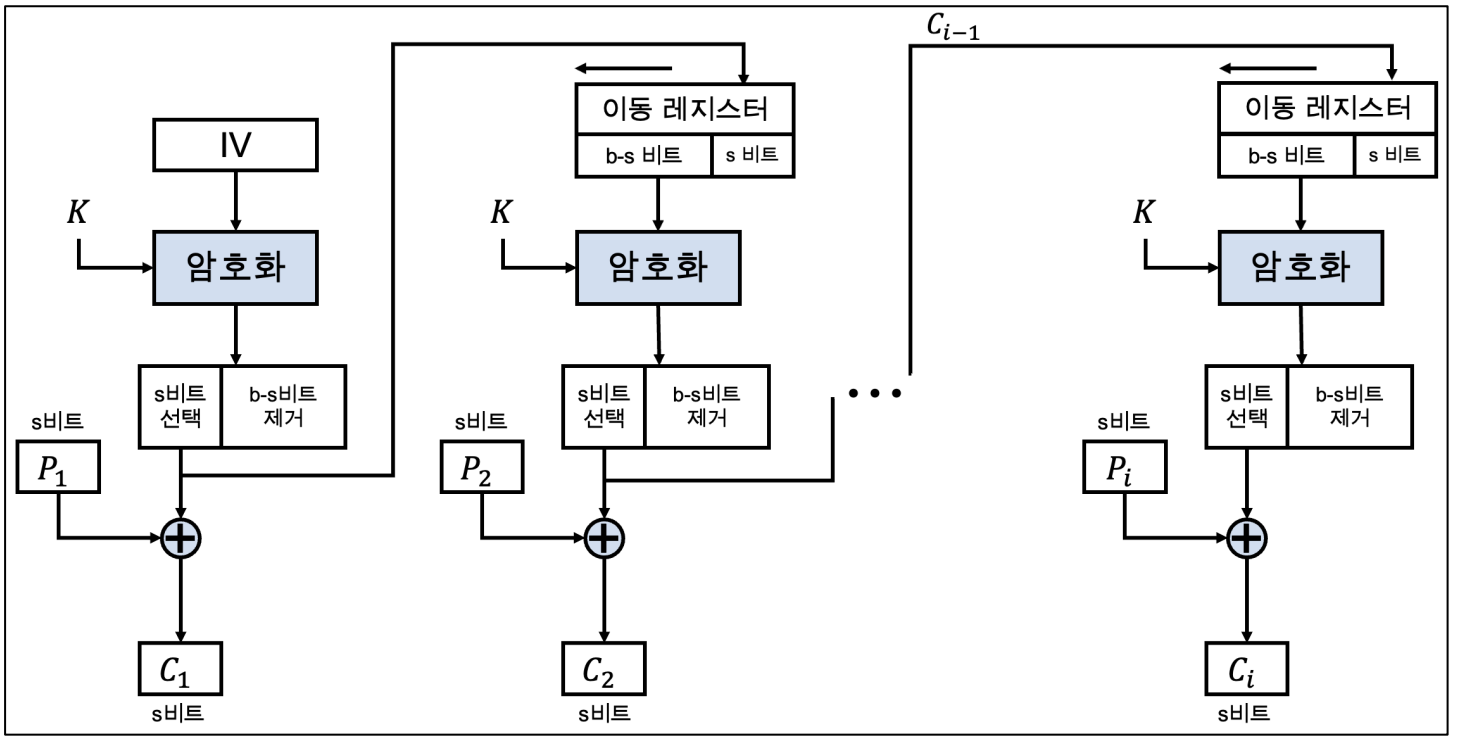
암호 블록 운용 모드

- 종류(4/5)
  - 출력 피드백 모드(OFB, Output Feedback Mode)
    - 정의
      - 평문 블록에 대한 직접적인 암호화 없이 이전 암호 알고리즘의 출력을 피드백하여 평문과 XOR하여 암호화하는 방식
    - 특징
      - 초기화 벡터(IV: Initialization Vector) 사용
      - 암호 피드백 모드를 이용하면 블록 암호를 스트림 암호를 바꿀 수 있음
        - 각 문자가 암호화 되는 즉시 전송 가능
      - 암호화 함수만 사용됨

