암호 관련 기초

제2장. 대칭 암호와 메시지 기밀성

암호 (1)

- Cryptology
  - Kryptos logo : 숨겨진 말 (그리스어)
  - 암호기법
    - 수학적으로 풀이하기 어렵게 하기 위한 기법
  - 암호분석(cryptanalysis)
    - 암호 메커니즘을 연구하여 해독하는 방법 연구
  - 암호학(Cryptology) = 암호기법 + 암호분석
- 암호화(encryption)
  - 키가 있어야만 내용을 볼 수 있도록 변환시키는 과정
- 복호화(decryption)
  - 키를 이용해 원문으로 되돌리는 변환 과정
- 대칭키(또는 비밀키) : 2장
  - 암호화와 복호화에 사용되는 키가 동일
- 비대칭키(또는 공개키) : 3장
  - 서로 다른 키를 두 과정에서 이용

암호 (2)

- 키의 명칭

키 형태	대칭(symmetric) 키	암호화/복호화 과정에서 동일한 키 사용
	비대칭(asymmetric) 키	암호화/복호화 과정에서 서로 키 사용
키 용도	비밀(secret) 키	대칭 암호 알고리즘의 키
	공개(public) 키	비대칭 암호 알고리즘에서 공개된 키
	개인(private) 키	비대칭 암호 알고리즘에서 공개하지 않는 키
	세션(session) 키	세션 연결 중에 사용되는 키
	인증(authentication) 키	사용자 및 시스템 인증 용도
키 교환 형태	사전 공유 (pre-shared) 키	미리 오프라인에서 합의한 키
	수동 키	임의적인 방법으로 수동 교환
	자동 키	동적인 방법으로 교환

암호의 필요성

- 암호화를 통해 제공되는 기능
  - 기밀성 보장
    - 키가 있는 사용자만이 내용을 볼 수 있으므로 기밀성 제공
  - 인증성 보장
    - 키의 소유 여부로 메시지의 출처가 명확해짐
  - 무결성 보장
    - 해쉬 알고리즘으로 계산된 값을 전송함으로써 위변조시 내용 변경을 인지할 수 있음
  - 부인방지 보장
    - 키의 소유로 해당 메시지를 송신한 자가 부인할 수 없게 함

암호의 역사

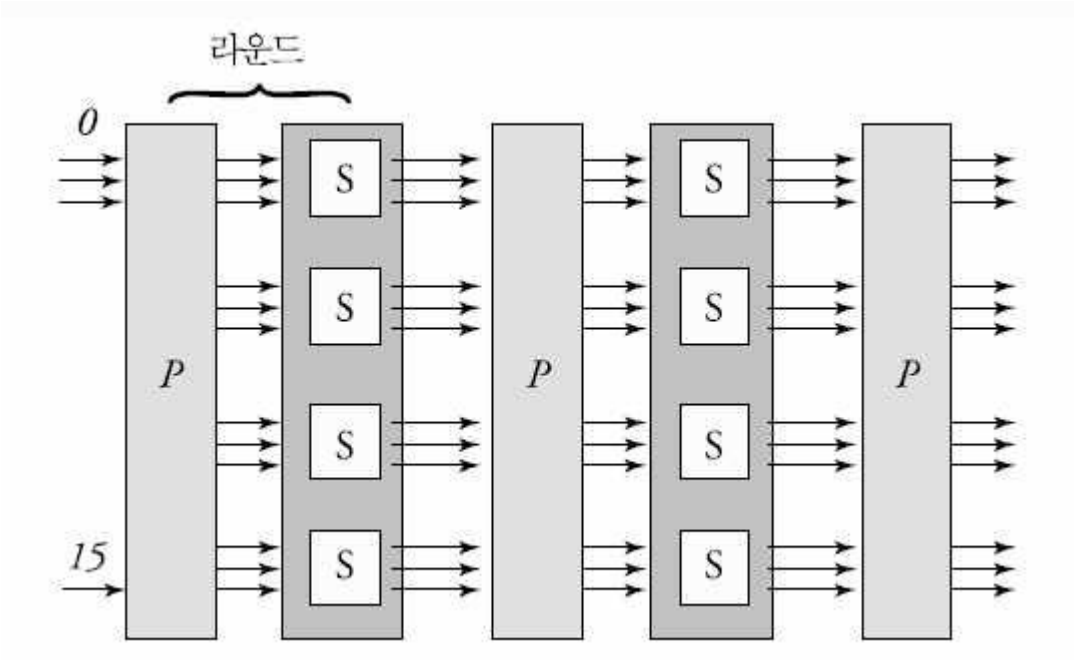
- 고대
  - 시저 암호(Caesar cipher)
    - 대체(substitution) 암호
  - 비즈네르(Vigenere) 암호
  - 뷰포트(Beaufort) 암호
- 근대
  - 복잡도를 높이기 위해 기계 사용
  - 독일의 ENIGMA
  - 미국의 M-209
- 현대
  - 컴퓨터 산업의 발달과 수학적 배경을 기본으로 발전
  - 대칭키 : DES
  - 공개키 : RSA, ElGamal, ECC

