정보보호론

대칭키 암호 시스템의 운영모드

한림대학교 소프트웨어융합대학 조효진

Contents

- 운영모드와 패딩

- 운영모드
  - Electronic Codebook Mode (ECB)
  - Cipher Block Chaining (CBC)
  - Counter (CTR)

- 각 운영모드의 특징

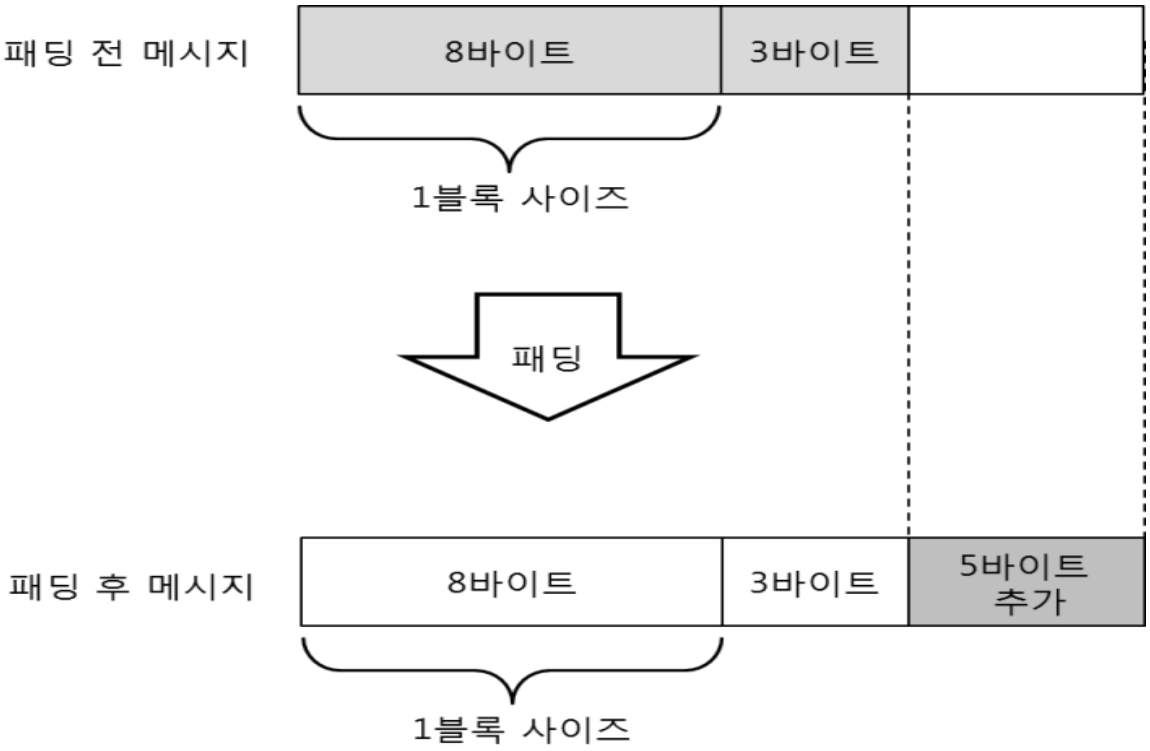
- 암호 가이드라인

운영모드와 패딩

- 운영모드(mode of operation) : DES나 AES와 같은 블록 암호를 사용하여 다양한 크기의 데이터를 암호화하는 방식
  - 실제로 사용되는 평문은 다양한 크기를 가지며 보통 블록크기보다 훨씬 큰 데이터
  - Electronic Codebook Mode (ECB)
  - Cipher Block Chaining (CBC)
  - Counter (CTR)
  - Cipher Feedback (CFB) → 수업에서 다루지 않음
  - Output Feedback (OFB) → 수업에서 다루지 않음

운영모드와 패딩

- 블록 Cipher의 경우, 평문의 길이가 정확하게 해당 블록 암호의 블록 크기의 배수가 되어야 함
  - 패딩은 평문의 전체가 블록 크기의 배수가 되도록 마지막 부분의 빈 공간을 채워 하나의 완전한 블록으로 만드는 작업

