2021/07/13, 2021보안 기초 세미나

Network Security Essentials

- Chapter_2 대칭 암호와 메시지 기밀성(1) -

박 재 형(jaehyoung@pel.sejong.ac.kr)

세종대학교 프로토콜공학연구실

목 차

- 대칭 암호 원리
- 대칭 암호 알고리즘
- 랜덤 넘버와 의사 랜덤 넘버

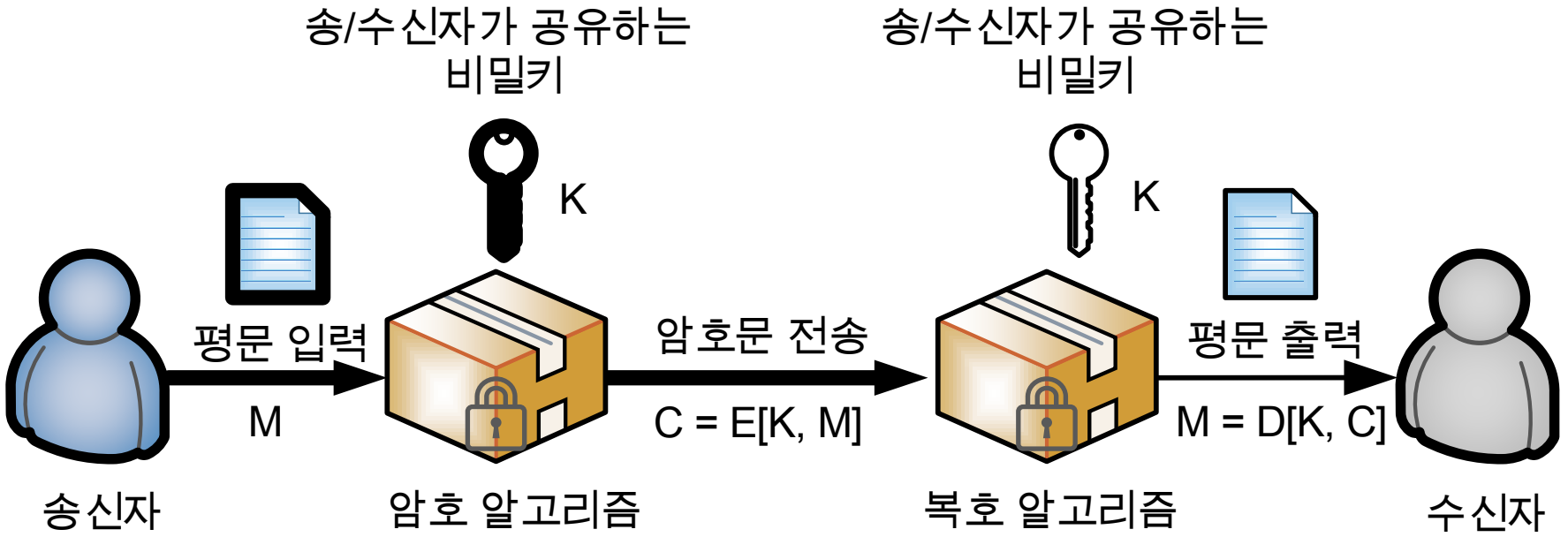
대칭 암호 원리

- 대칭 암호(Symmetric Cipher)
  - 정의
    - 메시지 암호화와 복호화에 같은 키를 사용하는 방식
  - 용어

용어	정의
평문(Plaintext)	누구나 읽을 수 있는 문서나 데이터를 의미
암호화(Encryption)	평문을 누구나 알아 볼 수 없는 형태로 변환하는 방식
암호문(Ciphertext)	평문이 암호 알고리즘에 의해 암호화된 메시지
비밀키(Secret key)	평문을 암/복호화 시키는 핵심 가변 정보 값
암호 알고리즘(Encryption Algorithm)	평문을 암호화하여 암호문으로 변화시키는 알고리즘
복호 알고리즘(Decryption Algorithm)	암호 알고리즘을 역으로 수행하여 암호문을 평문으로 복구하는 알고리즘

대칭 암호 원리

- 대칭 암호 구조
  - 송신자가 평문을 송/수신자가 공유하는 비밀키로 암호 알고리즘을 통해 암호화하여 암호문 전송
  - 수신자는 암호문을 송/수신자가 공유하는 비밀키로 복호 알고리즘을 통해 복호화하여 평문 획득



대칭 암호 원리

- 암호 시스템의 3가지 요소(1/2)
  - 연산 유형
    - 대체(Substitution)
      - 평문에 각 요소(비트, 문자, 블록)를 다른 요소로 변환하여 암호화
        - e.g.,(가, 나, 다,라) ⟶ (아, 자, 타,파)
    - 전치(Transposition)
      - 평문에 각 요소의 순서를 변환하여 암호화
        - e.g.,(가, 나, 다, 라) ⟶ (라, 나, 가, 다)
  - 사용되는 키
    - 송신자와 수신자가 같은 키를 사용하면 대칭키 암호
    - 송신자와 수신자가 다른 키를 사용하면 공개키 암호

대칭 암호 원리

- 암호 시스템의 3가지 요소(2/2)
  - 평문 처리 방법
    - 블록 암호(Block Cipher)
      - 데이터를 정해진 블록 단위로 암호화
        - e.g., 문자 메시지 전송
    - 스트림 암호(Stream Cipher)
      - 데이터를 연속된 비트 또는 바이트 단위로 암호화 하는 것
        - e.g., 오디오/비디오 스트리밍

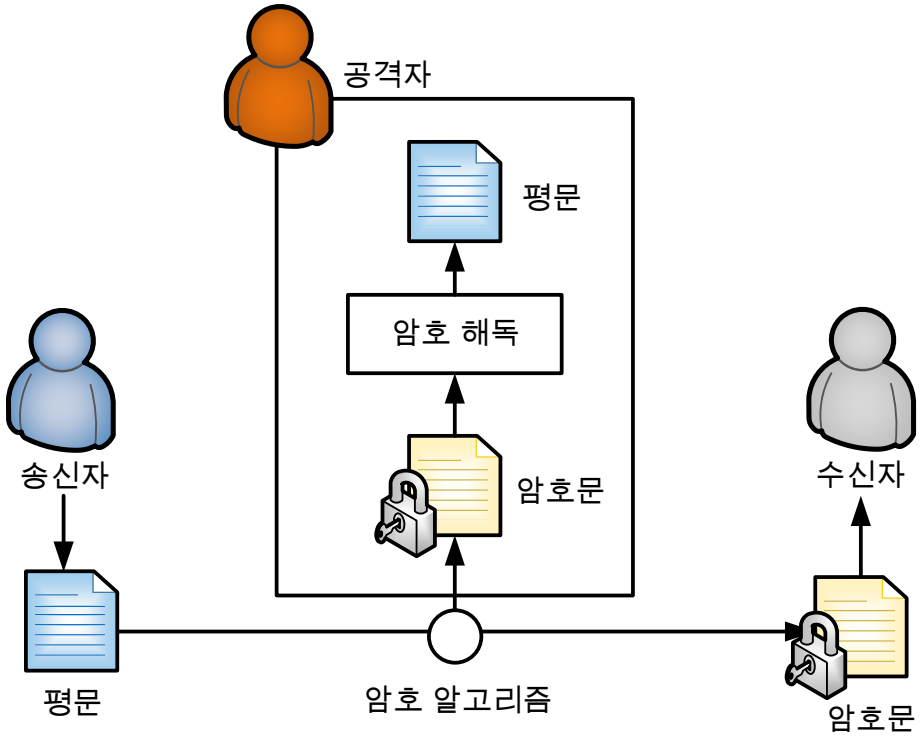
대칭 암호 원리

- 암호 해독
  - 정의
    - 암호 알고리즘에 사용된 평문이나 키를 찾으려는 시도
  - 특징
    - 암호 해독 전략은 암호 구조와 해독가의 정보에 따라 달라짐
      - 암호문 단독 공격(Ciphertext-Only-Attack)
      - 알려진 평문 공격(Known Plaintext Attack)
      - 선택 평문 공격(Chosen-Plaintext Attack)
      - 선택 암호문 공격(Chosen-Ciphertext Attack)