고대/근대 암호 (1)

- 대체 암호 (substitution)
  - 한 문자를 다른 문자로 대체
  - 한 문자에 대해 여러 문자로 대체되는 경우 : 다중 대체 암호
  - 분류
    - Monoalphabetic
      - Additive 암호 (shift 암호, Caesar 암호)
      - Multiplicative 암호(Affine 암호)
    - Polyalphabetic
      - Autokey 암호
      - Plyafair 암호
      - Vegenere 암호
      - Hill 암호
      - One time pad 암호
      - Rotor 암호
  - 대응하는 문자들이 중복되어서는 안된다.
  - 알파벳 26자를 순서대로 늘어놓는 순열 = 26! = $4 * 10^{26}$
  - 전수 조사는 어렵지만, 통계적 암호 공격에 취약

고대/근대 암호 (2)

- 대체 암호

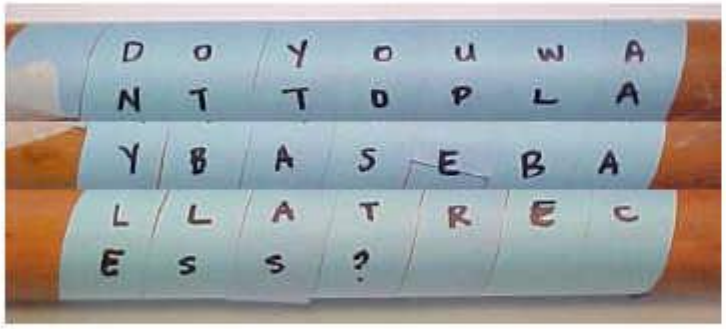


P(평문) $= C$(암호문) $= Z_{26}$, $0 \leq K \leq 25$일때 각 Permutation $\pi \in K$,

$E_\pi(x) = \pi(x)$, 그리고 $D_\pi(y) = \pi^{-1}(y)$, 여기서 π^{-1}은 inverse permutation이다.

고대/근대 암호 (3)

- 스키타일 암호



고대/근대 암호 (4)

- Shift 암호
  - 시저에 의해 기인
  - Monoalphabetic 중 Additive 암호로 분류

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

$P(\text{평문}) = C(\text{암호문}) = K(\text{키}) = Z_{26}$, $0 \leqq K \leqq 25$일 때 $E_K(x) = x + K \bmod 26$, 그리고 $D_K(y) = y - K \bmod 26 \, (x, y \in Z_{26})$

  - 최대 26회만 shift하면 해독 가능

고대/근대 암호 (5)

- Shift 암호 (계속)

[예] 평문 : SHIFT CIPHER

먼저 평문을 정수로 표현하면 다음과 같다.

평문 : SHIFT CIPHER
정수 : 18 7 8 5 19 2 8 15 7 4 17

평문 정수값에 키 값인 '10'을 각 정수에 더하여 mod 26을 하면 다음과 같다.
정수 : 2 17 18 15 3 12 18 25 17 14 1

암호문 : CRSPDMSZROB

결과적으로 암호문은 'CRSPDMSZROB'가 된다.

고대/근대 암호 (6)

- 시저 암호
  - 키 : client

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
c	l	i	e	n	t	a	b	d	f	g	h	j	k	m	o	p	q	r	s	u	v	w	x	y	z

  - 예 : kill him today
    - gcbb bdj smecy

고대/근대 암호 (7)

- Affine 암호

P(평문) $= C$(암호문) $= Z_{26}$이라 하고, $K = \{(a,b) \in Z_{26} \times Z_{26} : \gcd(a,26) = 1\}$라 할 경우, $K = (a,b) \in k$일 때, $E_K(x) = ax + b \bmod 26$이며, $D_K(y) = a^{-1}(y-b) \bmod 26 \; (x,y \in Z_{26})$가 된다(단, $\gcd(a,26) = 1$이어야만 유일한 b의 값을 갖는다. 따라서 이 조건을 항상 만족해야만 한다).

