

1장 : 암호이론과 보안개요

정보보호이론

Spring 2015

■ 교과서

- ✕ 현대암호학 개론 (이론출판사, 비매용)

- ▶ Some dazing math. things will be introduced only when needed

■ 참고서 :

- ✕ Cryptography and Network Security, Behrouz A. Forouzan, McGrawHill 2008

■ TA : not assigned yet.

■ Course Materials will be uploaded on ECU of university portal.

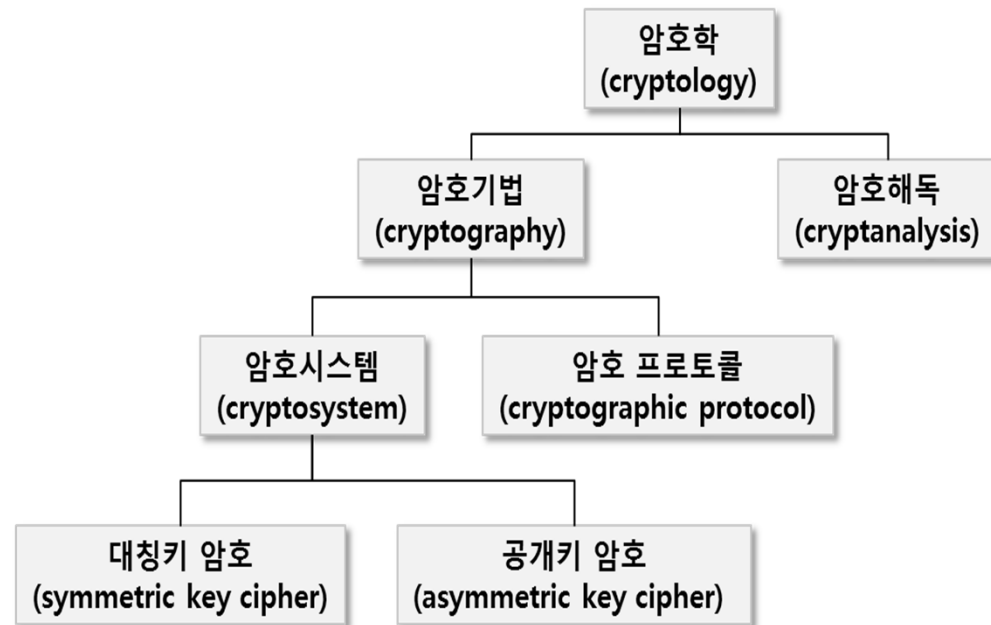
1-1 암호학 소개

■ 암호학(cryptology)

- ✕ 암호학은 보안시스템의 가장 중요한 부분이지만 그 자체로는 쓸모가 없다.
 - ▶ 웹(web)의 취약점을 찾는 공격자는 암호를 공격하지 않고도 버퍼 오버플로우(buffer overflow) 등을 이용하여 공격
 - ▶ 즉 "A security system is only as strong as its weakest link."

1-1 암호학 소개

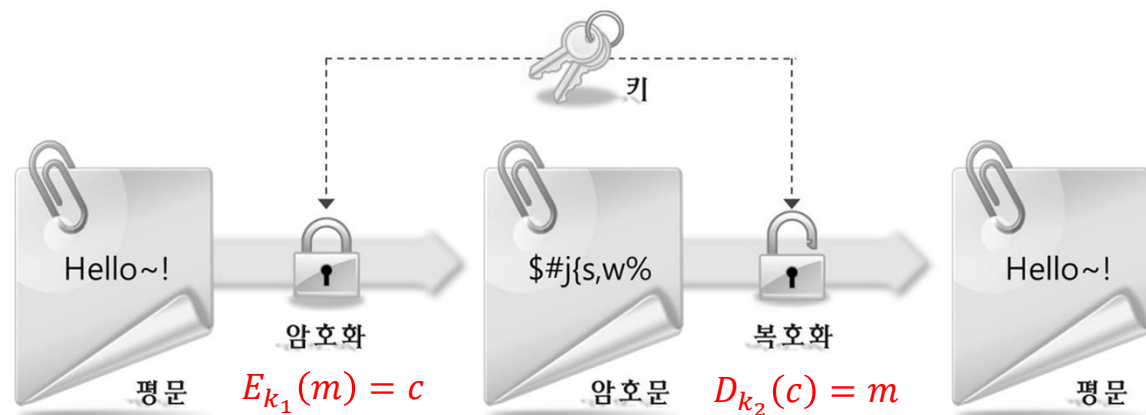
■ 암호학(cryptology)의 분류



1-1 암호학 소개

■ 암호기법(cryptography)

- ✕ 그리스어로 “비밀(secret)”을 의미하는 kryptos와 “쓰다(write)”를 의미하는 gráphō의 합성어
- ✕ 즉 메시지의 기밀성(confidentiality)을 제공하기 위하여 사용. 현재는 메시지를 공격자로부터 안전하게 보호하기 위하여 메시지를 변화하는 과학이나 기술을 의미.
- ✕ Key : uniformly distributed random string
- ✕ Symmetric if $k_1 = k_2$, Otherwise, asymmetric



Correctness: $D_{k_2}(E_{k_1}(m)) = m$

1.1.2 Kerckhoff's Principle

- Kerckhoffs의 원리 : 암호 알고리즘은 알고리즘의 모든 내용이 공개되어도 키가 노출되지 않으면 안전해야 한다.