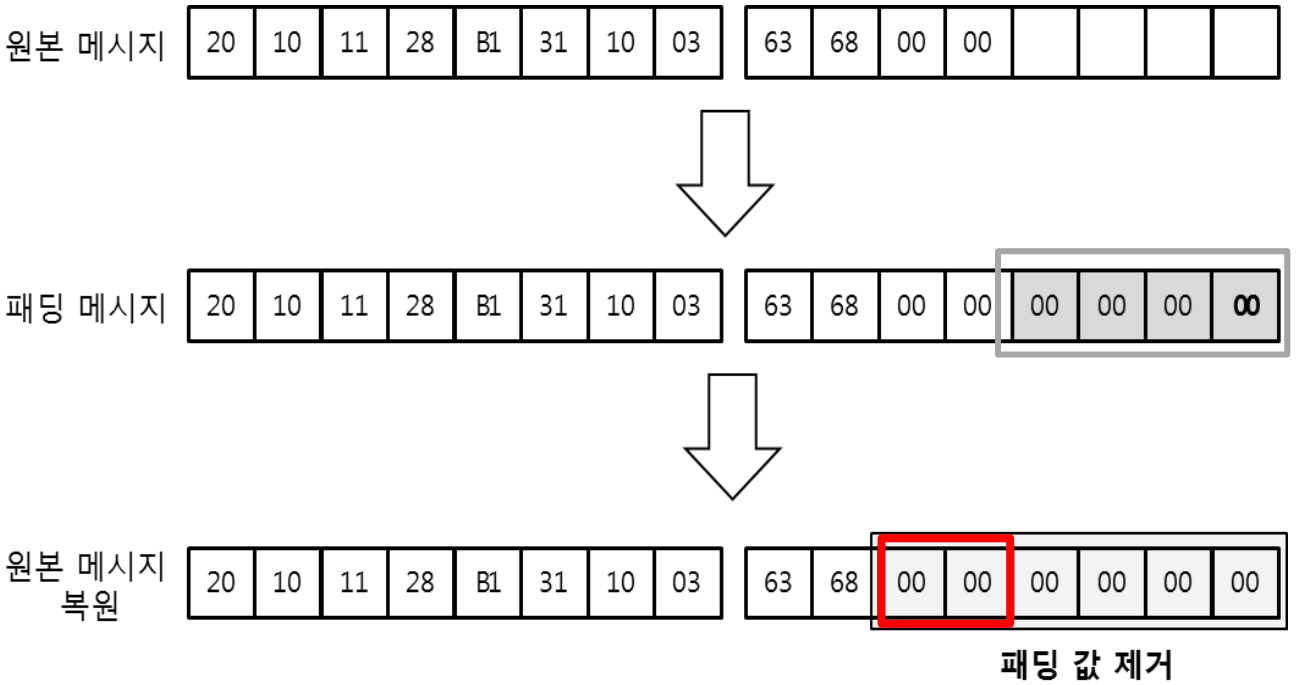


운영모드와 패딩

❒ 제로 패딩(Zero padding, Null padding)

... | 31 AB 34 FE 52 5E 97 12 | 3A FE 5A **00 00 00 00 00** |$_{(16)}$

❒ 제로 패딩의 문제점



운영모드와 패딩

❐ PKCS7 (Public-Key Cryptography Standard) 패딩

- 패딩 바이트 값을 패딩 바이트 크기로 사용

원본 메시지

23	AF	4E	30
AB	3E	7F	97
64	64	90	5E
6F	26	8A	6F

50	AF	4E	30
AB	3E	84	97
64	64		

패딩 메시지

23	AF	4E	30
AB	3E	7F	97
64	64	90	5E
6F	26	8A	6F

50	AF	4E	30
AB	3E	84	97
64	64	**06**	**06**
06	**06**	**06**	**06**

원본 메시지

23	AF	4E	30
AB	3E	7F	97
64	64	90	5E
6F	26	8A	6F

50	AF	4E	30
AB	3E	84	97
64	64	98	6F
3E	AC	68	20

패딩 메시지

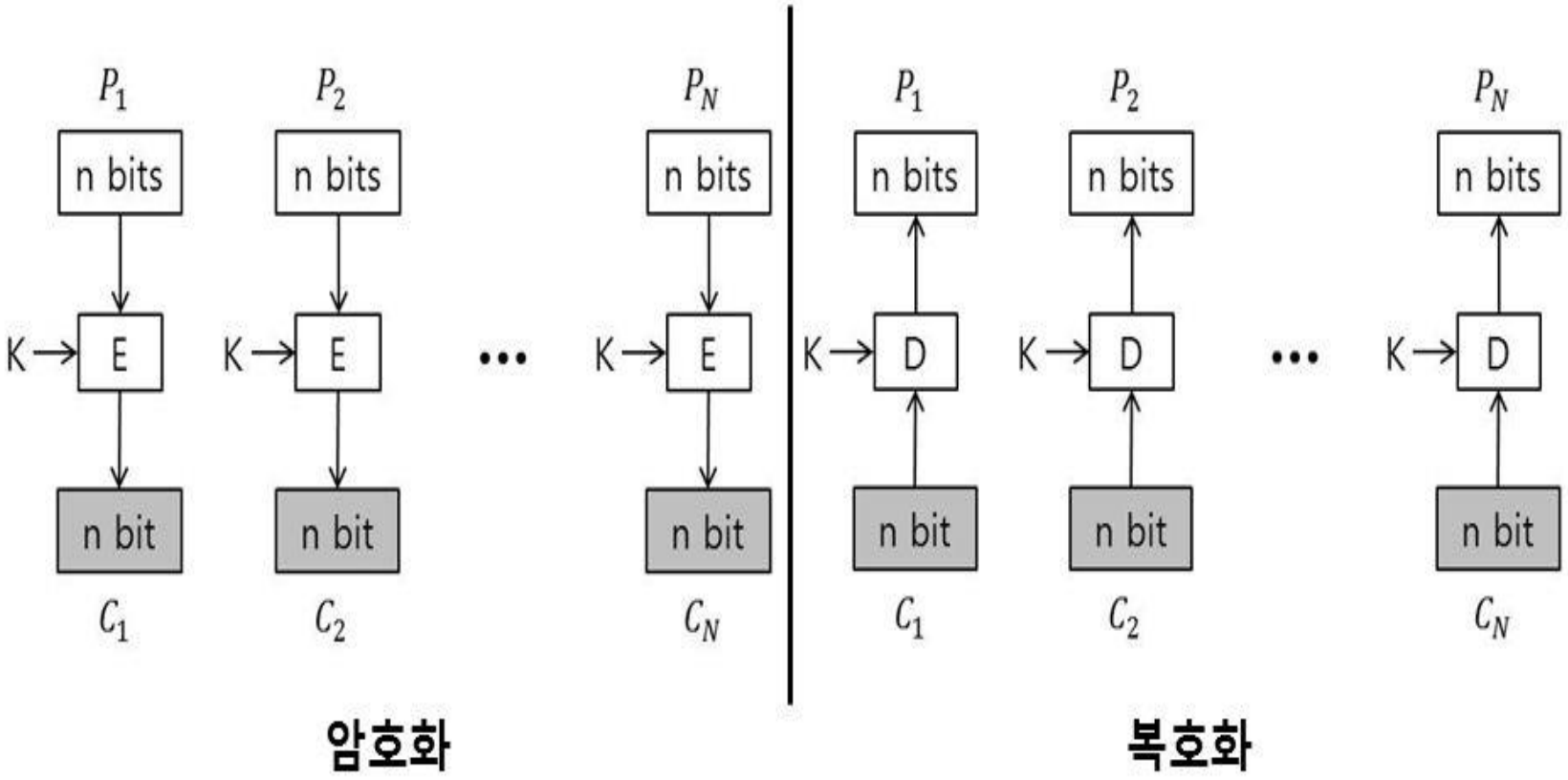
23	AF	4E	30
AB	3E	7F	97
64	64	90	5E
6F	26	8A	6F