대칭 암호 원리

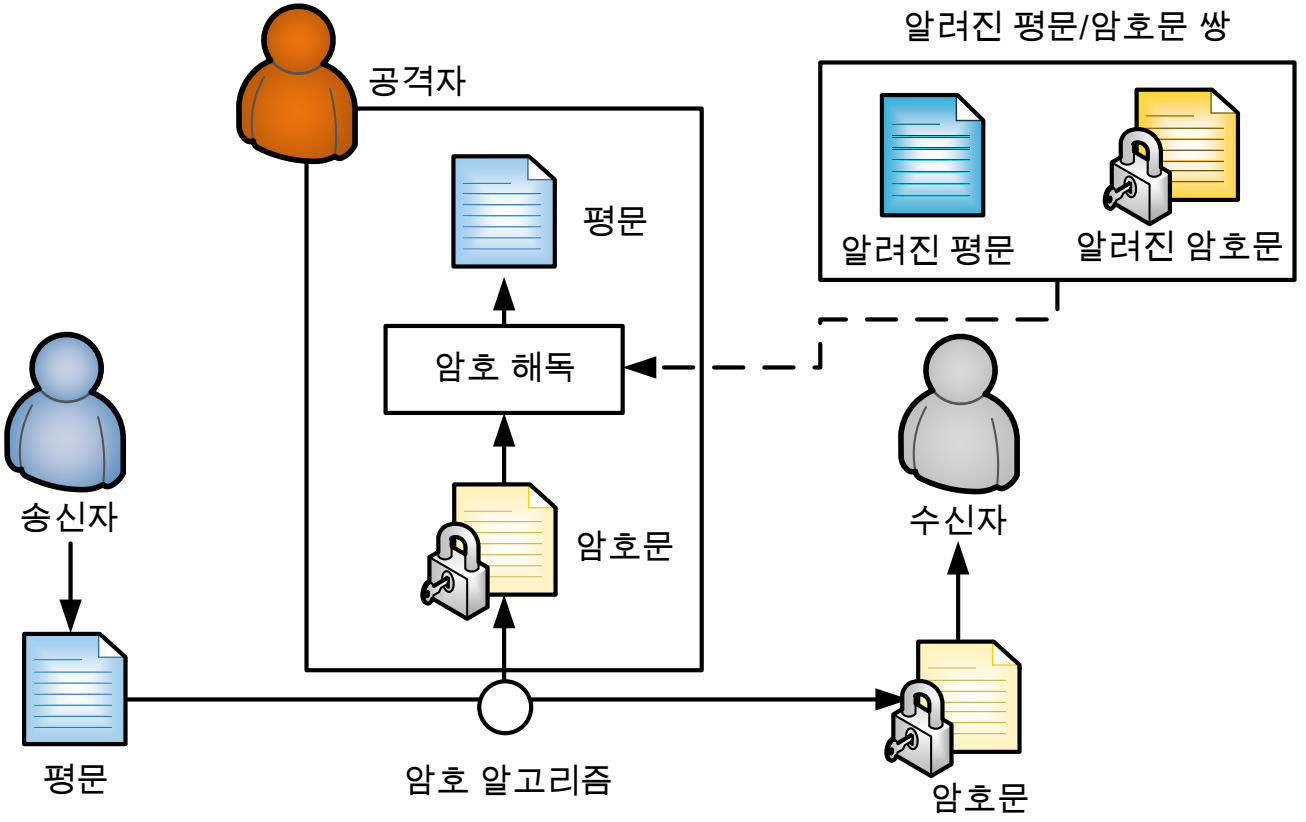
- 암호화된 메시지 공격 유형(1/4)
  - 암호문 단독 공격(Ciphertext-Only-Attack)
    - 단지 암호문만 가지고 평문이나 키를 찾아내는 공격 방법
      - 공격자는 암호 알고리즘과 암호문을 가짐
  - 공격 유형

공격 유형	설명
전수 조사 공격 (Brute Force attack)	암호문을 해독 할 수 있는 모든 경우를 시도해보는 공격
빈도 분석 공격 (Frequency Analysis Attack)	암호문에서 사용되는 문자 또는 문자열의 사용빈도 통계적으로 분석하여 암호문을 해독하는 공격
패턴 공격 (Pattern Attack)	암호문에 존재하는 특정 패턴을 이용하여 평문을 유추하는 공격



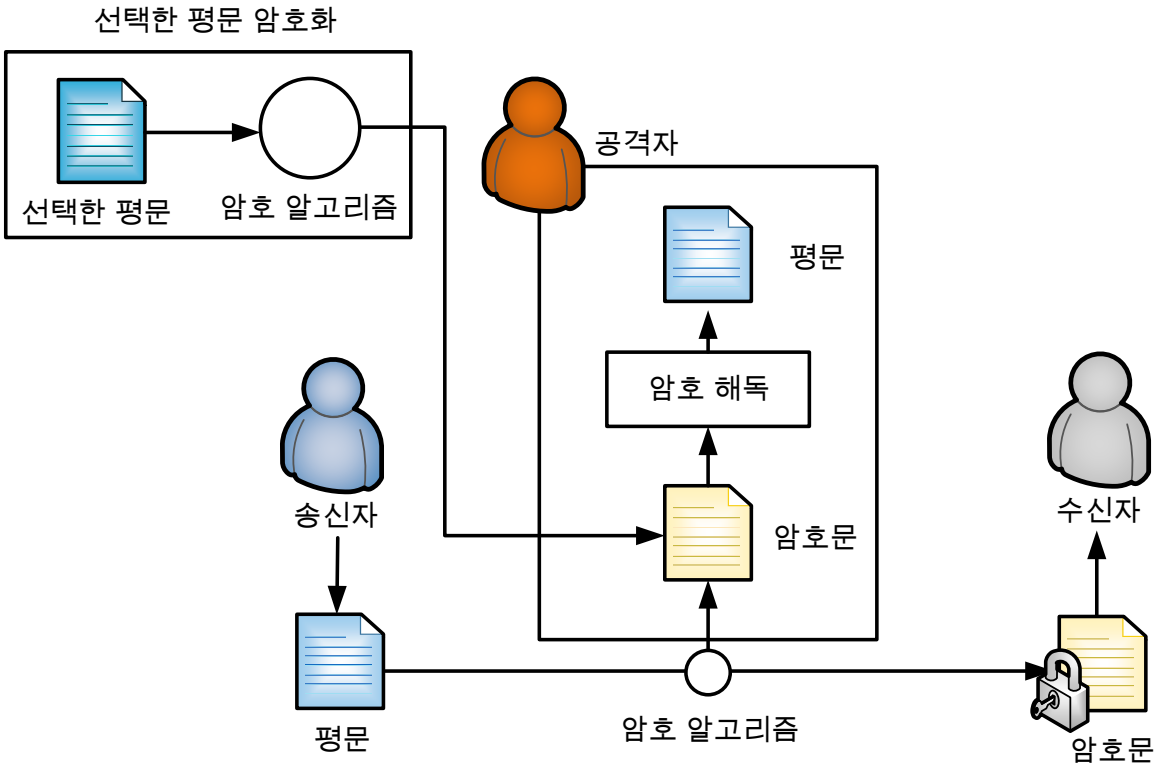
대칭 암호 원리

- 암호화된 메시지 공격 유형(2/4)
  - 알려진 평문 공격(Known Plaintext Attack)
    - 평문이 알려져 있어 암호문과 평문과의 관계로부터 키와 평문을 추정하여 해독하는 공격 방법



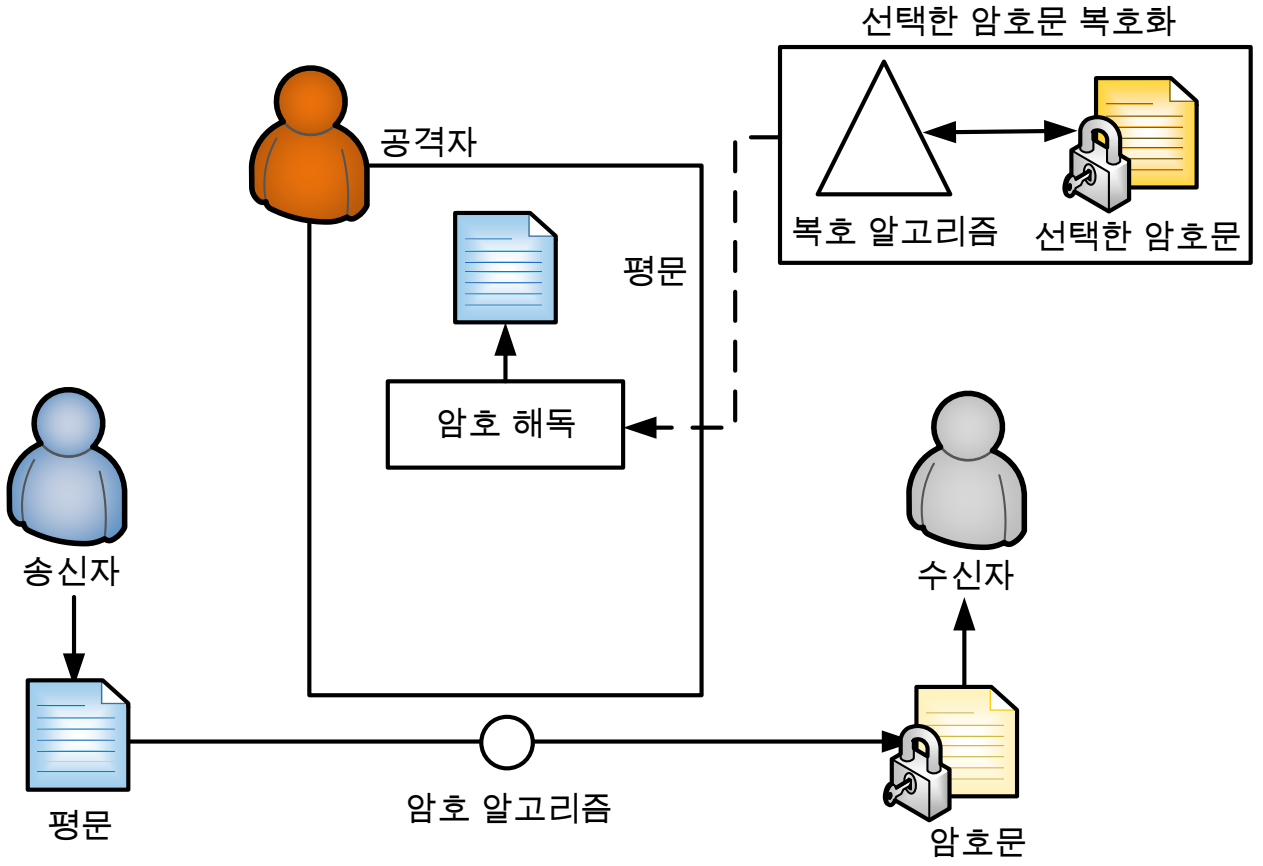
대칭 암호 원리

- 암호화된 메시지 공격 유형(3/4)
  - 선택 평문 공격(Chosen-Plaintext Attack)
    - 공격자가 암호 알고리즘 정보를 알고있을 경우, 평문을 선택하여 암호화하고 암호문 획득
    - 해당 암호문 분석으로 평문 획득