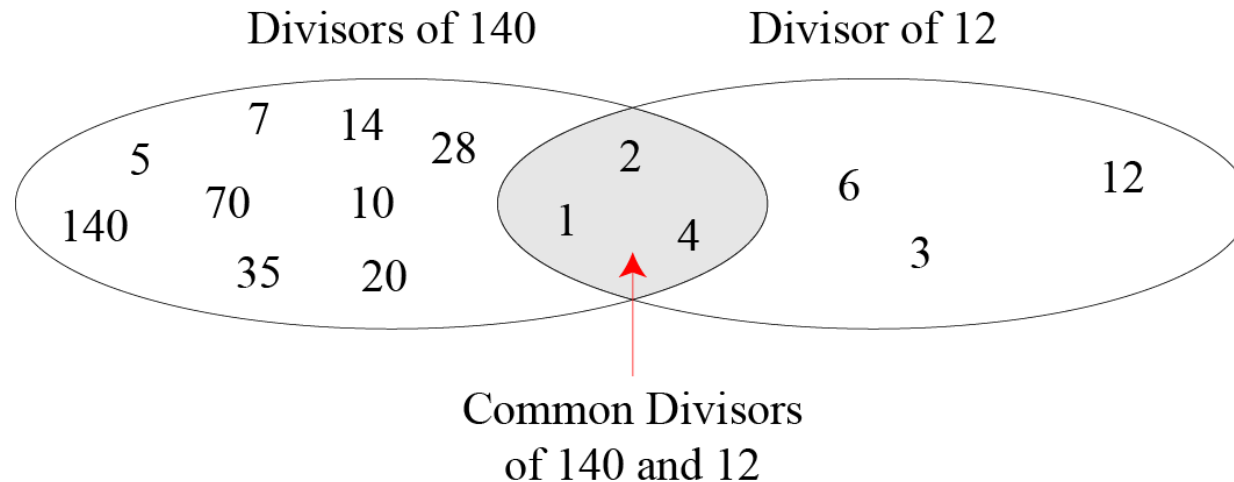
1. 짧은 길이의 키를 안전하게 보관하는 것은 키 보다 수천배의 사이즈인 암호 알고리즘 전체를 안전하게 보관하는 것 보다 용이. 또한 암호시스템은 역공학 등으로 노출될 수 있지만 키는 보통 난수이어서 역공학에 안전
2. 키가 노출되었을 때 키를 변경하는 것이 새로운 암호시스템을 설계하는 것보다 훨씬 용이
3. 암호시스템은 보통 다수의 사용자를 위하여 운영되며, 모든 사용자는 동일한 암호 알고리즘을 사용. 이 경우 암호 통신을 하는 당사자들마다 상이한 암호시스템을 사용하는 것 보다는 동일한 암호시스템을 사용하면서 키만 다르게 설정하는 것이 실용적. ➔ 표준화
4. 내부자나 역공학에 의하여 암호시스템이 공개되면 새로운 암호 알고리즘을 설계.

1.2 수학적 배경지식

■ 약수, 공약수, 최대공약수(GDC: Greatest Common Divisor)

✗ $\gcd(a, b) = \gcd(140, 12) = 4$

✗ 0이 아닌 두 정수 a, b 에 대하여, $\gcd(a, b) = 1$ 을 만족하면 a 와 b 는 서로소(relatively prime)



1.2 수학적 배경지식

■ 유클리드 알고리즘(Euclidean Algorithm)