50	AF	4E	30
AB	3E	84	97
64	64	98	6F
3E	AC	68	20

10	**10**	**10**	**10**
10	**10**	**10**	**10**
10	**10**	**10**	**10**
10	**10**	**10**	**10**

운영모드: Electronic Codebook Mode (ECB)

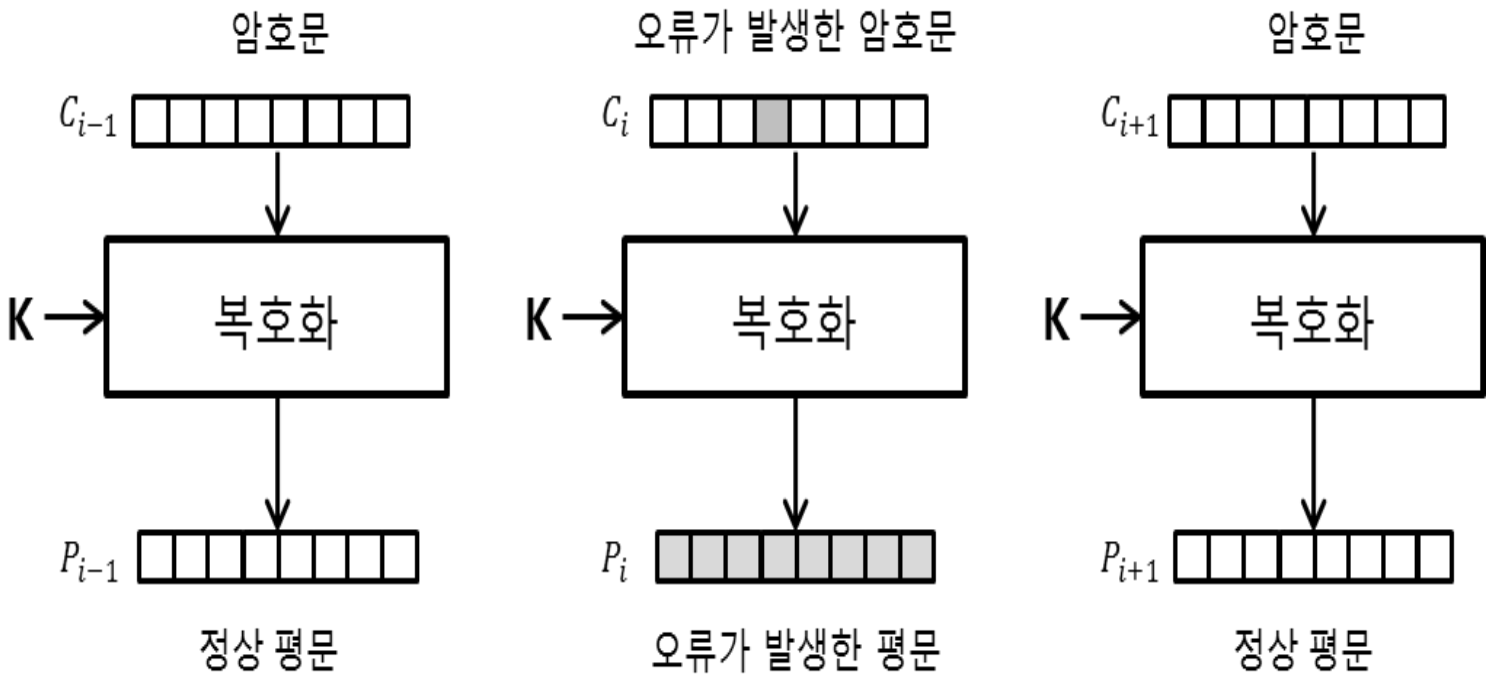
- 한 블록의 평문은 한 블록의 암호문으로 암호화된다
  - 암호화 : $C_i = E_K(P_i)$ 복호화 : $P_i = D_K(C_i)$



운영모드: Electronic Codebook Mode (ECB)

ECB모드의 장/단점

- 장점 : 병렬 처리가 가능 & 오류확산 x

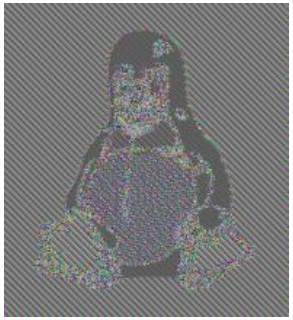


운영모드: Electronic Codebook Mode (ECB)