대칭 암호 원리

- 암호화된 메시지 공격 유형(4/4)
  - 선택 암호문 공격(Chosen-ciphertext attack)
    - 공격자가 복호 알고리즘 정보를 알고있는 경우, 암호문을 선택하여 복호화하여 평문 획득



대칭 암호 원리

- 대칭 암호의 요구 사항
  - 신뢰할 수 없는 제3자가 암호문을 해독할 수 없어야 함
  - 통신 주체인 송신자와 수신자는 비밀 키를 신뢰 할 수 있는 키 분배 방식을 통해 공유해야 함
    - e.g., Kerberos 등
  - 암호 알고리즘은 공개 되더라도 키는 알 수 없어야 함
    - 공격자가 암호/복호 알고리즘과 암호문을 알고 있어도 해당 암호문을 해독 할 수 없어야 하기 때문

대칭 암호 원리

- 암호 구조가 계산적으로 안전한 구조
  - 암호문을 복호화 하는데 드는 비용이 정보의 가치보다 더 큰 경우
  - 암호문을 복호화 하는데 소요 시간이 정보의 수명보다 더 긴 경우

- 전수 공격 시, 키 탐색 평균 요구 시간

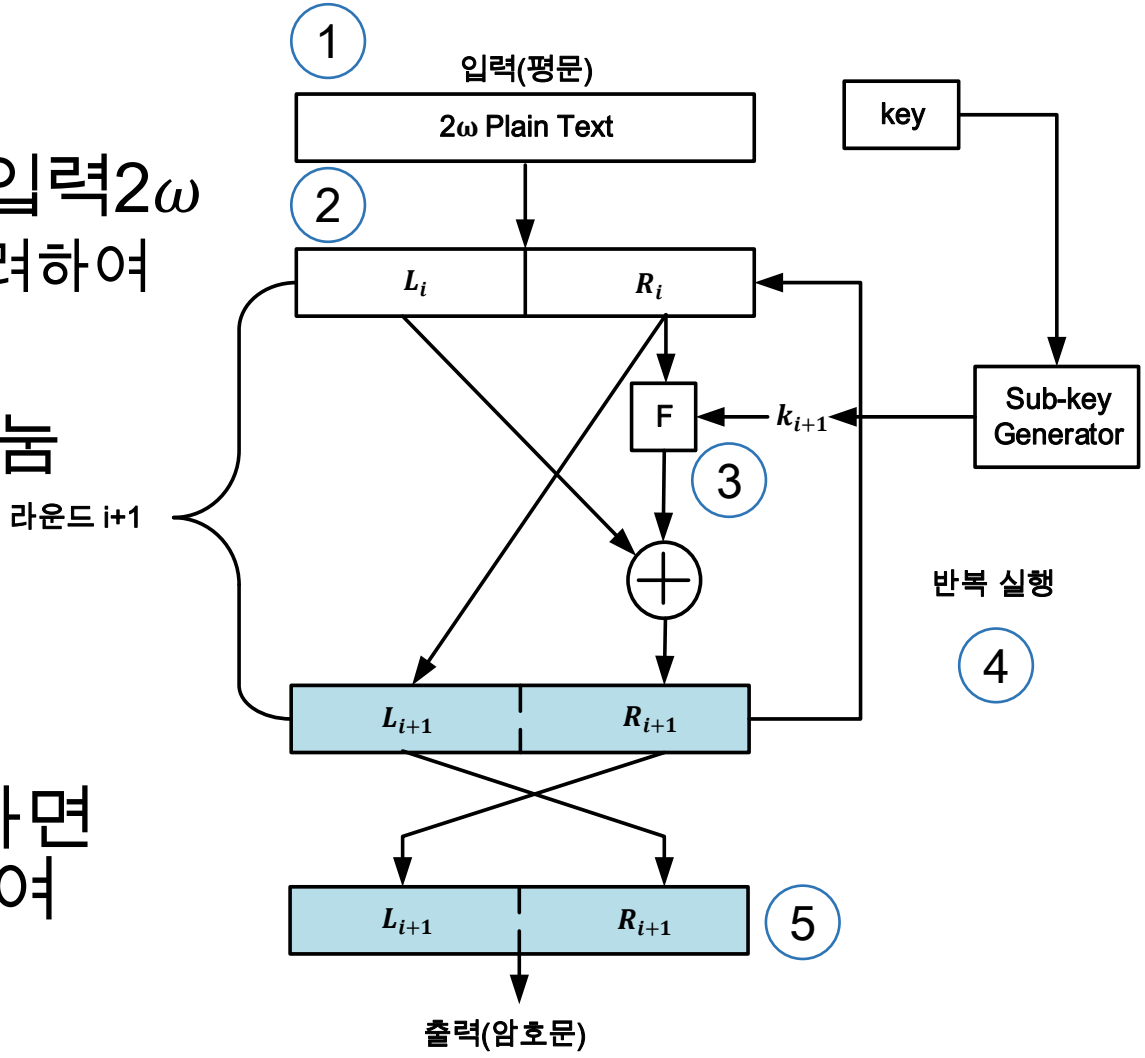
키 크기(비트)	키의 종류 수	μs당 한 번의 암호화를 할 때 소요되는 키 탐색 시간	μs당 106번의 암호화를 할 때 소요되는 키 탐색 시간
32	2^{32} = 4.3x10^{9}	$2^{31}\mu s$ = 35.8분	2.15μs
56	2^{56} = 7.2x10^{16}	$2^{55}\mu$s = 1142년	10.01시간
128	2^{128} = 3.4x10^{38}	$2^{127}\mu$s = 5.4x10^{24}년	5.4x10^{18}년
168	2^{168} = 3.7x10^{50}	$2^{167}\mu s$ = 5.9x10^{30}년	5.9x10^{30}년
26개 문자(치환)	26! = 4x10^{26}	2x$10^{26}\mu s$ = 6.4x10^{12}년	6.4x10^{6}년

대칭 암호 원리

- Feistel 암호 구조
  - 개요
    - 1973년 IBM(International Business Machines)의 Horst Feistel이 최초로 소개한 암호 구조
  - 정의
    - 라운드 함수를 통해 특정 연산을 반복 사용하는 블록암호
  - 특징
    - 대체와 변환을 반복 사용하여 메시지를 암/복호화
    - 여러 개의 라운드로 이루어짐
      - 일반적으로 16라운드 사용, 사용자가 라운드 수 조정 가능
    - 블록 크기와 길이가 길수록 안전성이 높음
      - 암/복호화 속도가 떨어짐

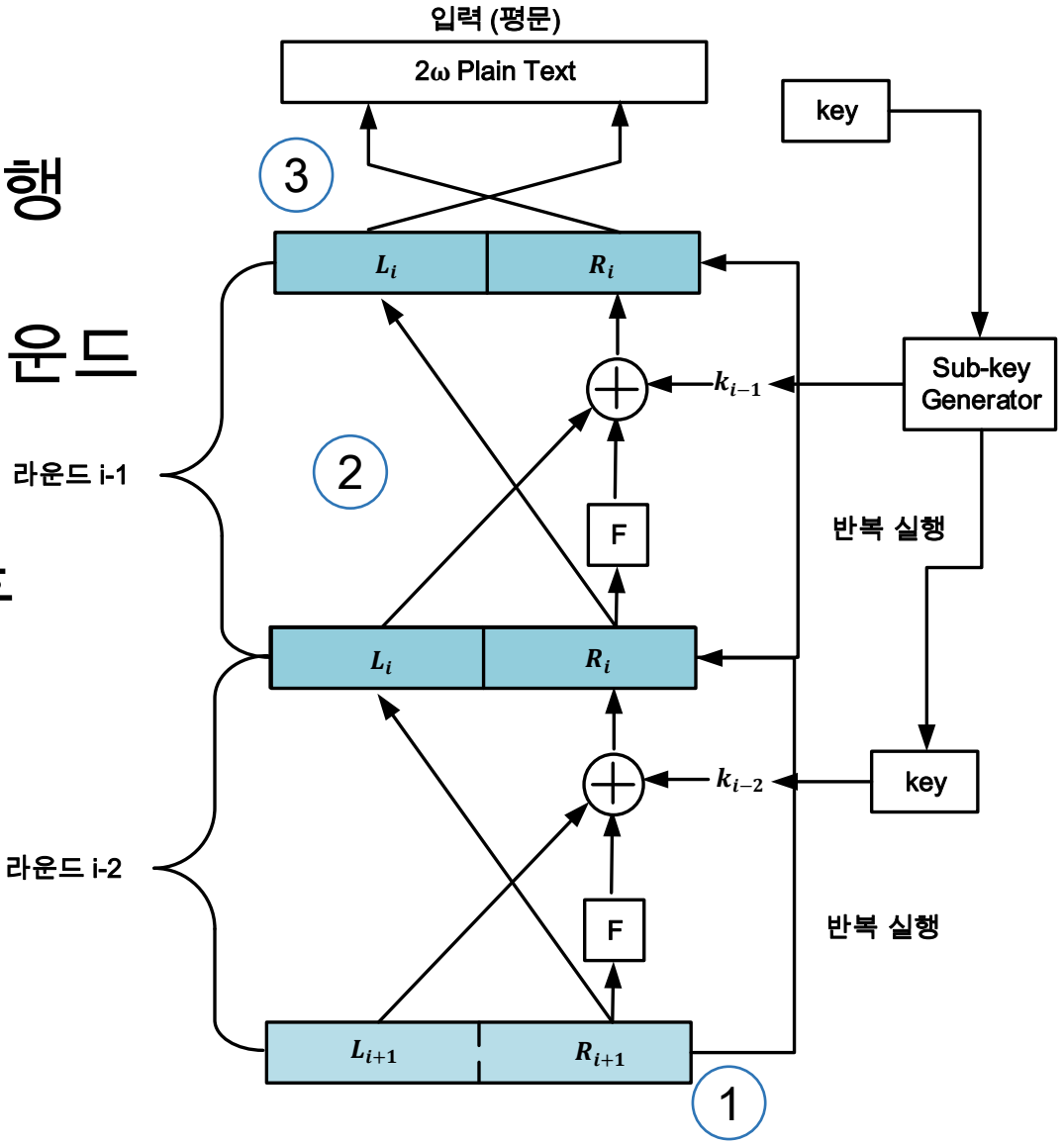
대칭 암호 원리

- Feistel 암호 구조
  - Feistel 암호화 과정
    1. 하나의 평문 블록을 입력2ω
       - 성능과 속도 효율을 고려하여 64bits권장
    2. 평문 블록을 반 씩 나눔
    3. $F(R_i, K_{i+1}) \oplus L_i = R_{i+1}$
    4. 16회 반복 실행
    5. 마지막 라운드가 끝나면 R_{16}의 위치를 교환하여 암호문 출력