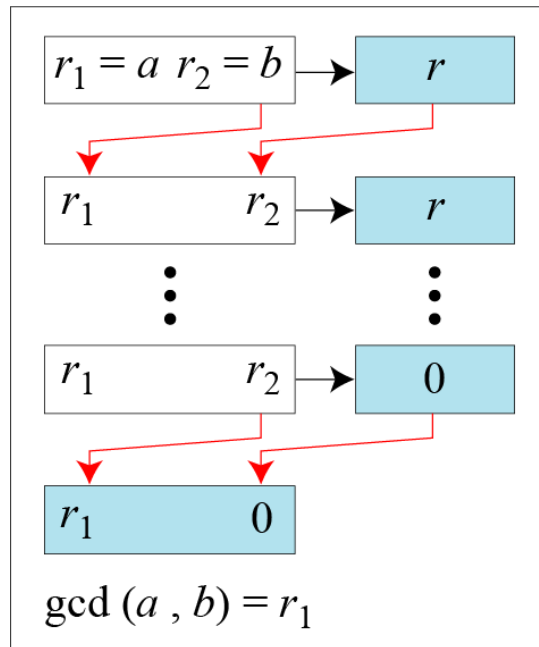
Fact 1: $\gcd(a, 0) = a$

Fact 2: $\gcd(a, b) = \gcd(b, r)$, where r is the remainder of dividing a by b

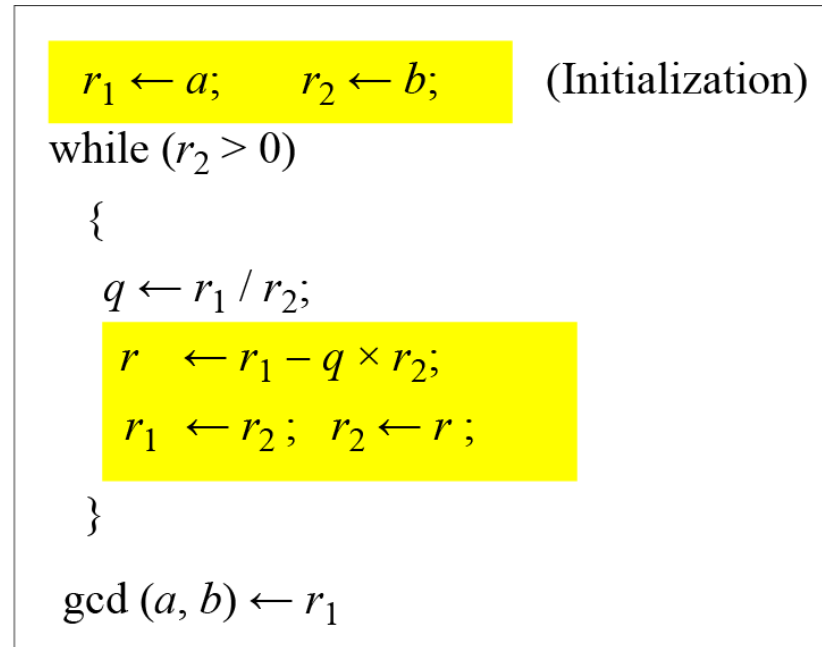
q	r_1	r_2	r
1	2740	1760	980
1	1760	980	780
1	980	780	200
3	780	200	180
1	200	180	20
9	180	20	0
	20	0	

1.2 수학적 배경지식

■ 유클리드 알고리즘(Euclidean Algorithm)



a. Process



b. Algorithm

1.2 수학적 배경지식

■ 확장 유클리드 알고리즘(Extended Euclidean Algorithm)

- ✕ 적어도 하나는 0이 아닌 두 정수 a 와 b 에 대하여 다음을 만족하는 s 와 t 가 존재한다.

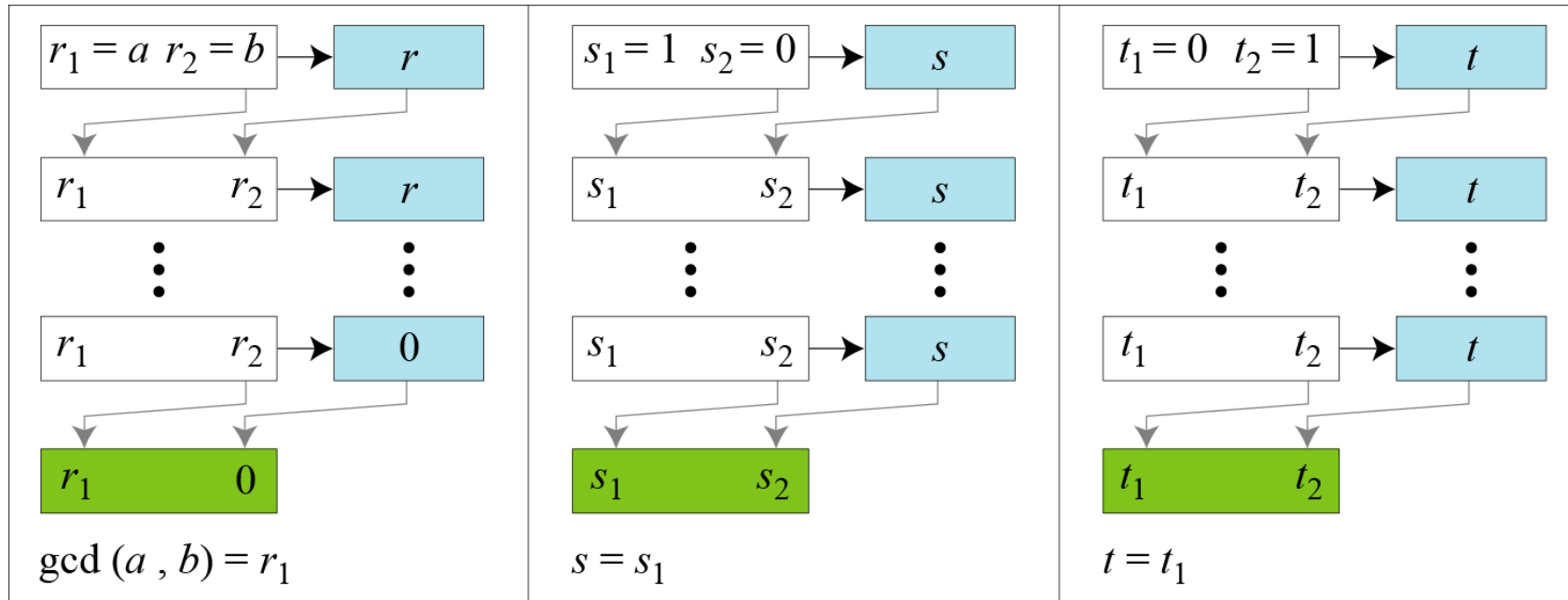
$$s \cdot a + t \cdot b = \gcd(a, b)$$

- ✕ $a = 75, b = 20$ 인 경우

$$75 \cdot (-1) + 20 \cdot 4 = \gcd(75, 20) = 5$$

확장 유클리드 알고리즘은 $\gcd(a, b)$ 뿐만 아니라 s 와 t 를 구해준다.