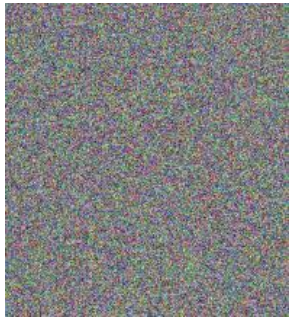
❒ ECB모드의 장/단점

- 단점 : 같은 평문에 대해 같은 암호문
  - 블록 단위의 패턴 유지

Original



Encrypted using ECB mode



Encrypted using other modes

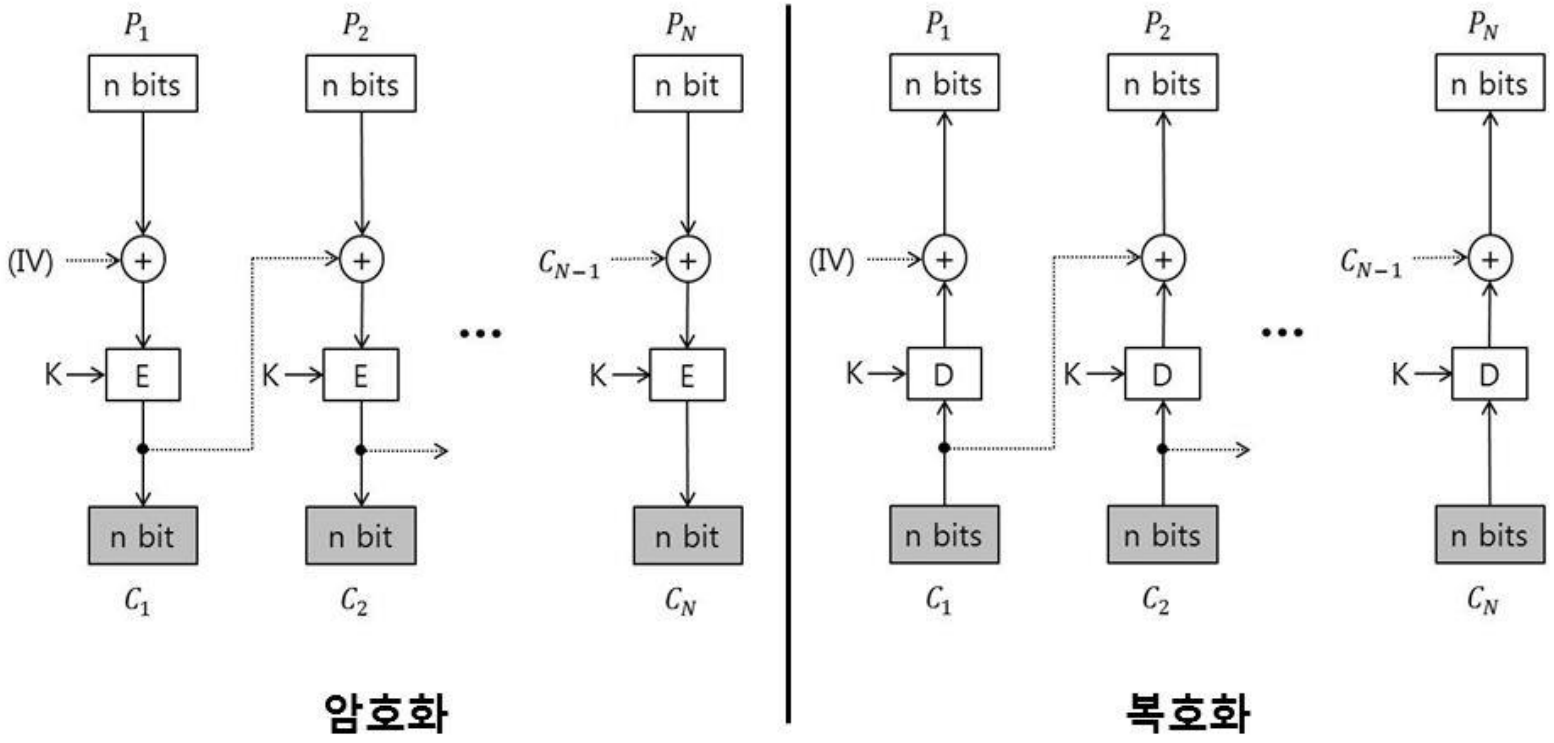
  - 블록 재사용 (Block Replay)

이름	암호화된 점수 (원본 점수)
Alice	0F14D3F2 (90)
Bob	3DE9001F (80)
Eve	549F2D4F (50)

이름	암호화된 점수 (원본 점수)
Alice	0F14D3F2 (90)
Bob	3DE9001F (80)
Eve	**0F14D3F2 (90)**

운영모드: Cipher Block Chaining (CBC)

- 한 평문 블록이 암호화 되기 이전에 바로 앞 평문 블록의 암호문과 XOR
  - 암호화: $C_0 = IV, \quad C_i = E_K(P_i \oplus C_{i-1}),\ \mathrm{i} = 1, 2, 3, \cdots, \mathrm{N}$
  - 복호화: $C_0 = IV, \quad P_i = D_K(C_i) \oplus C_{i-1},\ \mathrm{i} = 1, 2, 3, \cdots, \mathrm{N}$



IV: Initial Vector

운영모드: Cipher Block Chaining (CBC)

❒ 초기 벡터(IV, Initialization Vector)

- 평문을 암호화할 때마다 초기 벡터(IV)를 바꿈으로 임의화(randomization) → 확률적 암호 알고리즘 (probabilistic encryption algorithm)
  - 확률적 알고리즘이란? 동일한 평문이 암호화될 때 마다 통계적으로 독립된 서로 다른 암호문이 생성되는 성질
  - 현대 암호에서는 반드시 만족되어야 하는 성질
  - ECB 모드의 경우 동일한 평문에 대하여 동일한 암호문이 생성 → 결정적 암호 알고리즘 (deterministic encryption algorithm)