대칭 암호 원리

- Feistel 암호 구조
  - Feistel 복호화 과정
    1. 암호화의 역순으로 진행
    2. 64 bits평문 기준 16라운드 진행
    3. 마지막 라운드 진행 후 L_i와R_i의 자리 교환

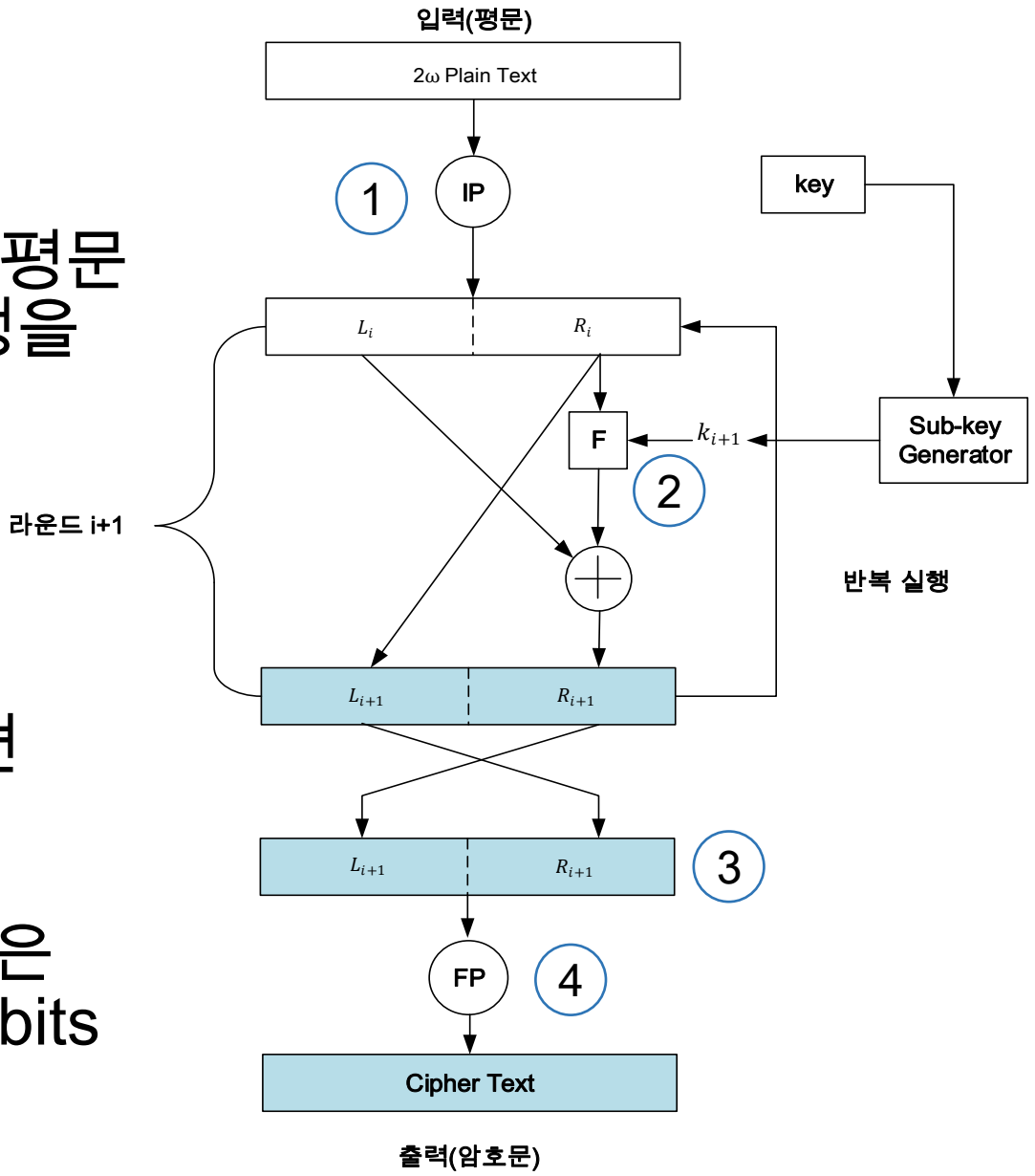


대칭 암호 알고리즘

- DES (Data Encryption Standard)
  - 정의
    - 1972년 미국 국가기술표준원(NIST, National Institute of Standards and Technology)이 개발한 미국 정부 규모의 표준 암호 알고리즘
  - 특징
    - 평문의 길이 64bits
    - 키의 길이 56bits
    - 라운드 횟수 16회
    - Feistel 암호 알고리즘 구조를 취함
    - 각 라운드 마다 서브키를 사용 함
    - 길이가 56bits인 키로부터 길이가 48bits인 서로 다른 16개의 서브 키가 생성 된다

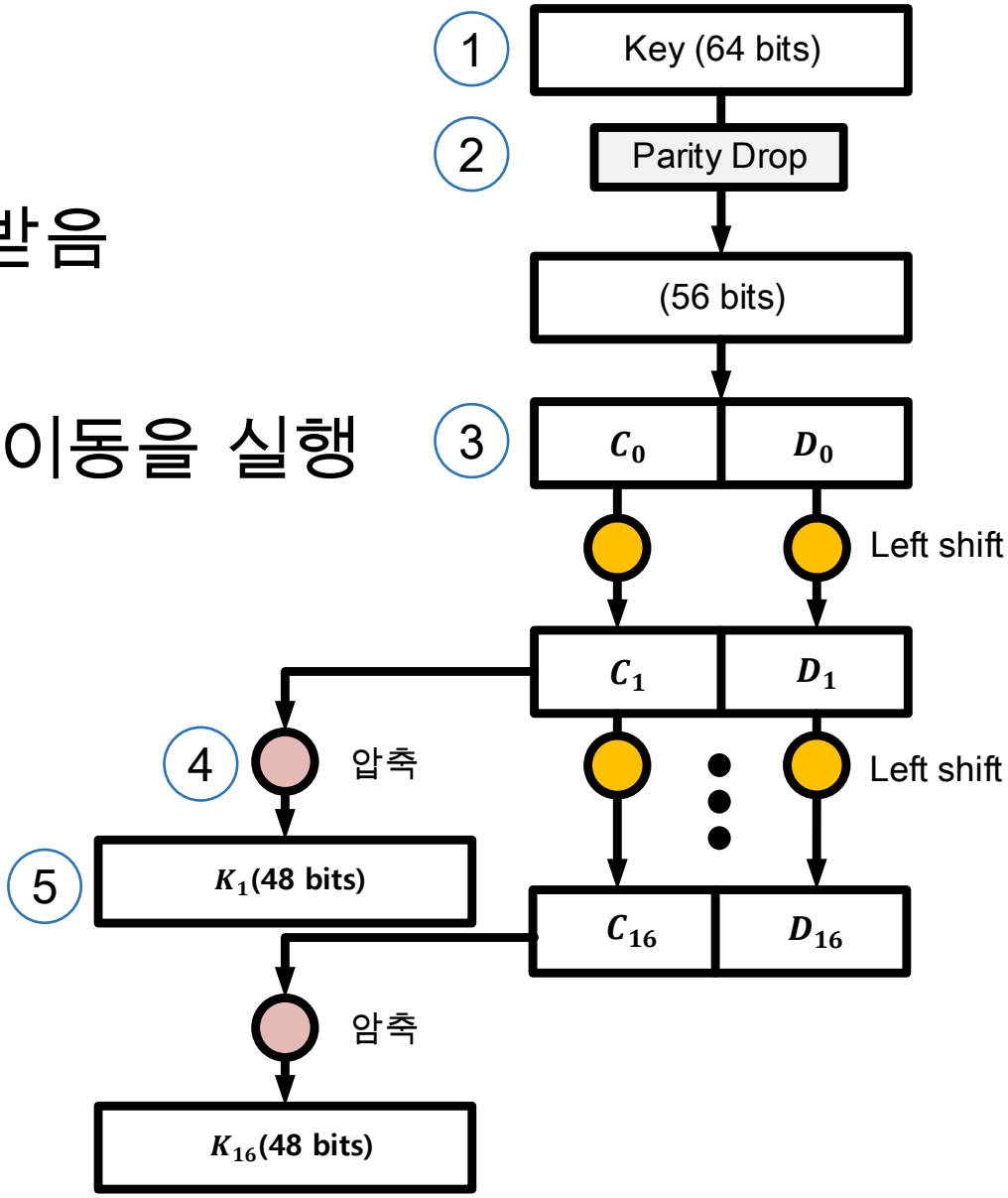
대칭 암호 알고리즘

- DES
  - 암호화 과정
    1. 입력으로 들어온 64bits평문 블록은 초기치환 IP과정을 거친 후 L_i와 R_i으로 32bits 씩 나뉨
    2. $F(R_i, K_{i+1}) \oplus L_i = R_{i+1}$
    3. 마지막 라운드가 끝나면 L_{16}과R_{16}의 위치 교환
    4. 위치 교환 후 평문 블록은 최종 치환FP를 거쳐 64bits 암호문으로 출력됨