1.2 수학적 배경지식



a. Process

$$r = r_1 - q \times r_2, \quad s = s_1 - q \times s_2, \quad t = t_1 - q \times t_2$$

1.2 수학적 배경지식

```
 $r_1 \leftarrow a; \quad r_2 \leftarrow b;$   
 $s_1 \leftarrow 1; \quad s_2 \leftarrow 0;$   
 $t_1 \leftarrow 0; \quad t_2 \leftarrow 1;$ 
```

(Initialization)

while ($r_2 > 0$)

{

$q \leftarrow r_1 / r_2;$

$r \leftarrow r_1 - q \times r_2;$

$r_1 \leftarrow r_2; \quad r_2 \leftarrow r;$

(Updating r 's)

$s \leftarrow s_1 - q \times s_2;$

$s_1 \leftarrow s_2; \quad s_2 \leftarrow s;$

(Updating s 's)

$t \leftarrow t_1 - q \times t_2;$

$t_1 \leftarrow t_2; \quad t_2 \leftarrow t;$

(Updating t 's)

}

$\text{gcd}(a, b) \leftarrow r_1; \quad s \leftarrow s_1; \quad t \leftarrow t_1$

b. Algorithm

1.2 수학적 배경지식

- Ex : Given $a = 161$ and $b = 28$, find $\gcd(a, b)$ and the values of s and t .

q	r_1	r_2	r	s_1	s_2	s	t_1	t_2	t
5	161	28	21	1	0	1	0	1	-5
1	28	21	7	0	1	-1	1	-5	6
3	21	7	0	1	-1	4	-5	6	-23
	7	0		-1	4		6	-23	

- $r = r_1 - q \times r_2$, $s = s_1 - q \times s_2$, $t = t_1 - q \times t_2$

1.2.2 모듈라 연산(modular arithmetic)

- 임의의 정수 a 를 양의 정수 n 으로 나누면 몫이 q 가 되고 음이 아닌 나머지 r 을 얻는다.

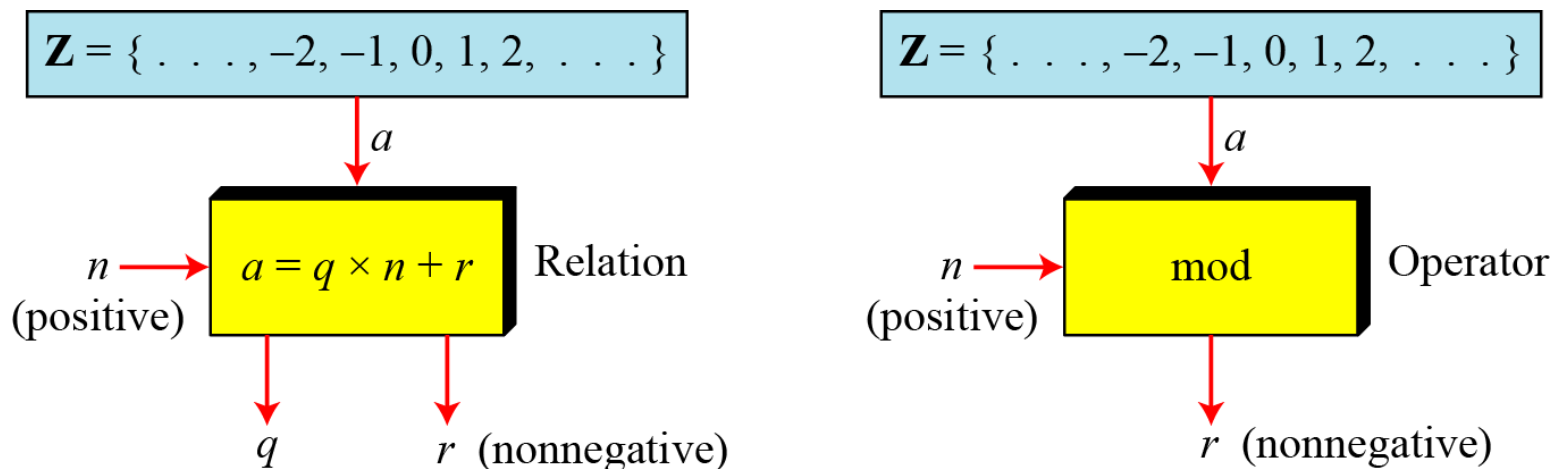
$$a = qn + r \quad 0 \leq r < n$$

$$23 = 4 \times 5 + 3; -17 = (-3) \times 5 + (-2) = (-4) \times 4 + 3$$

- mod 연산

$$a \bmod n = r$$

$$23 \bmod 5 = 3; -17 \bmod 5 = 3$$



1.2.2 모듈라 연산(modular arithmetic)

- **mod 연산**은 임의의 정수 a 를 양의 정수 n 으로 나누면 몫이 q 가되고 음이 아닌 나머지 r 을 얻는다.

$$a = qn + r \quad 0 \leq r < n$$

- mod 연산은 **완전잉여계** \mathbb{Z}_n 을 만든다.

$$\mathbb{Z}_n = \{ 0, 1, 2, 3, \dots, (n-1) \}$$

$$\mathbb{Z}_2 = \{ 0, 1 \}$$

$$\mathbb{Z}_6 = \{ 0, 1, 2, 3, 4, 5 \}$$

$$\mathbb{Z}_{11} = \{ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 \}$$

- **합동(Congruence)**

$$2 \equiv 12 \pmod{10}$$

$$13 \equiv 23 \pmod{10}$$

$$3 \equiv 8 \pmod{5}$$

$$8 \equiv 13 \pmod{5}$$

1.2.2 모듈라 연산(modular arithmetic)