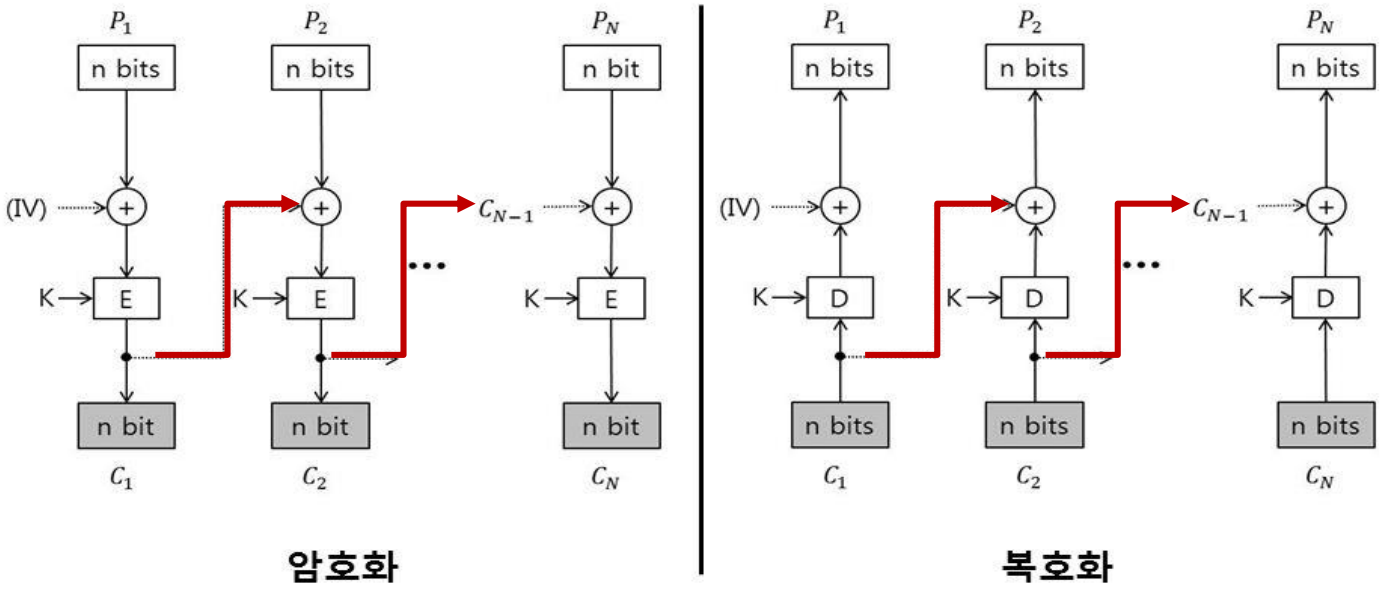
운영모드: Cipher Block Chaining (CBC)

- 초기 벡터(IV, Initialization Vector)
  - Nonce 사용
    - 랜덤한 난수를 만들어 송신자가 수신자에게 그대로 보내는 방법
    - 혹은 서로 동기화된 카운터(counter)를 사용하기도 함
  - 안전성을 강화하기 위하여 nonce를 암호화하여 생성된 암호문을 IV로 사용할 수도 있음
    - Nonce 암호화에 사용되는 키는 송신자와 수신자가 사전에 공유하는 것으로 가정함
  - 하지만, 실제 환경에서는 IV의 기밀성이 아니라 무결성이 중요함
    - 만약 공격자가 전송되는 IV의 한 비트를 변경시킨다면 수신자는 제대로 된 평문을 얻을 수 없기 때문

운영모드: Cipher Block Chaining (CBC)

❒ 연결성(chaining)

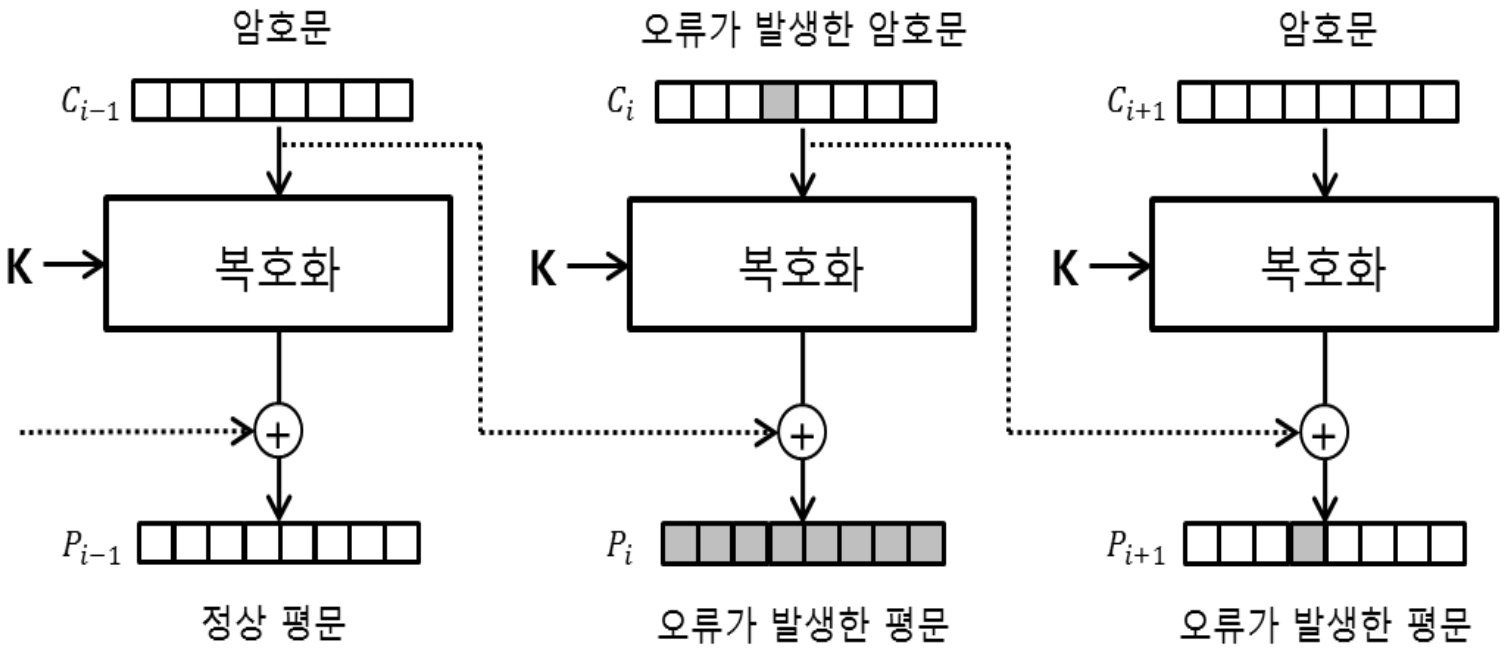
- 한 평문 안에 동일한 두 개의 블록에 대응되는 암호문 블록이 상이
- ECB모드에서 보이는 평문의 블록 패턴들이 CBC의 암호문에서는 더 이상 보이지 않게 됨
- 블록단위의 재사용이 불가능