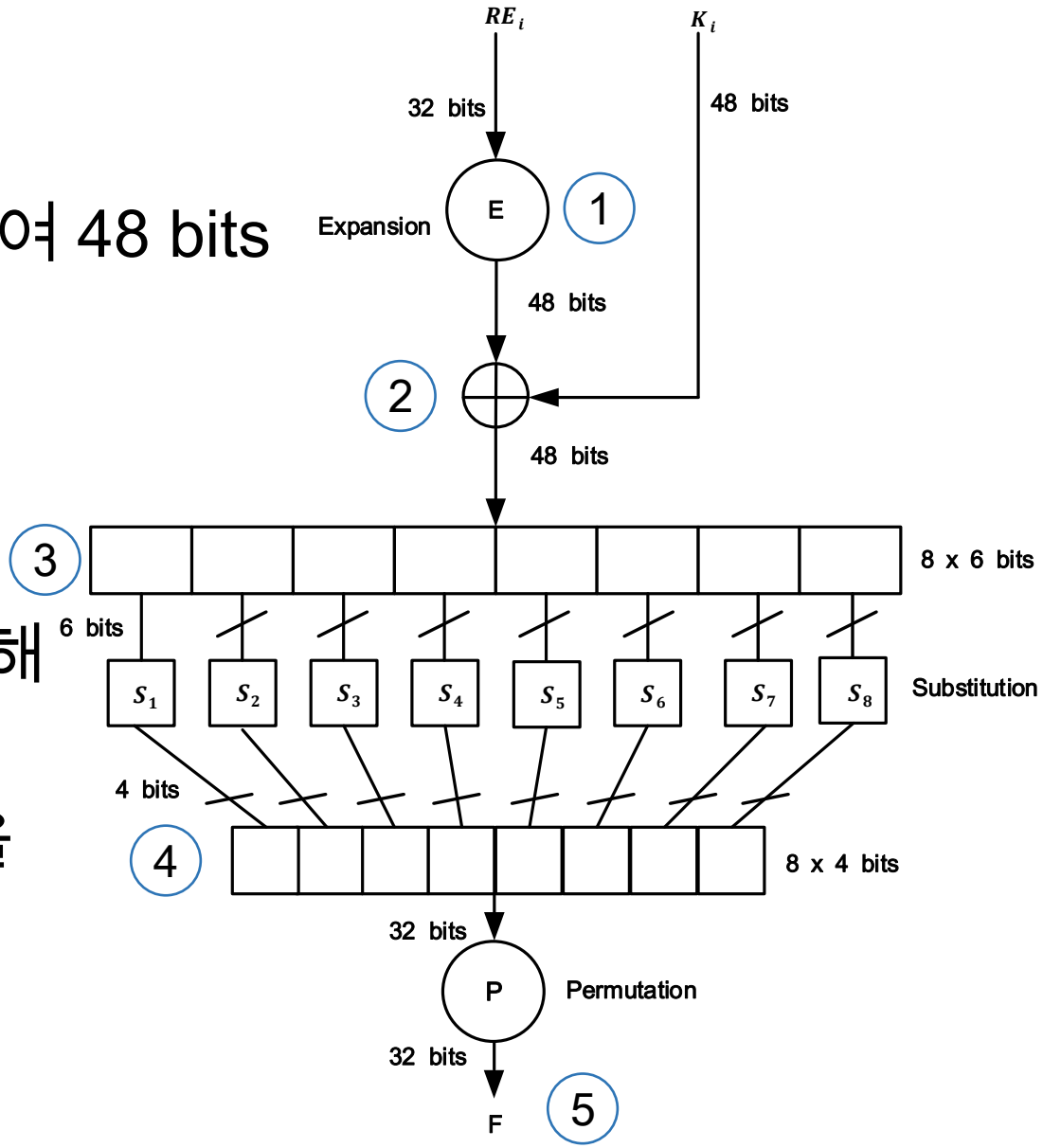
대칭 암호 원리

- DES
  - 서브키(Sub key) 생성 과정
    1. 64 bits크기의 키를 입력 받음
    2. 패리티 비트 제거(56 bits)
    3. 반으로 나누어 좌측 순환 이동을 실행
       - 1, 2, 9, 16 라운드는 1 bits
       - 나머지 라운드는 2 bits
    4. 압축
    5. 서브키 생성
    6. 위 과정을 16회 반복
  - 서브키는 모두 다르다



대칭 암호 원리

- DES
  - F함수(라운드 함수)
    1. 32 bits 값을 확장(E)하여 48 bits 값으로 만듦(P-box)
    2. XOR연산을 함
    3. 48 bits를 8개로 분할
    4. 6 bits 값이 S_1~S_8을 통해 4 bits로 변환(S-box)
    5. 최종 32 bits를 P연산을 하여 F함수 값이 나옴



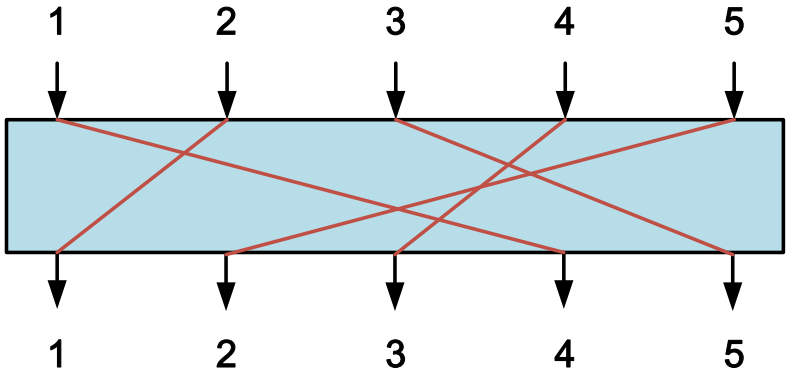
대칭 암호 원리

- DES
  - S-box(Substitution-box)
    - 6 bits의 입력을 4 bits의 출력으로 축소시켜 변환하는 함수
  - S-box Table
    - e.g., 110001(외부 bits = (1, 1), 중간 4 bits = (1000)) = 0110

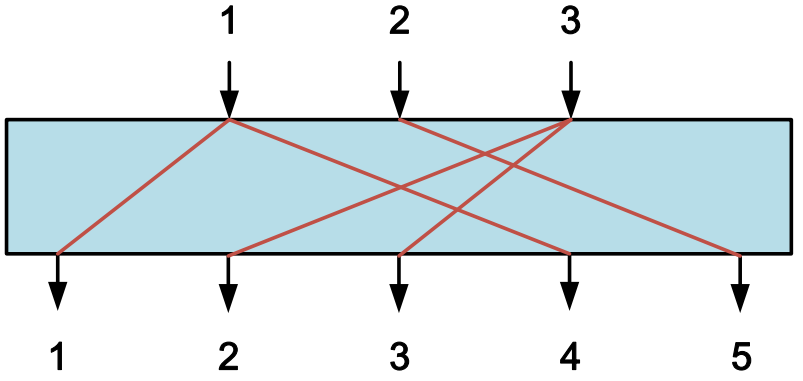
S		중간 4 bits 입력															
		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
외부 bits	00	0010	1100	0100	0001	0111	1010	1011	0110	1000	1010	0011	1111	1101	0000	1110	1001
	01	1110	1011	0010	1100	0100	0111	1101	0001	0101	0000	1111	1010	0011	1001	1000	0110
	10	0100	0010	0001	1011	1010	1101	0111	1000	1111	1001	1100	0101	0110	0011	0000	1110
	11	1011	1000	1100	0111	0001	1110	0010	1101	0110	1111	0000	1001	1010	0100	0101	0011

대칭 암호 원리

- DES
  - P-box(Permutation-box)
    - 단순(Straight) P-box
      - n bits를 입력 받아 변화된 m bits를 출력 함



    - 확장(Expansion) P-box
      - n bits를 입력 받아 변화된 m bits를 출력 함(n < m)
      - bits를 치환하고 크기를 늘릴 때 사용함