■ mod 연산의 성질

First Property: $(a + b) \bmod n = [(a \bmod n) + (b \bmod n)] \bmod n$

Second Property: $(a - b) \bmod n = [(a \bmod n) - (b \bmod n)] \bmod n$

Third Property: $(a \times b) \bmod n = [(a \bmod n) \times (b \bmod n)] \bmod n$

$$10 \bmod 3 = 1 \quad \rightarrow \quad 10^n \bmod 3 = (10 \bmod 3)^n = 1$$

$$10 \bmod 9 = 1 \quad \rightarrow \quad 10^n \bmod 9 = (10 \bmod 9)^n = 1$$

$$10 \bmod 7 = 3 \quad \rightarrow \quad 10^n \bmod 7 = (10 \bmod 7)^n = 3^n \bmod 7$$

$$a = a_n \times 10^n + \dots + a_1 \times 10^1 + a_0 \times 10^0$$

$$\text{For example: } 6371 = 6 \times 10^3 + 3 \times 10^2 + 7 \times 10^1 + 1 \times 10^0$$

$$a \bmod 3 = (a_n \times 10^n + \dots + a_1 \times 10^1 + a_0 \times 10^0) \bmod 3$$

$$= (a_n \times 10^n) \bmod 3 + \dots + (a_1 \times 10^1) \bmod 3 + (a_0 \times 10^0) \bmod 3$$

$$= (a_n \bmod 3) \times (10^n \bmod 3) + \dots + (a_1 \bmod 3) \times (10^1 \bmod 3) + (a_0 \bmod 3) \times (10^0 \bmod 3)$$

$$= a_n \bmod 3 + \dots + a_1 \bmod 3 + a_0 \bmod 3$$

$$= (a_n + \dots + a_1 + a_0) \bmod 3$$

1.2.3 역원 (Inverses)

■ 덧셈상의 역원, 곱셈상의 역원

✕ \mathbb{Z}_n 상에서 덧셈상의 역원

$$a + b \equiv 0 \pmod{n}$$

✕ \mathbb{Z}_n 상에서 곱셈상의 역원

$$a \times b \equiv 1 \pmod{n}$$

✕ In modular arithmetic, an integer **may or may not** have a multiplicative inverse. Number a has the mult. Inverse iff $\gcd(n, a) \equiv 1 \pmod{n}$

1.2.3 역원 (Inverses)

- Find all multiplicative inverses in Z_{10} .

- ✗ There are only three pairs: (1, 1), (3, 7) and (9, 9). The numbers 0, 2, 4, 5, 6, and 8 do not have a multiplicative inverse.

- Find the mult. Inverse of b in Z_n

- ✗ Use the extended Euclidean algorithm

- $s \times a + t \times b = \gcd(a, b)$

- $s \times n + t \times b = 1$ (since $\gcd(n, b) \equiv 1 \pmod{n}$)

- $(s \times n + t \times b) \pmod{n} = 1 \pmod{n}$

- $(t \times b) \pmod{n} = 1 \pmod{n}$

- t is the inverse of b .

1.3 고전암호

- 2 가지 원칙: 치환(Substitution)과 전치(Transposition)
- 암호 단위 : 고전 암호- 문자; 현대 암호-비트
- 공격 유형
 - × 전사적 공격(Brute Force Attack) -전수 키 탐색 공격 (Exhaustive Key Search Attack), 현대 암호는 키의 길이가 길기 때문에 전사적 공격은 사실상 불가능
 - × 빈도수 분석(Frequency Analysis)-평문의 통계학적 특성이 암호문에 나타나는 성질을 이용하여 공격하는 방법
 - × cryptanalytic attack
 - ▶ the nature of the algorithm plus perhaps some knowledge of the general characteristics of the plaintext or even some sample plaintext-ciphertext pairs.

1.3.1 치환 암호(Substitution Cipher)

■ 치환 암호

- × 단일 문자 치환 암호(Monoalphabetic Substitution Cipher) :
평문의 한 문자와 암호문의 한 문자는 언제나 일대일 관계
- × 다중 문자 치환 암호(Polyalphabetic Substitution Cipher)
- × Plaintext and ciphertext in Z_{26}

Plaintext →	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
Ciphertext →	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
Value →	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

1.3.1 치환 암호(Substitution Cipher)

■ 단일 문자 치환 암호: 덧셈 암호(Additive Cipher)

✧ 시저 암호 (Caesar cipher)

▶ 평문의 한 문자가 오른쪽 세 자리 뒤에 위치한 문자로 치환

✧ 암호화 : $c \equiv m + 3 \pmod{26}$, $m \in \mathbb{Z}_{26}$

✧ 복호화 : $m \equiv c - 3 \pmod{26}$, $c \in \mathbb{Z}_{26}$,

m	A	B	C	D	E	F	G	H	I	J	K	L	M
	0	1	2	3	4	5	6	7	8	9	10	11	12
$c \equiv m + 3 \pmod{26}$	d	e	f	g	h	i	j	k	l	m	n	o	p
	3	4	5	6	7	8	9	10	11	12	13	14	15
m	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
	13	14	15	16	17	18	19	20	21	22	23	24	25
$c \equiv m + 3 \pmod{26}$	q	r	s	t	u	v	w	x	y	z	a	b	c
	16	17	18	19	20	21	22	23	24	25	0	1	2

표 1.2 시저 암호에서 평문과 암호문

1.3.1 치환 암호(Substitution Cipher)

■ 덧셈 암호(Additive Cipher)

$$\text{암호화} : c \equiv m + k \pmod{26}, \quad m \in \mathbb{Z}_{26}$$

$$\text{복호화} : m \equiv c - k \pmod{26}, \quad c \in \mathbb{Z}_{26}$$

■ 치환암호의 경우

✕ 가능한 키의 개수는 총 $26 \times 25 \times \cdots \times 1 = 26!$ 개

▶ 전사적 공격을 이용하여 공격자가 키를 찾는 것은 불가능

1.3.1 치환 암호(Substitution Cipher)