운영모드: Cipher Block Chaining (CBC)

❒ 오류 확산(Error Propagation)

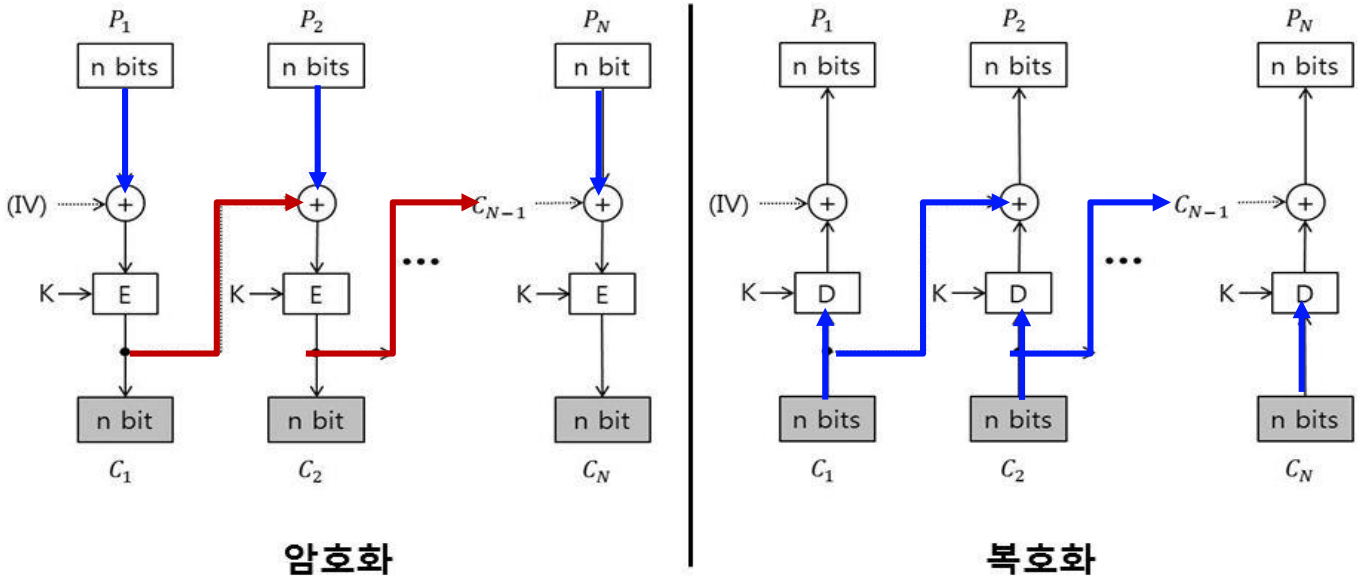
- stage i: $D_k(c_i) \oplus c_{i-1} = P_i$
- stage (i+1): $D_k(c_{i+1}) \oplus c_i = P_{i+1}$
- after stage (i+1), CBC is self-recovering



운영모드: Cipher Block Chaining (CBC)

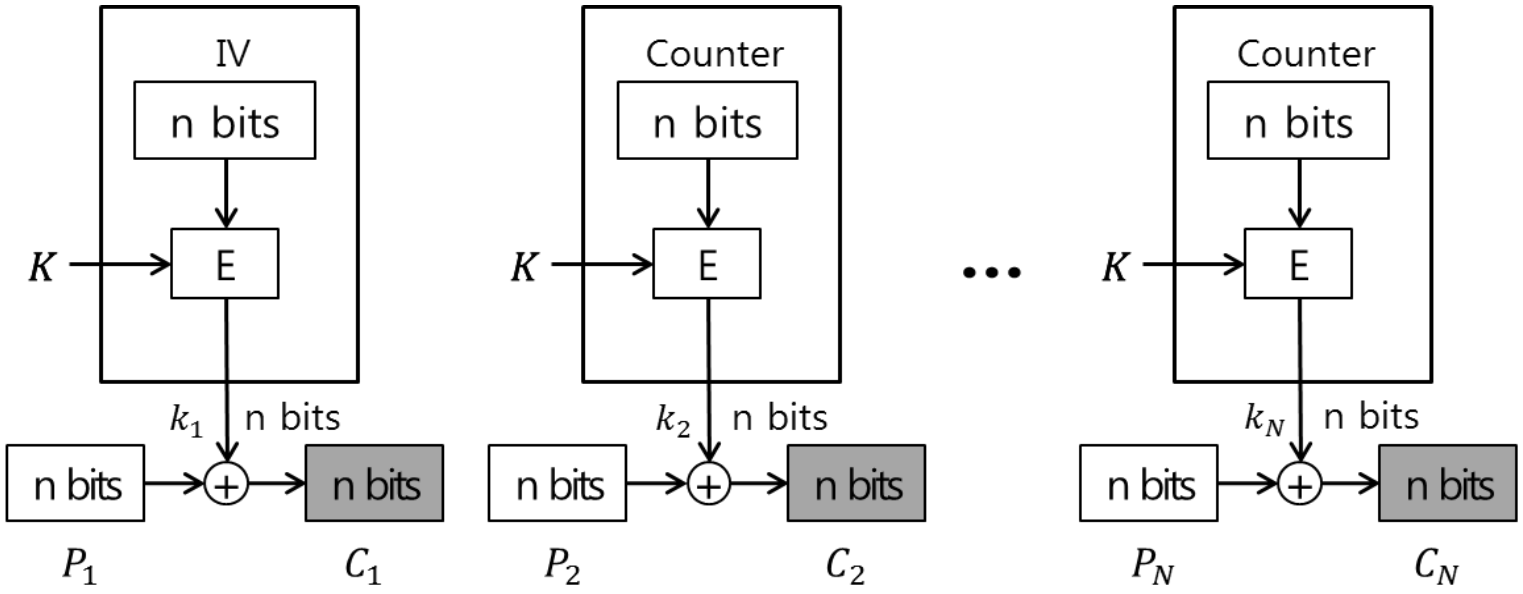
❒ 병렬처리

- 암호화는 불가능함
- 복호화는 가능함



운영모드: Counter (CTR)