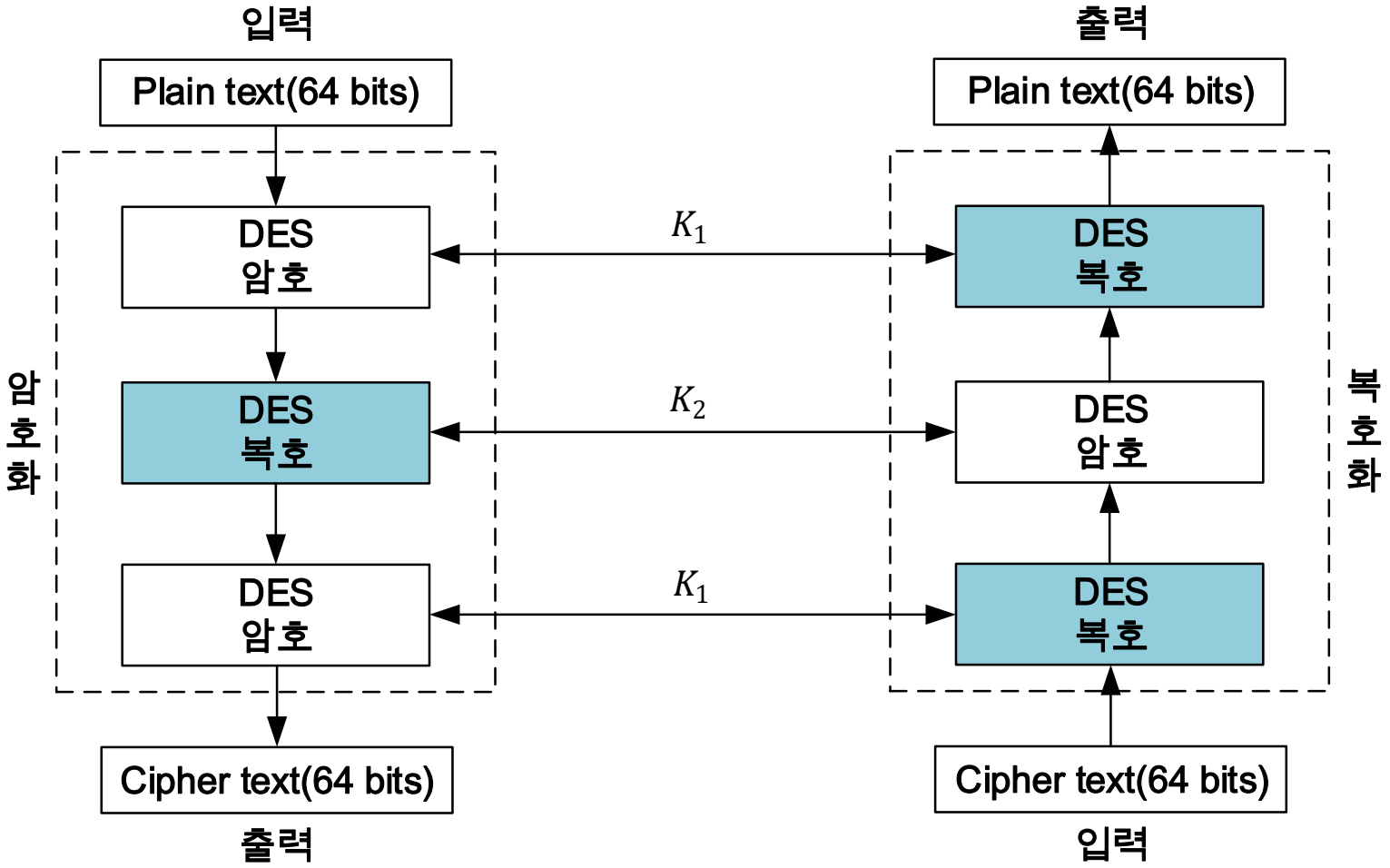


대칭 암호 알고리즘

- 3DES(Triple Data Encryption Standard)
  - DES의 안전성을 향상 시키기 위해 암호화와 복호화에 대하여 DES를 세번 사용한 알고리즘
  - 특징
    - 키의 길이가 168bits가 되기 때문에 전수 공격 예방
    - 라운드 횟수가 3배가되어 DES 보다 속도가 느림
    - 2개의 키를 갖는 3DES
      - 서브키 2개를 사용하여 DES의 암/복호화에 3회 사용
    - 3개의 키를 갖는 3DES
      - 서브키 3개를 사용하여 DES의 암/복호화에 3화 사용

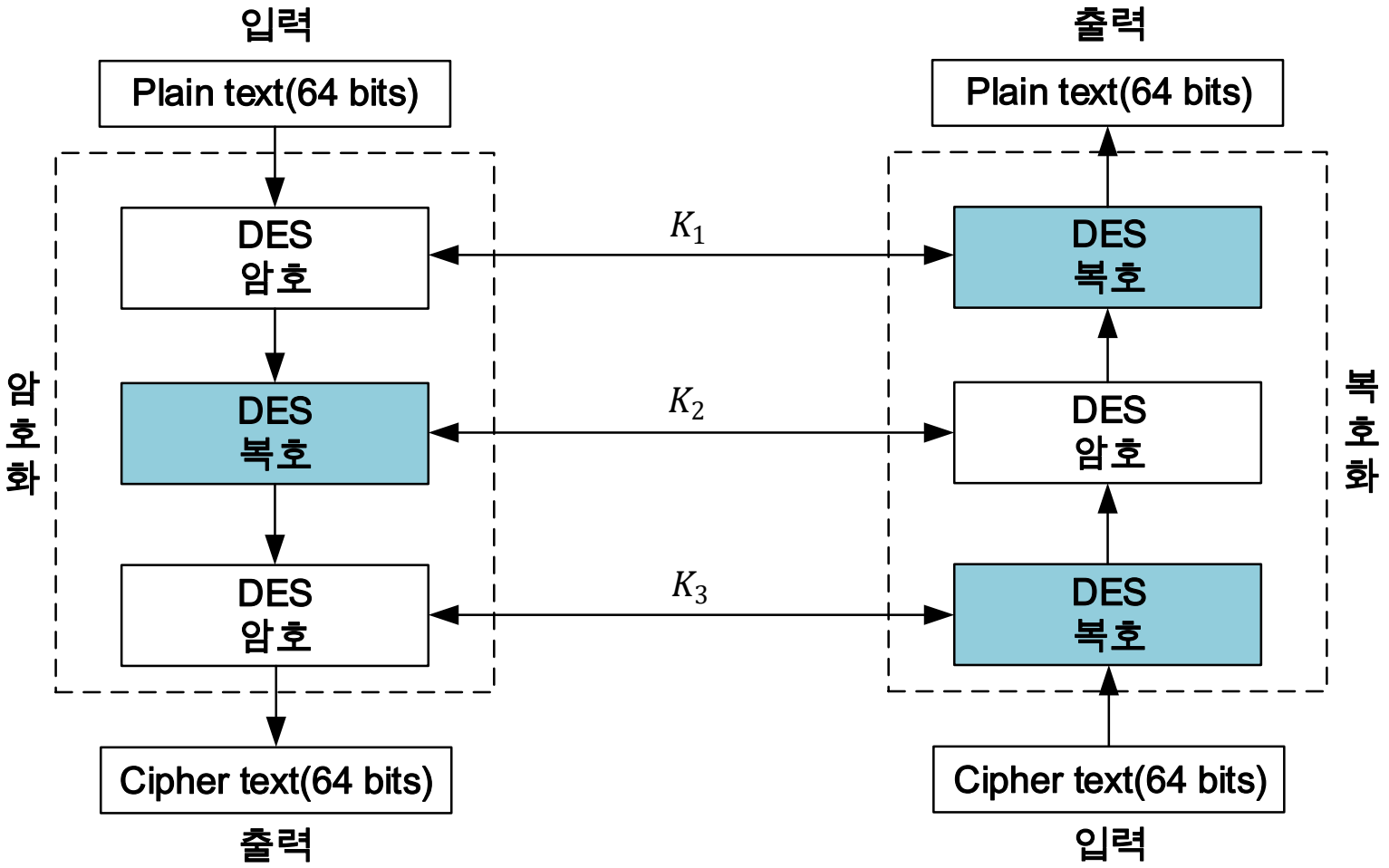
대칭 암호 알고리즘

- 3DES
  - 2개의 서브키를 갖는 3DES



대칭 암호 알고리즘

- 3DES
  - 3개의 서브키를 갖는 3DES

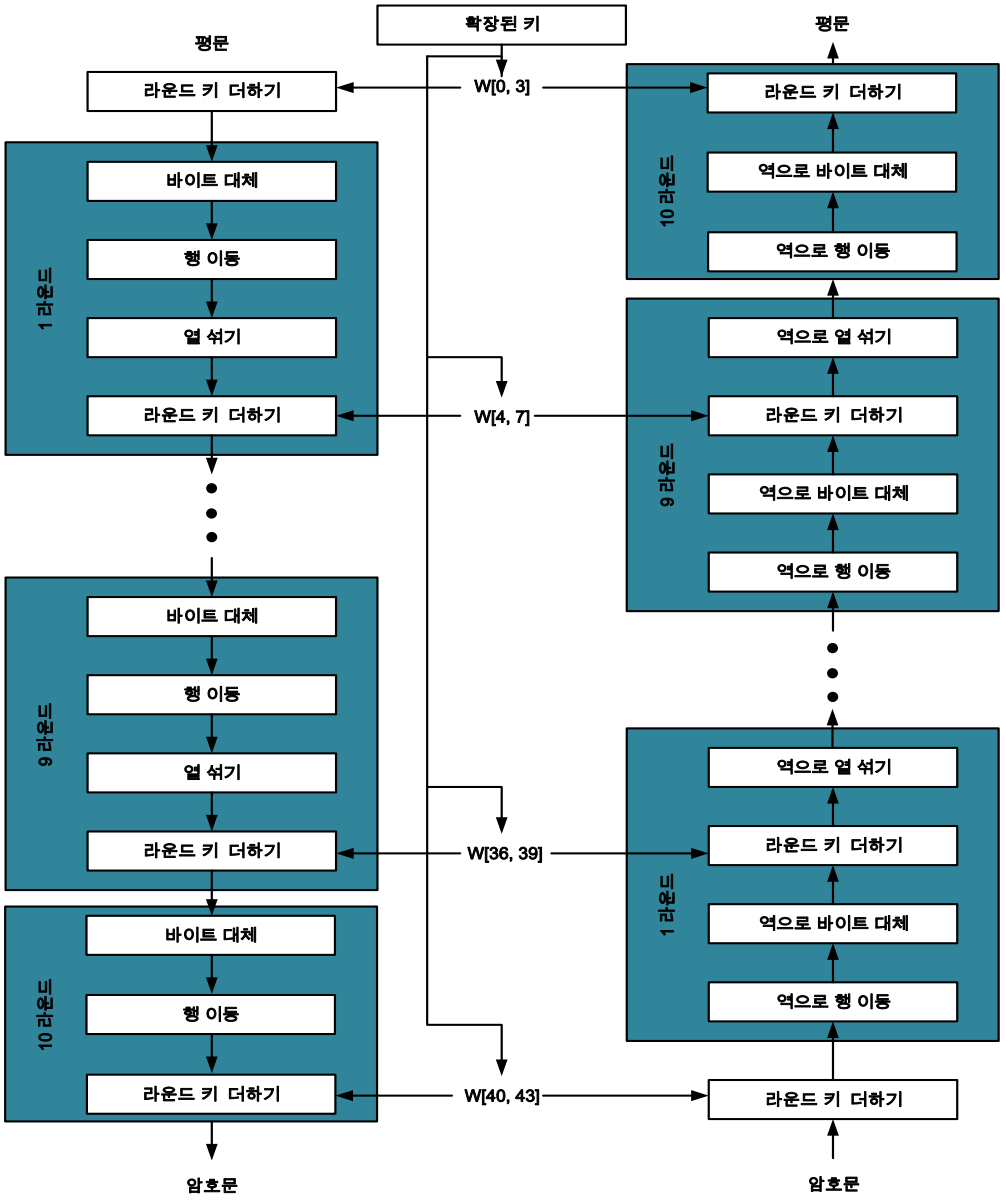


대칭 암호 원리

- AES(Advanced Encryption Standard)
  - 정의
    - 2001년 미국 국립기술표준원 (NIST)에서 공표한 대칭키 암호 알고리즘
  - 특징
    - 128bits 블록의 평문을 사용 함
    - 키의 크기는 128, 192, 256 bits 선택해서 사용 가능
    - 라운드 수는 키의 길이에 따라 다름
      - 128 bits = 10라운드, 192 bits = 12라운드, 256 bits = 14라운드
    - Feistel 암호 구조가 아님