■ 빈도수 공격 : 통계적 특성 이용

✕ 988,968 개의 영어 단어 중, "E"가 사용되는 횟수는 12.7%

Letter	Probability	Letter	Probability	Letter	Probability
A	0.082	B	0.015	C	0.028
D	0.043	E	0.127	F	0.022
G	0.020	H	0.061	I	0.070
J	0.002	K	0.008	L	0.040
M	0.024	N	0.067	O	0.075
P	0.019	Q	0.001	R	0.060
S	0.063	T	0.091	U	0.028
V	0.010	W	0.023	X	0.001
Y	0.020	Z	0.001		

1.3.1 치환 암호(Substitution Cipher)

■ 다중 문자 치환 암호: 비제네르 암호 (Vigenère Cipher)

✧ 길이가 l인 키워드를 암호화 키로 사용

✧ $K = k_1k_2k_3\dots k_d$, $f_i(m) = (m + k_i) \bmod n$

▶ 예 : $K = \text{"CIPHER"} \ (l = 6)$

▶ $\text{"THISISASECRETMESSAGE"}$

평문	T	H	I	S	I	S	A	S	E	C	R	E	T	M	E	S	S	A	G	E
	19	7	8	18	8	18	0	18	4	2	17	4	19	12	4	18	18	0	6	4
키	C	I	P	H	E	R	C	I	P	H	E	R	C	I	P	H	E	R	C	I
	2	8	15	7	4	17	2	8	15	7	4	17	2	8	15	7	4	17	2	8
암호문	V	P	X	Z	M	J	C	A	T	J	V	V	V	U	T	Z	W	R	I	M
	21	15	23	25	12	9	2	0	19	9	21	21	21	20	19	25	22	17	8	12

주기 $l = 6$

1.3.1 치환 암호(Substitution Cipher)

■ 다중 문자 치환 암호: 비제네르 암호 (Vigenère Cipher)

✧ 공격 : KASISKI METHOD

▶ period " d "를 결정하는 방법


LIOMWGFEGGDVWGHHCQUCRHRWAGWIOUQLKGZETKKMEVLWPCZVGTH-
VTSGXQOVGCSVETQLTJSUMVWVEUVLXEWSLGFZMVVWLGYHCUSWXQH-
KVGSHEEVFLCFDGVSUMPHKIRZDMPHHBVVWJWIXGFWLTSHGJOUEEHH-
VUCFVGOWICQLTJSUKGLW

String	First Index	Second Index	Difference
JSU	68	168	100
SUM	69	117	48
VWV	72	132	60
MPH	119	127	8

1.3.1 치환 암호(Substitution Cipher)

- 다중 문자 치환 암호: 비제네르 암호 (Vigenère Cipher)
 - × 차이의 GCD는 4 \rightarrow 키 길이는 4의 배수
 - × First try $l = 4$. (C1, C2, C3, C4에 빈도수 분석)

```
C1: LWGWCRAOKTEPGTQCTJVUEGVGUQGECVPRPVJGTJEUGCJG
P1: jueuapymircneroarhtsthihytrahcieixsthcarrehe
C2: IGGGQHGWGKVCTSSOSQSWVWFVYSHSVFSHZHWWFSSOHCOQSL
P2: ussctsisiswhofeaeceihcetesoeatnpntherhctecex
C3: OFDHURWQZKLZHGVVLUVLSZWHWKHFDUKDHVIWHUHFWLUW
P3: lcaerotnwhiwedssirsiirhketehretltiideatrairt
C4: MEVHCWILEMWVVXGETMEXLMLCXVELGMIMBWXLGEVVITX
P4: lardysehaisrrtcapiafpwtethecarhaesfterectpt
```



Julius Caesar used a cryptosystem in his wars, which is now referred to as Caesar cipher.
It is an additive cipher with the key set to three. Each character in the plaintext is shifted three characters to create ciphertext.

1.3.2 전치 암호(Transposition Cipher)