- CTR이 암호화됨 → 단 모든 평문 블럭마다 CTR은 달라해 함
  - 가장 간단한 방법은 CTR = CTR + 1
  - 전처리, 병렬처리 가능
  - 암호화 : $C_i = P_i \oplus E_K(\text{Counter})$, $\text{i} = 1, 2, 3, \cdots, \text{N}$
  - 복호화 : $P_i = C_i \oplus E_K(\text{Counter})$, $\text{i} = 1, 2, 3, \cdots, \text{N}$



운영모드: Counter (CTR)

❒ CTR 모드에서의 고려사항

- 서로 다른 두 평문에 대해 같은 카운터가 두 번 쓰이면 보안문제가 발생함

$$P_1 \oplus k_1 = C_1$$

$$P_2 \oplus k_2 = C_2$$

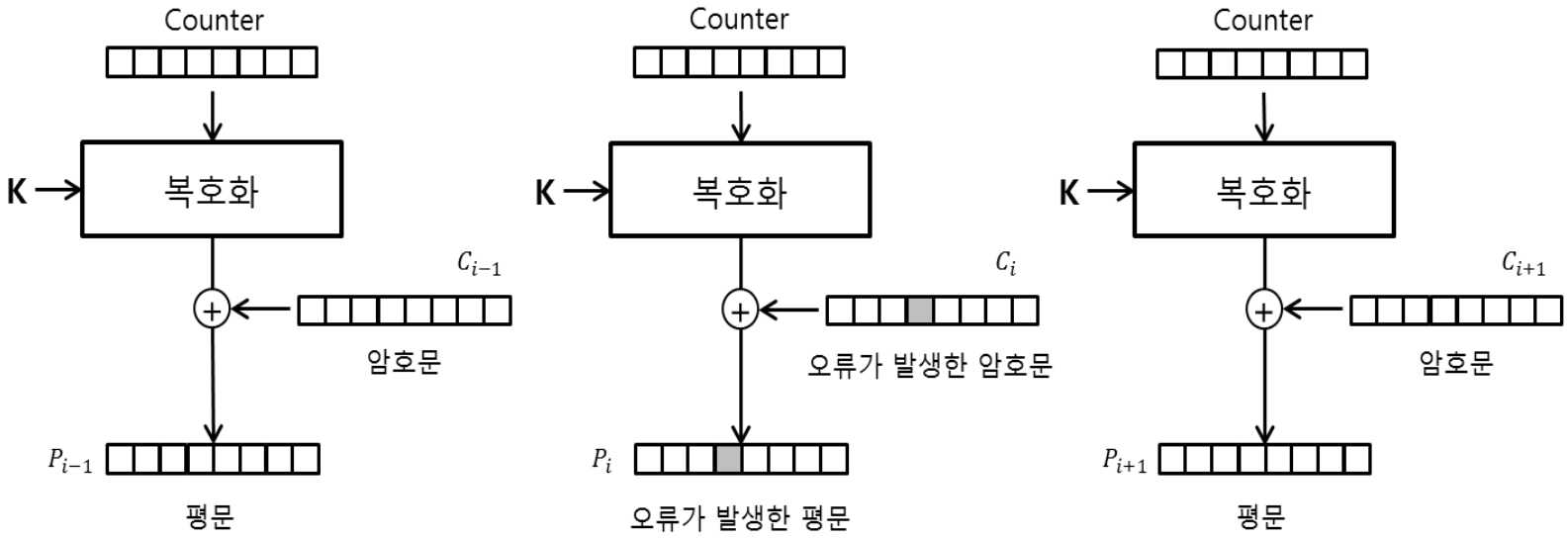
$$\therefore \boldsymbol{C_1} \oplus \boldsymbol{C_2} = (P_1 \oplus k_1) \oplus (P_2 \oplus k_2) = \boldsymbol{P_1} \oplus \boldsymbol{P_2} \ (\because k_1 = k_2)$$

- 즉, 평문과 암호문의 관계식을 알게 되면 전수조사보다 효율적인 공격이 가능해짐
- 따라서, 미 표준문서 800-38A에 따르면, CTR 모드에서는 블록들을 암호화 할 때 마다 서로 다른 카운터 값을 사용할 것을 권장하고 있음

운영모드: Counter (CTR)

❒ CTR 모드의 오류 확산

- 평문과 암호문의 한 비트 오류는 각각 대응되는 암호문과 평문의 한 비트에만 영향을 줌



각 운영모드의 특징

	ECB	CBC	CTR
블록 패턴 유지	○	X	X
전처리 가능성	X	X	○
병렬 처리	○	복호화시 가능	○
오류 확산	X	(P_i, P_{i+1}) 블록에 영향	X
암호화 단위	n	n	$r \leq n$

암호 가이드라인

❐ 한국 인터넷 진흥원 (KISA)