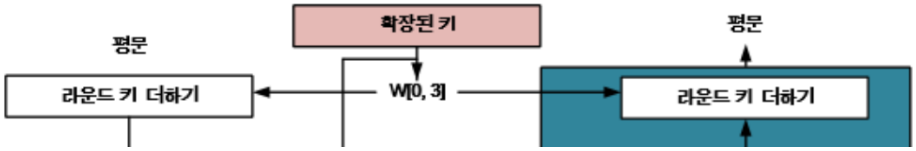
대칭 암호 알고리즘

- AES
  - 암호화 과정
    - 키를 확장하여 각 라운드에 사용
    - 입력으로 들어온 평문을 4가지 단계로 암호화
  - 복호화 과정
    - 키를 확장하여 역순으로 사용
    - 입력으로 들어온 암호문을 4가지 단계로 복호화
    - 암호화 했던 4가지 단계 연산은 전부 역연산 가능



대칭 암호 알고리즘

- AES
  - 키 확장 (Key schedule)
    - 키 확장을 통해 라운드 마다 사용되는 키 생성
      - XOR 연산
        - 새로 생성하는 라운드 키 행렬의 첫 번 째 열 생성
          - Cipher key의 첫 번째 열과 1, 2번의 수행한 결과 값과 Rcon의 라운드 수 번째 열을 XOR연산

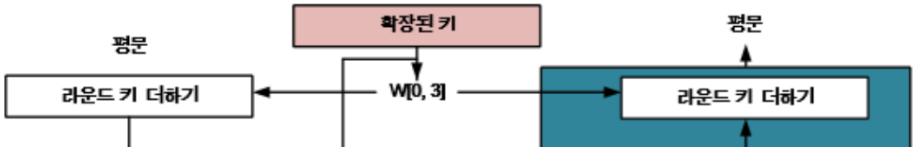


09	⊕ 8A	⊕ 01	= A0	01	02	04	08	10	20	40	80	1B	36
CF	84	00	FA	00	00	00	00	00	00	00	00	00	00
4F	EB	00	FE	00	00	00	00	00	00	00	00	00	00
3C	01	00	17	00	00	00	00	00	00	00	00	00	00

< Rcon >

대칭 암호 알고리즘

- AES



  - 키 확장 (Key schedule)
    - 키 확장을 통해 라운드 마다 사용되는 키 생성
      - XOR 연산
        - 라운드 키의 2번째 열 계산
          - Rcon과 XOR연산한 값과 Cipher key의 첫번째 열을 XOR연산
        - 라운드 키의 3, 4 번째 열 계산
          - 새로운 행렬의 열과 기존 Cipher key의 열을 XOR연산

2B	28	AB	09	A0			
7E	AE	F7	CF	FA			
15	D2	15	4F	FE			
16	A6	88	3C	17			

<Cipher key>

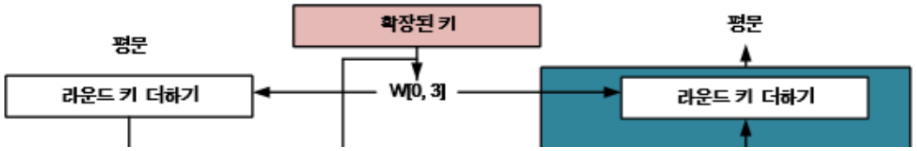
2B	28	AB	09	A0	88	23	2A
7E	AE	F7	CF	FA	54	A3	6C
15	D2	15	4F	FE	2C	39	76
16	A6	88	3C	17	B1	39	05

<Round key 1>

2B		A0		88	AB		88		23	09		23		2A
7E	⊕	FA	=	54	F7	⊕	54	=	A3	CF	⊕	A3	=	6C
15		FE		2C	15		2C		39	4F		39		76
16		17		B1	88		B1		39	3C		39		05

대칭 암호 알고리즘

- AES
  - 키 확장 (Key schedule)
    - 생성된 라운드 키



2B	28	AB	09	A0	88	23	2A	A0	88	23	2A	A0	88	23	2A
7E	AE	F7	CF	FA	54	A3	6C	FA	54	A3	6C	FA	54	A3	6C
15	D2	15	4F	FE	2C	39	76	FE	2C	39	76	FE	2C	39	76
16	A6	88	3C	17	B1	39	05	17	B1	39	05	17	B1	39	05

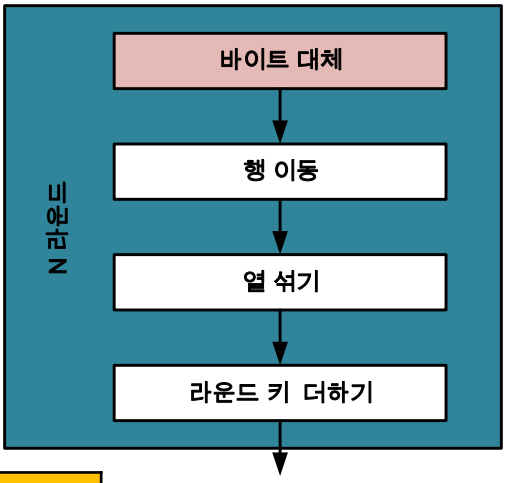
< Cipher key > < Round key 1 > < Round key 2 > < Round key 3 >

A0	88	23	2A
FA	54	A3	6C
FE	2C	39	76
17	B1	39	05

< Round key 10 >

대칭 암호 알고리즘

- AES
  - 라운드 구조 (1/4)
    - 바이트 대체 (Substitute Bytes)
      - S-box를 이용하여 bits 단위인 블록을 Byte 단위로 변환



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

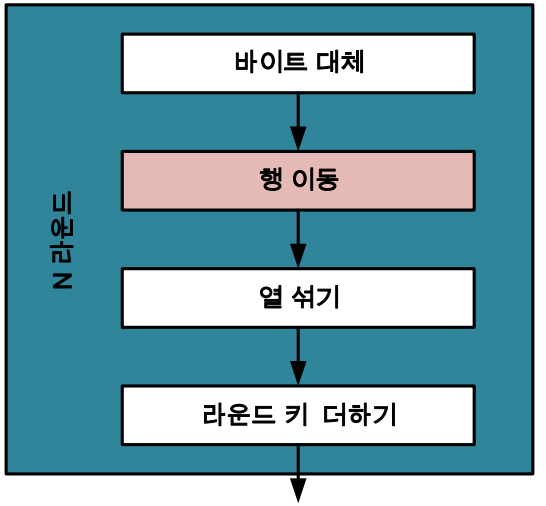
< S-box >

19	A0	9A	E9
3D	F4	C6	F8
E3	E2	8D	48
BE	2B	2A	08

D4	E0	B8	1E
27	BF	B4	41
11	98	5D	52
AE	F1	E5	30

대칭 암호 알고리즘

- AES
  - 라운드 구조 (2/4)
    - 행 이동 (Shift rows)
      - 행과 행을 Byte 단위로 치환
        - 치환을 통해 암호화 과정 평문의 모든 비트에 영향을 주기 위함
        - 1행은 그대로
        - 2행은 Left shift 1회
        - 3행은 Left shift 2회
        - 4행은 Left shift 3회

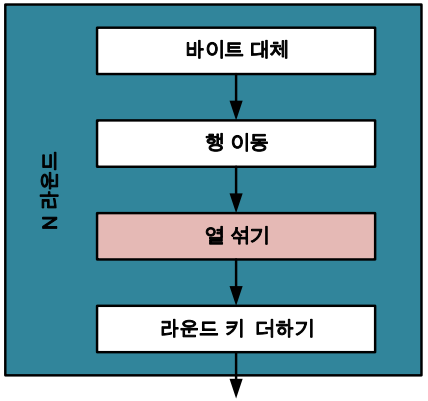


D4	E0	B8	1E
27	BF	B4	41
11	98	5D	52
AE	F1	E5	30

D4	E0	B8	1E
BF	B4	41	27
5D	52	11	98
30	AE	F1	E5

대칭 암호 알고리즘

- AES
  - 라운드 구조 (3/4)
    - 열 섞기 (Mix columns)
      - 열에 있는 각 Byte를 대체하여 변환
      - 암호가 역으로 작동되기 위해 마지막 라운드에서는 수행하지 않음