■ 평문 메시지의 문자들을 재배열

- × 전치 암호의 암호화 함수를 π 라 하고 문자열의 길이를 l 이라 하면,

$$\pi = (\pi(1), \dots, \pi(l))$$

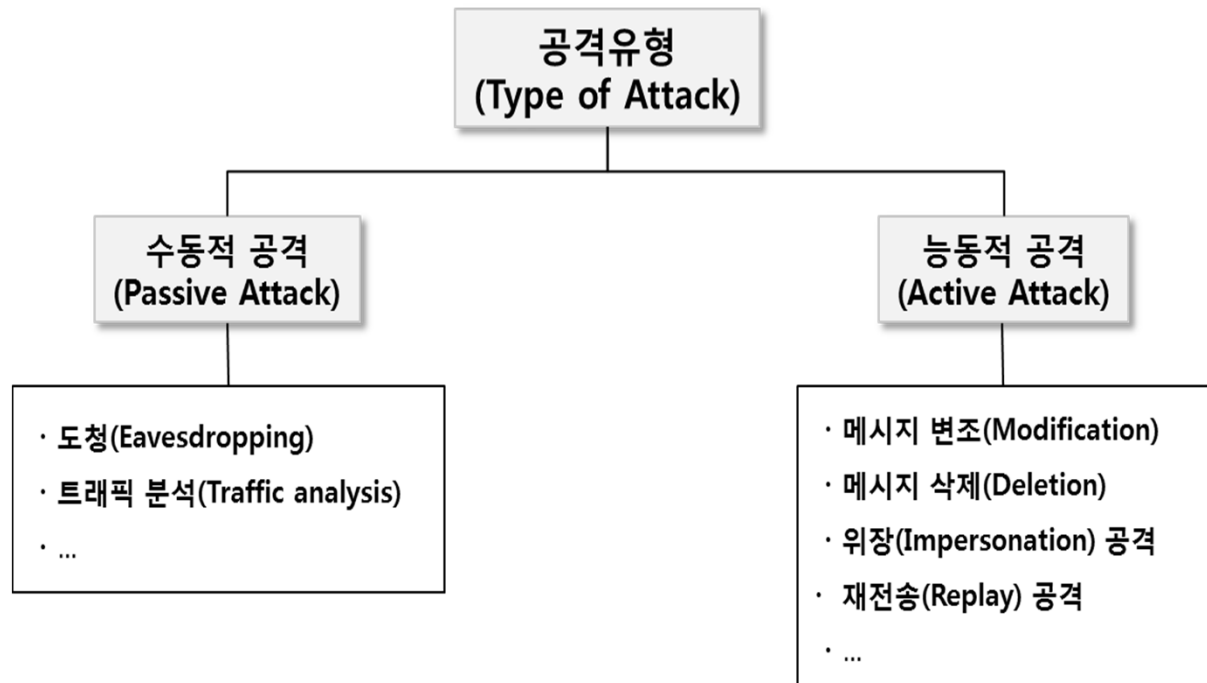
- × $\pi(i)$ 는 평에서 i 번째 위치에 있는 문자의 암호문에서의 위치
- × 예제 : $\pi = (\pi(1), \dots, \pi(5)) = (3, 1, 4, 5, 2)$

평문	T	H	I	S	I	S	A	S	E	C	R	E	T	M	E	S	S	A	G	E
암호문	H	I	T	I	S	A	C	S	S	E	E	E	R	T	M	S	E	S	A	G

- ▶ 메시지의 길이가 30인 경우 $\rightarrow 1! + 2! + 3! + \dots + 30!$
- ▶ 블록이 30의 약수! $\rightarrow 30 = 1 \times 2 \times 3 \times 5 \rightarrow 30$ 의 인수인 1, 2, 3, 5, 6, 10, 15, 30를 이용하여 $1! + 2! + 3! + 5! + 6! + 10! + 15! + 30!$

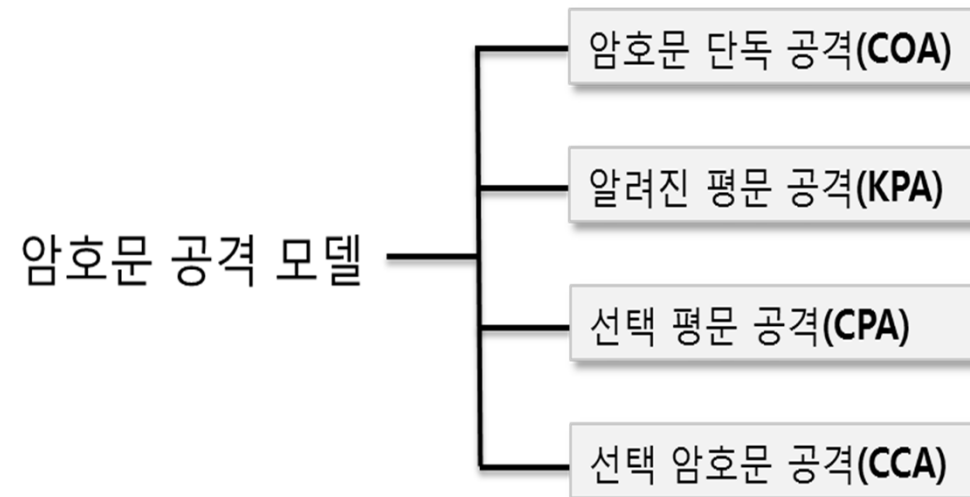
1.4 암호시스템의 안전성

■ 공격유형(Type of Attack)



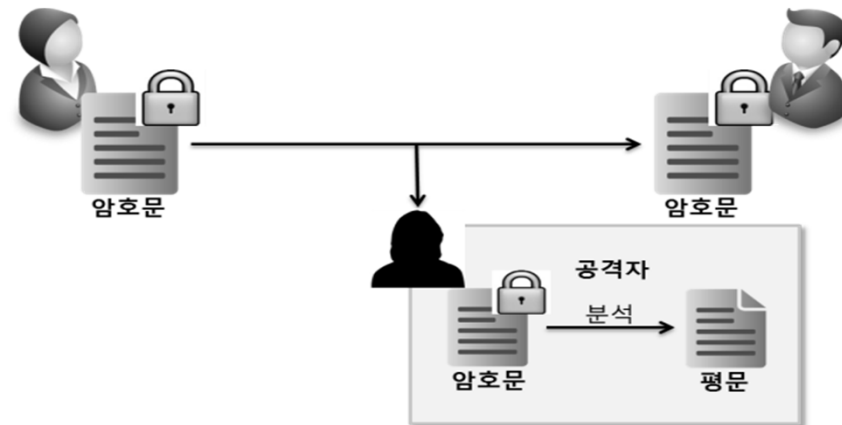
1.4 암호시스템의 안전성

■ 공격모델(Attack Model)