암호 블록 운용 모드

- 종류(4/5)
  - 출력 피드백 모드(OFB, Output Feedback Mode)
    - 암호화

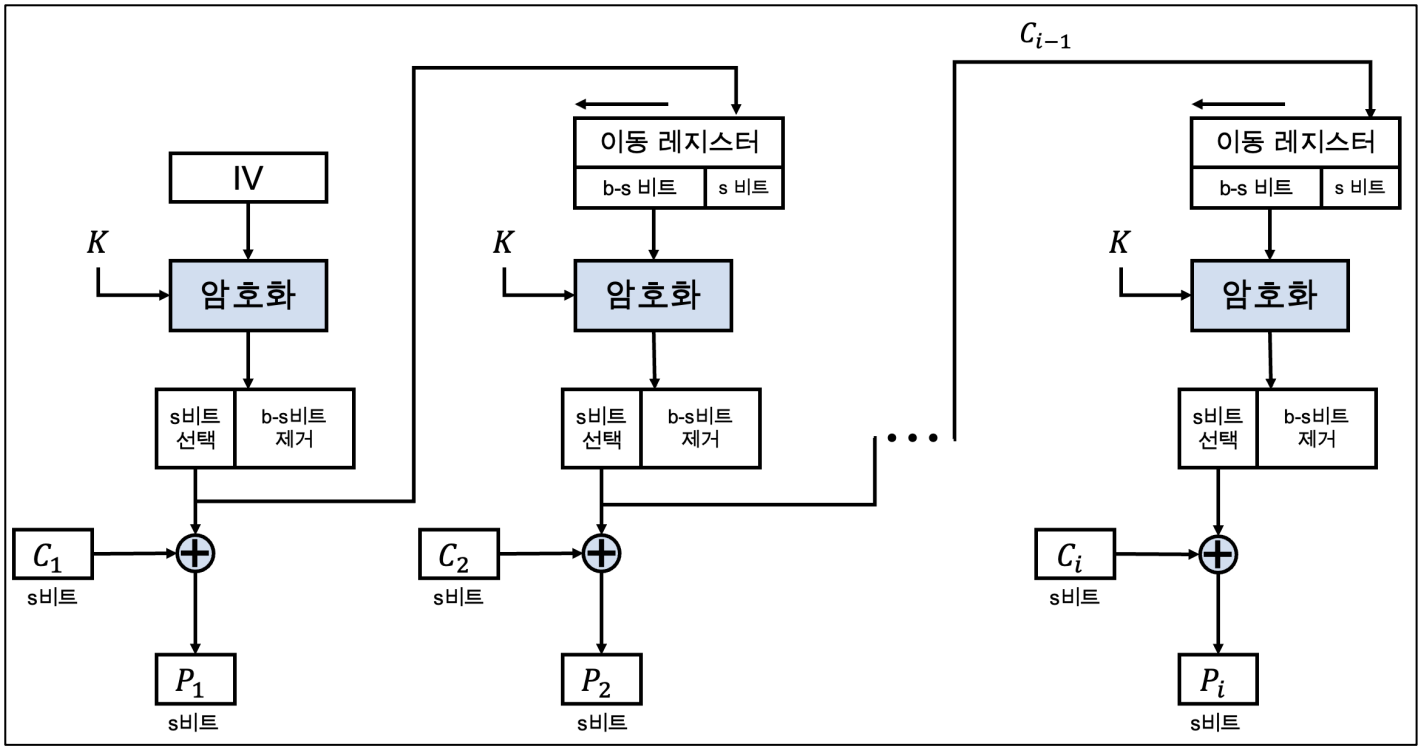
$$C_i = P_i \oplus S_i$$
$$S_i = E(K, S_{i-1})$$