KISA-GD-2018-0034

www.kisa.or.kr

암호이용활성화

암호 알고리즘 및 키 길이 이용 안내서

과학기술정보통신부 Ministry of Science and ICT

KISA 한국인터넷진흥원

보안강도	NIST(미국)	CRYPTREC(일본)	ECRYPT(유럽)	국내
112 비트 이상	AES-128 AES-192 AES-256 3TDEA	AES-128 AES-192 AES-256 Camellia-128 Camellia-192 Camellia-256	AES-128 AES-192 AES-256 Camellia-128 Camellia-192 Camellia-256 Serpent-128 Serpent-192 Serpent-256	SEED HIGHT ARIA-128 ARIA-192 ARIA-256 LEA-128 LEA-192 LEA-256
128 비트 이상	AES-128 AES-192 AES-256	AES-128 AES-192 AES-256 Camellia-128 Camellia-192 Camellia-256	AES-128 AES-192 AES-256 Camellia-128 Camellia-192 Camellia-256 Serpent-128 Serpent-192 Serpent-256	SEED HIGHT ARIA-128 ARIA-192 ARIA-256 LEA-128 LEA-192 LEA-256
192 비트 이상	AES-192 AES-256	AES-192 AES-256 Camellia-192 Camellia-256	AES-192 AES-256 Camellia-192 Camellia-256 Serpent-192 Serpent-256	ARIA-192 ARIA-256 LEA-192 LEA-256
256 비트 이상	AES-256	AES-256 Camellia-256	AES-256 Camellia-256 Serpent-256	ARIA-256 LEA-256

암호 가이드라인

❐ 다양한 국산 대칭키 암호 알고리즘

[한국인터넷정보학회 칼럼] 국내 자체 개발 블록 암호 알고리즘 ...
보안뉴스 - 2017. 3. 29.
현재까지 국내의 자체 연구로 개발된 블록 암호는 **SEED** 암호, HIGHT 암호, **ARIA** 암호 및 **LEA** 암호가 있다. 먼저 **SEED** 블록 암호는 민간 부분인 ...

대한민국 경량 블록암호 '**LEA**', ISO/IEC 표준 승인의 의미
보안뉴스 - 2019. 11. 12.
권대성 센터장은 "전자상거래용 국내 암호 알고리즘인 '**SEED**(1999)'와 전자정부 등 국가 공공분야용 암호 '**ARIA**(2003)', 경량 환경 정보보호 암호 ...

국민대 정보보안암호수학과 이옥연 교수 연구팀, 양자난수 ...
전자신문 - 2018. 11. 19.
12mm×12mm, 22mm×22mm 등 두 가지 크기로 개발된 하드웨어 기반의 암호모듈 'DUSSQ 시리즈'는 AES 뿐만 아니라, **ARIA**, **SEED**, **LEA**, HiGHT ...

경량화되고 최적화된 IoT 보안 플랫폼 'Gzone Security'
CCTV NEWS - 2019. 4. 17.
스마트 경량 IoT 에 적용 가능한 국내외 표준을 준수하며, AES, **ARIA**, **LEA**, **SEED**, HIGHT, ECC, ECDSA, ECDC, PSK 등의 기반한 대칭/비대칭 ...

Appendix#1 암호 라이브러리

- Bouncy Castle (https://www.bouncycastle.org/)

WELCOME

Welcome to the home of the **Legion of the Bouncy Castle**. A fun place to stay, if you've got some time to kill.

BouncyCastle

Home of open source libraries of the Legion of the **Bouncy Castle** and their Java cryptography and C# cryptography resources.

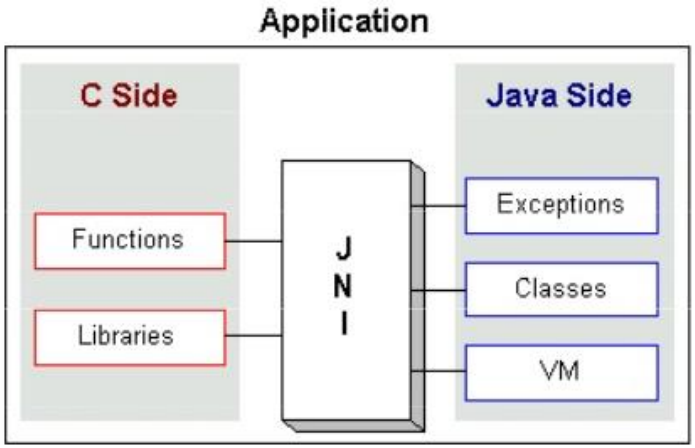
- OpenSSL (https://www.openssl.org/)

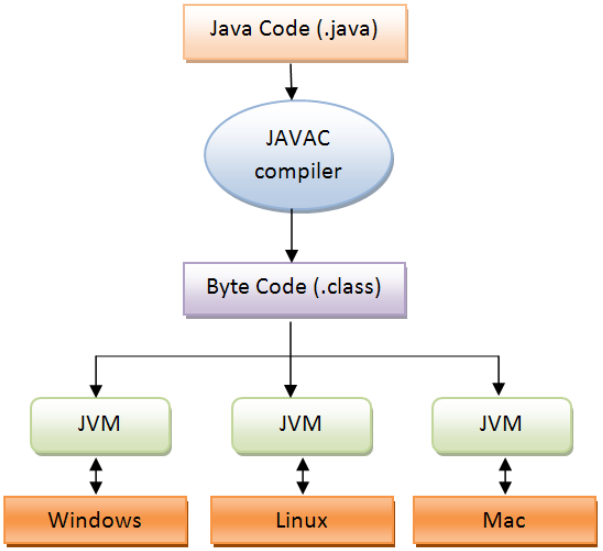
OpenSSL
Cryptography and SSL/TLS Toolkit

Appendix#1 암호 라이브러리

❒ 최적화 구현

- E.g., JNI (Java Native Interface)





- CHES Conference
  - Conference on Cryptographic Hardware and Embedded Systems

HW #4

- 대칭키 암호에서 패딩과 운영모드가 필요한 이유에 대해 예를 들어 설명하시오.

- ECB, CBC, CTR 운영모드에서 병렬처리가 가능한지와 병렬처리가 가능한 이유를 상세히 설명하시오.

- 4페이지 이내 (A4, 10 pt)
  - 6월 26일 정오 12시까지
  - 늦은 제출 시, 감점
  - 과제 copy 시, 관련된 과제들 모두 0점 처리
  - 제출양식: hwp or pdf

Thank you