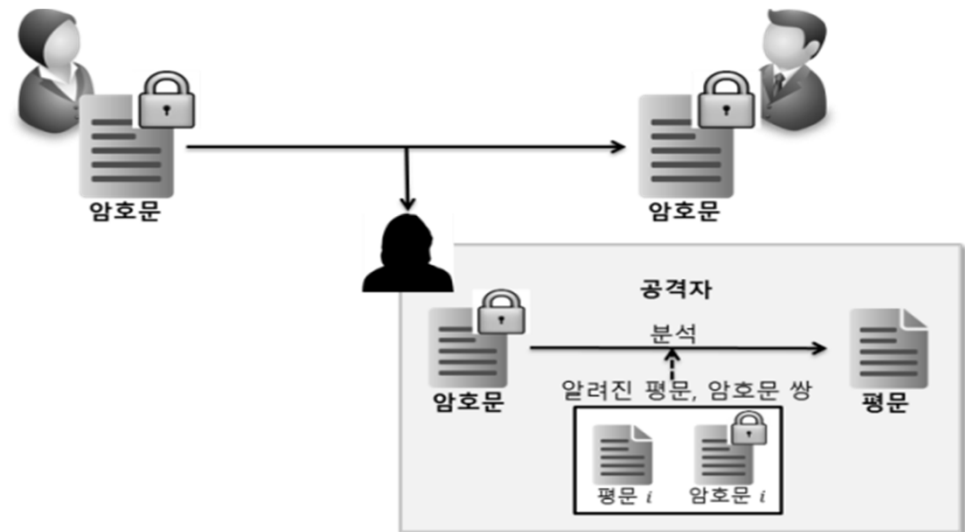


1.4 암호시스템의 안전성

- ✕ 암호문 단독 공격 (Ciphertext Only Attack, COA)

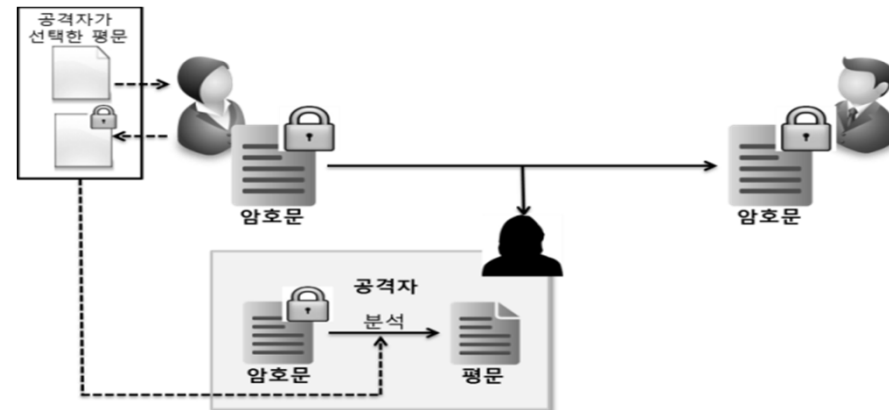


- ✕ 알려진 평문 공격 (Known Plaintext Attack, KPA)

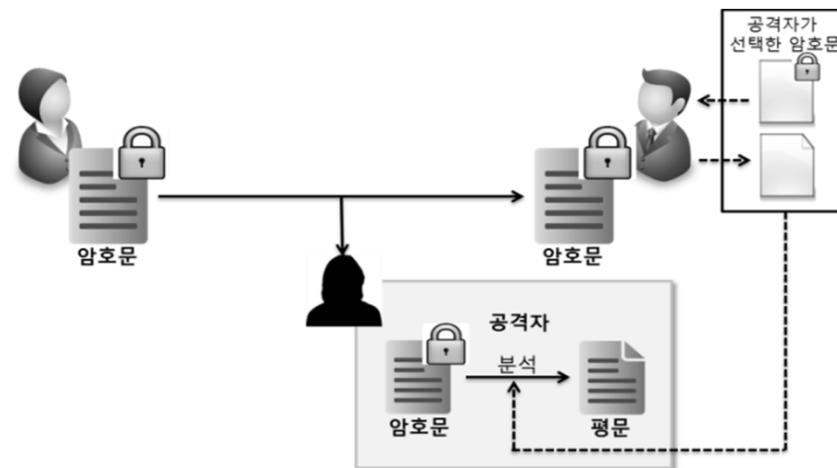


1.4 암호시스템의 안전성

- ✕ 선택 평문 공격
(Chosen Plaintext Attack, CPA)



- ✕ 선택 암호문 공격
(Chosen Ciphertext Attack, CCA)



1.4 암호시스템의 안전성

■ 암호의 안전성 개념

- ✕ Secure Encryption ?

- ✕ Given C ,

1. No adversary can find k ?
2. No adversary can find P ?
3. No adversary can find any character?
4. No adversary can find any meaningful information?

Meaningful? → definitions of security should suffice for all potential applications!

5. No adversary can compute any function of P from C .

1.5 정보보호 서비스

■ 3대 정보보호 서비스(NIST)

- × 기밀성(Confidentiality)
- × 무결성(Integrity)
- × 가용성(Availability)

■ 위의 정보보호 서비스 이외에 환경에 따라 요구되는 서비스는 다양하며 다음과 같다.

- × 인증(Authentication)
 - ▶ 개체 인증(Entity Authentication) : 개체가 정당한(혹은 개체가 주장하는) 개체인지를 확인하는 성질을 의미한다.
 - ▶ 메시지 인증(Message Authentication) : 수신된 메시지가 정당한 송신자로부터 전송된 것인지를 확인하는 성질을 의미한다. 즉 수신된 메시지의 송신자를 인증하는 과정이다.
- × 부인방지(Non-Repudiation)
- × 접근제어(Access Control)