암호 블록 운용 모드

- 종류(4/5)
  - 출력 피드백 모드(OFB, Output Feedback Mode)
    - 복호화

$$P_i = C_i \oplus S_i$$
$$S_i = E(K, S_{i-1})$$



암호 블록 운용 모드

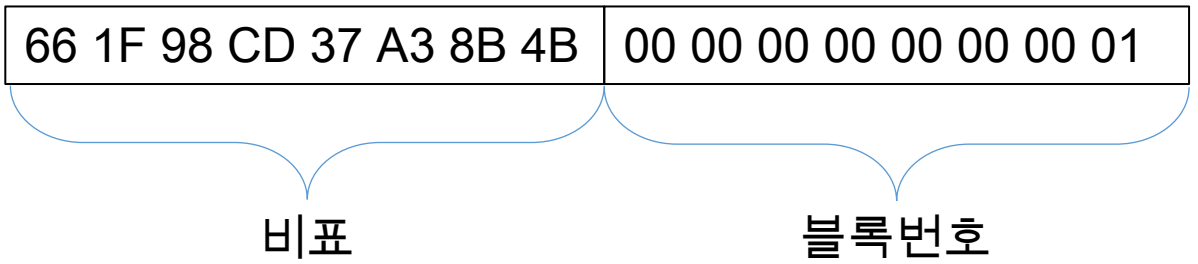
- 종류(4/5)
  - 출력 피드백 모드(OFB, Output Feedback Mode)
    - 장점
      - 한 메시지 내에서 블록 단위의 패턴이 유지되지 않음
        - 동일한 평문 블록들은 서로 다른 암호화 블록으로 암호화 됨
    - 단점
      - 블록 간의 연관성 존재
        - 병렬 처리 불가능

암호 블록 운용 모드

- 종류(5/5)
  - 카운터 모드(CTR, Counter Mode)
    - 정의
      - 카운터를 암호화 하고 평문 블록과 XOR하여 암호 블록을 생성하여 암호화하는 방식
    - 특징
      - 카운터 값이 암호화 할 각각의 평문 블록별로 달라야 함
        - 초기값으로 사용할 카운터 값을 결정한 다음에 그 다음 블록에서 사용할 카운터 값은 이전 카운터에 N을 더하여 만듦
      - 암호화 함수만 사용됨

암호 블록 운용 모드

- 종류(5/5)
  - 카운터 모드(CTR, Counter Mode)
    - 카운터 구조



66 1F 98 CD 37 A3 8B 4B 00 00 00 00 00 00 00 01 (초기값)

66 1F 98 CD 37 A3 8B 4B 00 00 00 00 00 00 00 02 (카운터+1)