  - 여기서 a는 26 과 서로 소 (2와 13의 배수가 아니며, a -1 { 1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23, 25 }
    - 9 * 3 = 27 => 1 (mod 26)
    - 21 * 5 = 105 => 1 (mod 26)

고대/근대 암호 (8)

- Affine 암호 (계속)

[예] $K=(a,b)\in k$로, K가 $K=(9,2)$일 때, $9^{-1} \bmod 26 = 3$이다. 암호화 함수는 $E_K(x) = 9x+2$이고 복호화 함수는 $D_K(y) = 3(y-2) = 3y-6 \equiv 3y+20 \equiv x$이다.
여기서 모든 연산은 Z_{26}에서 실행된다. 이 결과를 검증하면 $D_K(E_K(x)) = x, x\in Z_{26}$가 되어야 한다. 평문 'Affine'을 암호화하려면, 우선 알파벳들을 정수로 표현한다. 정수는 다음과 같다. 0, 5, 5, 8, 13, 4. 이 정수들을 수식을 통해 암호화하면 다음과 같다.

$9\times 0+2 \bmod 26 = 2 \bmod 26 = 2$
$9\times 5+2 \bmod 26 = 47 \bmod 26 = 21$
$9\times 5+2 \bmod 26 = 47 \bmod 26 = 21$
$9\times 8+2 \bmod 26 = 74 \bmod 26 = 22$
$9\times 13+2 \bmod 26 = 119 \bmod 26 = 15$
$9\times 4+2 \bmod 26 = 38 \bmod 26 = 12$

암호문 '2, 21, 21, 22, 15, 12'를 생성하고, 이에 해당되는 알파벳은 'CVVWPM'이 된다. 복호화는 수식 $x \equiv D_K(y) \equiv 3(y)+20$을 이용해 연산한다.

고대/근대 암호 (9)

- Vigenere 암호
  - 복잡한 표를 미리 만들고, 이에 따라 암호를 조립하거나 푸는 방법

m을 양의 정수라 가정하고, $P=C=K=(Z_{26})^m$, $K=(k_1,k_2,\cdots\cdots,k_m)$으로 수식으로는 $K=(k_1,k_2,\cdots\cdots,k_m)$

$$E_K(x_1,x_2,\cdots\cdots,x_m)=(x_1+k_1,x_2+k_2,\cdots\cdots,x_m+k_m)$$
$$D_K(y_1,y_2,\cdots\cdots,y_m)=(y_1-k_1,y_2-k_2,\cdots\cdots,y_m-k_m)$$

가 되며, 모든 연산은 Z_{26}으로 수행된다.

※ m = 키 길이, K = 키, E = 암호화, D = 복호화, X = 평문, Y = 암호문

고대/근대 암호 (10)

- Vigenere 암호 (계속)

[예] 키 길이가 m=6이고, keyword가 'CIPHER'라 하면, K=(2, 8, 15, 7, 4, 17)가 된다.

평문 : vigenere cipher

위의 평문을 6개의 키를 이용해 암호화하면 다음과 같다.

평문	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
정수	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

평문 : vigenere cipher
정수 : 21,8,6,4,13,4,17,4,2,8,15,7,4,17

21	8	6	4	13	4	17	4	2	8	15	7	4	17
2	8	15	7	4	17	2	8	15	7	4	17	2	8
23	16	21	11	17	21	19	12	17	15	19	24	6	25

정수 : 23,16,21,11,17,21,19,12,17,15,19,24,6,25
암호문 : xqvlrvtmrptygz

고대/근대 암호 (11)

- 행렬 암호 (Hill Cypher)
  - 행렬을 이용한 암호화 기법

평문 $m = m_1 m_2 m_3 \cdots m_d$라 가정할 때, 암호 키가 가역 행렬(*invertible matrix*) $K \in P_{d\times d}(Z_{26})$의 $M_{d\times d}(Z_{26})$은 원소 $\{0,1,2,3,\ldots 25\}$에서 택하는 $d \times d$ 행렬들의 집합을 의미하고, 이를 역행렬을 K^{-1}라 한다. 암호화는 평문 m을 행벡터 $P=(m_1 m_2 m_3 \cdots m_d)$로 나타낸 후, P과 암호키 K를 곱해 암호문 C를 얻게 된다. m은 양의 정수이며, $P=C=(Z_{26})^m$와 $K=\{m \times m$ 행렬$\}$이라 놓는다.