02	03	01	01
01	02	03	01
01	01	02	03
03	01	01	02

•

D4
BF
5D
30

=

04
66
81
E5

< P-box >

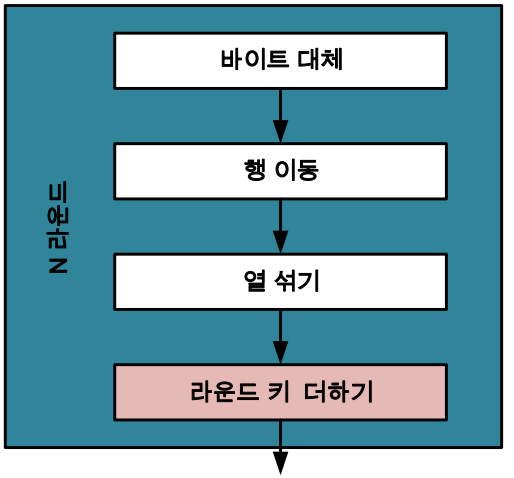
D4	E0	B8	1E
BF	B4	41	27
5D	52	11	98
30	AE	F1	E5

04	E0	B8	1E
66	B4	41	27
81	52	11	98
E5	AE	F1	E5

04	E0	48	28
66	CD	F8	06
81	19	D3	26
E5	9A	7A	4C

대칭 암호 알고리즘

- AES
  - 라운드 구조 (4/4)
    - 라운드 키 더하기 (Add round key)
      - 확장된 키와 현재 상태 배열에 있는 블록을 비트 별로 XOR연산



04	E0	48	28
66	CD	F8	06
81	19	D3	26
E5	9A	7A	4C

A0	88	23	2A
FA	54	A3	6C
FE	2C	39	76
17	B1	39	05

< Round key >

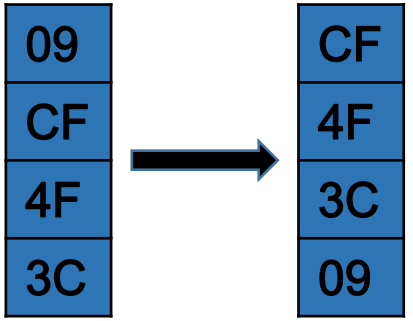
A4	68	6B	02
9C	9F	5D	6A
7F	35	EA	50
F2	2B	43	49

04		A0	=	A4
66	⊕	FA	=	9C
81		FE	=	7F
E5		17	=	F2

대칭 암호 알고리즘

- AES
  - 키 확장 (Key schedule)
    - 키 확장을 통해 라운드 마다 사용되는 키 생성

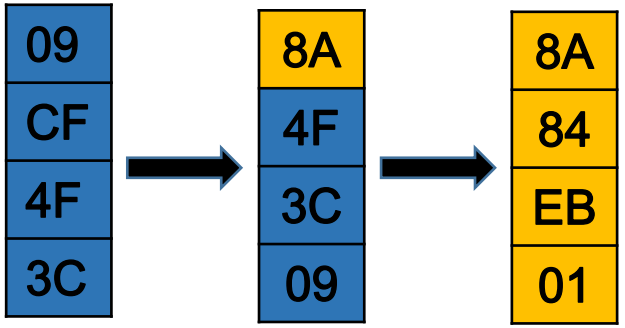
1. 열의 이동 (Shift column)



2. 바이트 대체 (Substitute bytes)

2B	28	AB	09
7E	AE	F7	CF
15	D2	15	4F
16	A6	88	3C

< Cipher key >



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

대칭 암호 알고리즘

- DES와 AES 차이

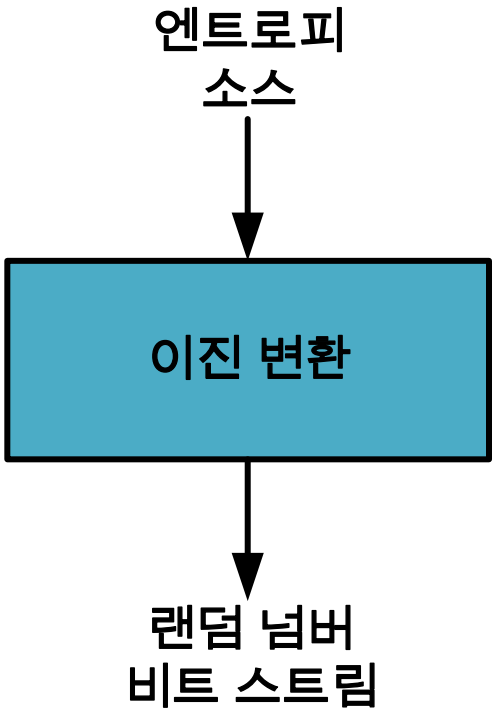
항목	DES	AES
데이터 처리	데이터 블록을 절반으로 나누어 처리	전체 데이터 블록을 단일 행렬로 처리
구조	Feistel 암호 구조	대체 및 치환 구조
평문 크기	64 bits	128bits
키 크기	56 bits	128, 192, 256 bits
라운드 수	16라운드	10라운드, 12라운드, 14라운드
속도	AES에 비해 느림	DES에 비해 빠름

랜덤 넘버와 의사 랜덤 넘버

- 랜덤 넘버(Random Number)
  - 정의
    - 비트 크기의 같은 범위 내에서 무작위로 추출된 수
  - 특징
    - 무작위성
      - 수열이 어느 한 쪽으로 치우치지 않고 무작위로 분포 되어야 함
      - 균등 분포(Uniform distribution)
        - 비트열에 나타나는 0과 1이 나타나는 빈도가 비슷해야 함
      - 독립성(Independence)
        - 수열에서 추출한 부분 수열이 다른 수열로부터 추측할 수 없어야 함
    - 예측불가능성
      - 수열의 일부를 보고 이어지는 수를 예측할 수 없어야 함

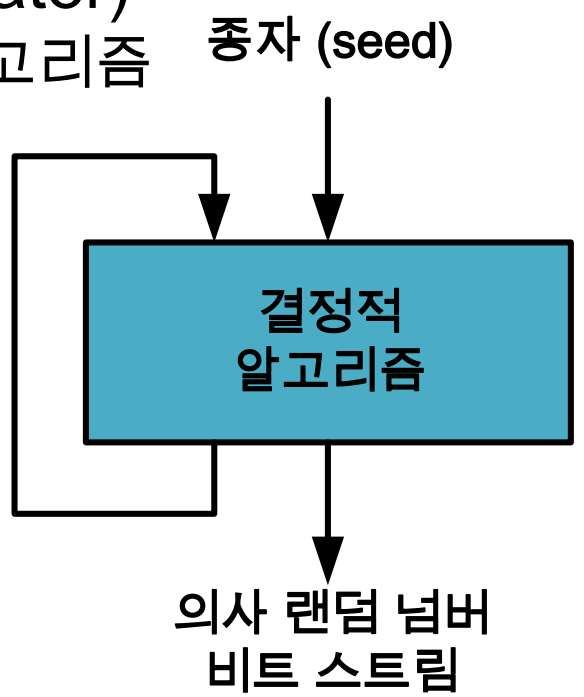
랜덤 넘버와 의사 랜덤 넘버

- 랜덤 넘버
  - 진성 랜덤 넘버(True Random Number)
    - 입력 값이 랜덤한 정보
    - 엔트로피 소스(Entropy Source)
      - 컴퓨터에서 물리적으로 얻을 수 있는 랜덤 정보
        - e.g., 키 입력 타이밍 패턴, 마우스 움직임 등
    - TRNG(True Random Number Generator)
      - 엔트로피 소스를 입력으로 랜덤한 값을 출력



랜덤 넘버와 의사 랜덤 넘버

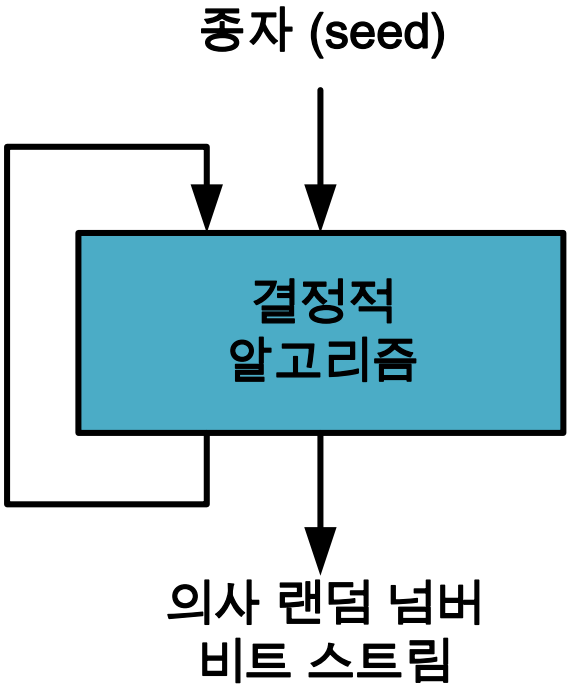
- 랜덤 넘버
  - 의사 랜덤 넘버(Pseudo Random Number)
    - 특정 알고리즘에 기초하여 생성되고 무작위성 테스트를 통과 할 수 있는 수열
  - PRNG(Pseudo Random Number Generator)
    - 무한 비트열을 생성 하기 위해 사용 되는 알고리즘
    - 종자(Seed)는 고정된 값
      - 입력 받은 값
    - 종자 값에 따라 난수 결정
    - 결정적 알고리즘 사용
      - 종자 값이 같으면 출력 값이 같은 알고리즘
      - e.g.,중앙 제곱법



랜덤 넘버와 의사 랜덤 넘버

- PRNG
  - 중앙 제곱법

Seed	제곱	난수
1234	$(1234)^2 = 1522756$	5227
5227	$(5227)^2 = 27321529$	3215
3215	$(3215)^2 = 10336225$	3362



Thanks!

박 재 형 (jaehyoung@pel.sejong.ac.kr)