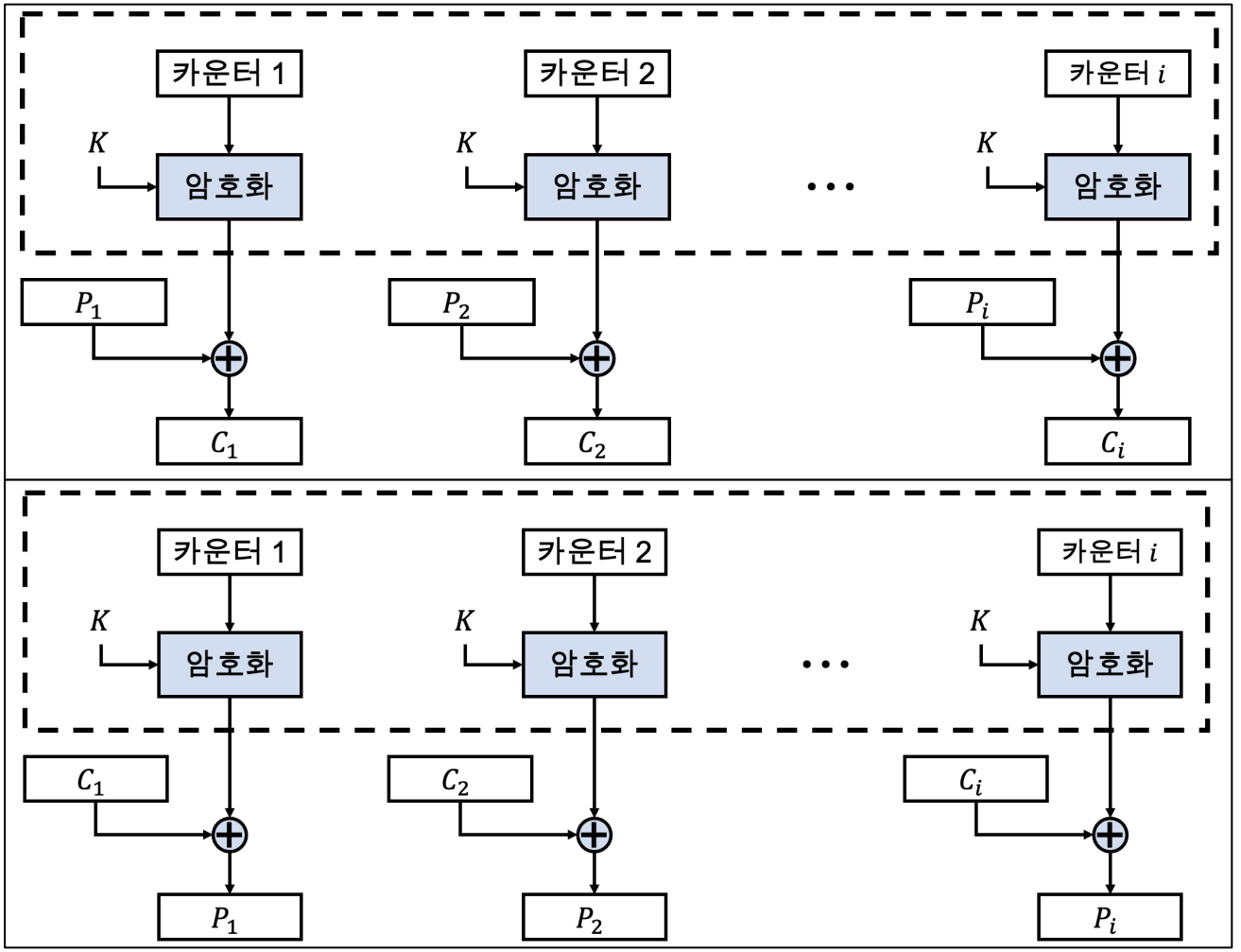
66 1F 98 CD 37 A3 8B 4B 00 00 00 00 00 00 00 03 (카운터+2)

•
•
•

암호 블록 운용 모드

- 종류(5/5)
  - 카운터 모드(CTR, Counter Mode)
    - 암호화

$$C_i = P_i \oplus E(K, Counter)$$



    - 복호화

$$P_i = C_i \oplus E(K, Counter)$$

암호 블록 운용 모드

- 종류(5/5)
  - 카운터 모드(CTR, Counter Mode)
    - 장점
      - 한 메시지 내에서 블록 단위의 패턴이 유지되지 않음
      - 임의 접근 가능
        - 병렬 처리 가능
      - 사전처리 가능
    - 단점
      - 암호화를 수행하기 위해 완전한 비트 블록이 입력되어야 함
        - 실시간 처리 불가능

암호 블록 운용 모드

- 암호 블록 운용 모드 비교

운용 모드	블록 간의 관계	병렬 처리	초기화 벡터	패딩
ECB	독립성	O	X	O
CBC	연관성	복호화만	O	O
CFB	연관성	복호화만	O	X
OFB	연관성	X	O	X
CTR	독립성	O	X	X

Thanks!

손 우 영 (wooyoung@pel.sejong.ac.kr)