$$C \equiv PK \equiv (m_1 m_2 m_3 \cdots m_d) \begin{vmatrix} k_{11} k_{12} \cdots k_{1d} \\ \cdot \\ \cdot \\ k_{d1} k_{d2} \cdots k_{dd} \end{vmatrix} (\mathrm{mod}\, 26)$$

복호화 과정은 암호화의 역 과정으로 암호문 C에 역행렬 K^{-1}을 곱하여 다음 수식과 같이 평문을 구한다.

$$CK^{-1} \equiv PKK^{-1} \equiv P (\mathrm{mod}\, 26)$$

결과적으로 K에 대해서 $E_K(x) = xK$이고 $D_K(y) = yK^{-1}$이다. 모든 연산은 Z_{26}에서 수행된다.

※ d = 블록 크기, P = 평문, C = 암호문, K–1 = 역행렬, E = 암호화, D = 복호화

고대/근대 암호 (12)

- 치환 (permutation) 암호

m을 고정된 양의 정수라 가정하고, $P \equiv C \equiv (Z_{26})^m$ K는 $\{0,1,\cdots,m\}$의 모든 permutation으로 구성된다. 키가 π일 때, 암호화는 $E_\pi(P_1,\cdots,P_m) = (P_{\pi(1)},\cdots,P_{\pi(m)})$가 되고, 복호화는 $D_\pi(C_1,\cdots,C_m) = (C_{\pi^{-1}(1)},\cdots,P_{\pi^{-1}(m)})$이 된다($\pi^{-1}$는 Inverse Permutation이다).

[예] m=6이고, 다음의 permutation π을 키라고 한다면, 다음과 같은 순서로 문자열을 암호화한다.

①

1	2	3	4	5	6
3	5	1	6	4	2

②

s	h	e	s	e	1
3	5	1	6	4	2

③

s	h	e	s	e	1
e	e	s	1	s	h

④

1	2	3	4	5	6
e	e	s	1	s	h

그리고 암호화된 문자열은 inversion permutation π^{-1}을 통해 다음과 같은 순서로 문자열을 복호화한다.

①

1	2	3	4	5	6
3	6	1	5	2	4

②

e	e	s	1	s	h
3	6	1	5	2	4

③

e	e	s	1	s	h
s	h	e	s	e	1

④

1	2	3	4	5	6
s	h	e	s	e	1

고대/근대 암호 (13)

- 스트림 암호

$Z_1 = K, z_i = P_{i-1} (i \geq 2)$, $0 \leq z \leq 25$라고 할 때, $E_z(P) = P + k \bmod 26$, 그리고 $D_z(C) = C - Z_i \bmod 26$이다(단, $(P, C \in Z_{26})$이다).

[예] 키 길이는 $K=5$이고, 평문이 'CIPHER'일 때 암·복호화에 대해서 알아본다. 먼저 평문을 정수로 변환하면 2, 8, 15, 7, 4, 17이 되고, 키 스트림은 아래와 같다.

$z_1 = K$	$z_2 = P_1$	$z_3 = P_2$	$z_4 = P_3$	$z_5 = P_4$	$z_6 = P_5$
5	2	8	15	7	4

그리고 평문과 이에 해당하는 키 스트림을 더하여 modulo 26을 하면, 암호 정수 7, 10, 23, 22, 11, 21로 변환되며, 이 정수들을 다시 문자로 변환하면 'HKXWLVM'으로 암호화 된다. 만약 다시 복호화 하려면, 암호화 과정의 역순을 취하면 된다. 암호문을 정수로 변환하면 7, 10, 23, 22, 11, 21이 되고, $P_1 = D_5(7) = 7 - 5 \bmod 26 = 2$, $P_2 = D_2(10) = 10 - 2 \bmod 26 = 8$, 이와 같은 수식에 의해 P_{10}까지 실행한 후, 다시 정수를 문자로 변환하면 된다. 계산 과정에서 각 실행마다 계산되는 평문의 정수 값은 다음 평문을 계산하는 키 스트림의 원소로 사용하게 된다.