2장 암호의 역사

박 종 혁

Tel: 970-6702
Email: jhpark1@snut.ac.kr

2.0 주요 내용

- 암호
  - 시저 암호
  - 단일 치환 암호
  - 애니그마
- 암호 해독법
  - 전사 공격(brute force attack)
  - 빈도 분석
- 알고리즘과 키의 관계

2.1 시저 암호

- 단순한 암호인 시저 암호를 소개

2.1.1 시저 암호란

- **시저 암호**(Caesar cipher)는 줄리어스 시저가 사용했다고 하는 암호이다.
  - 시저는 기원전 100년경에 로마에서 활약했던 장군이었다
- 시저 암호에서는 평문에서 사용되고 있는 알파벳을 일정한 문자 수 만큼 「평행이동」시킴으로써 암호화를 행한다

알파벳을 3문자씩 평행 이동시키기

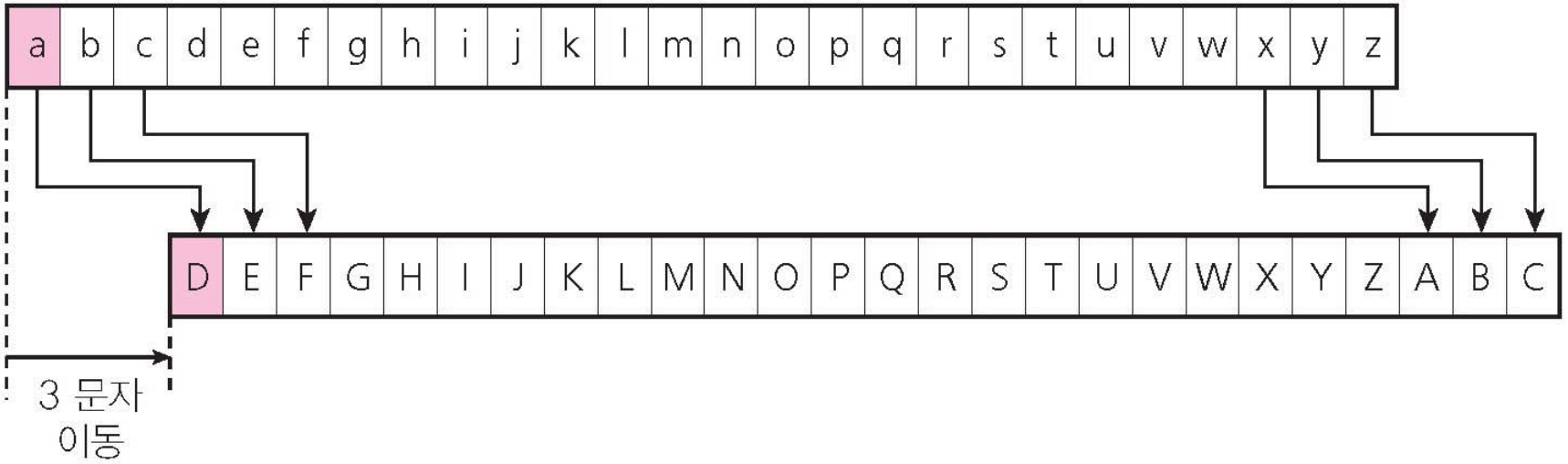


그림 2-1 시저 암호에서는 알파벳을 「평행 이동 시킨다.」

2.1.2 시저 암호의 암호화

- 예: 평문=kabsoonyee

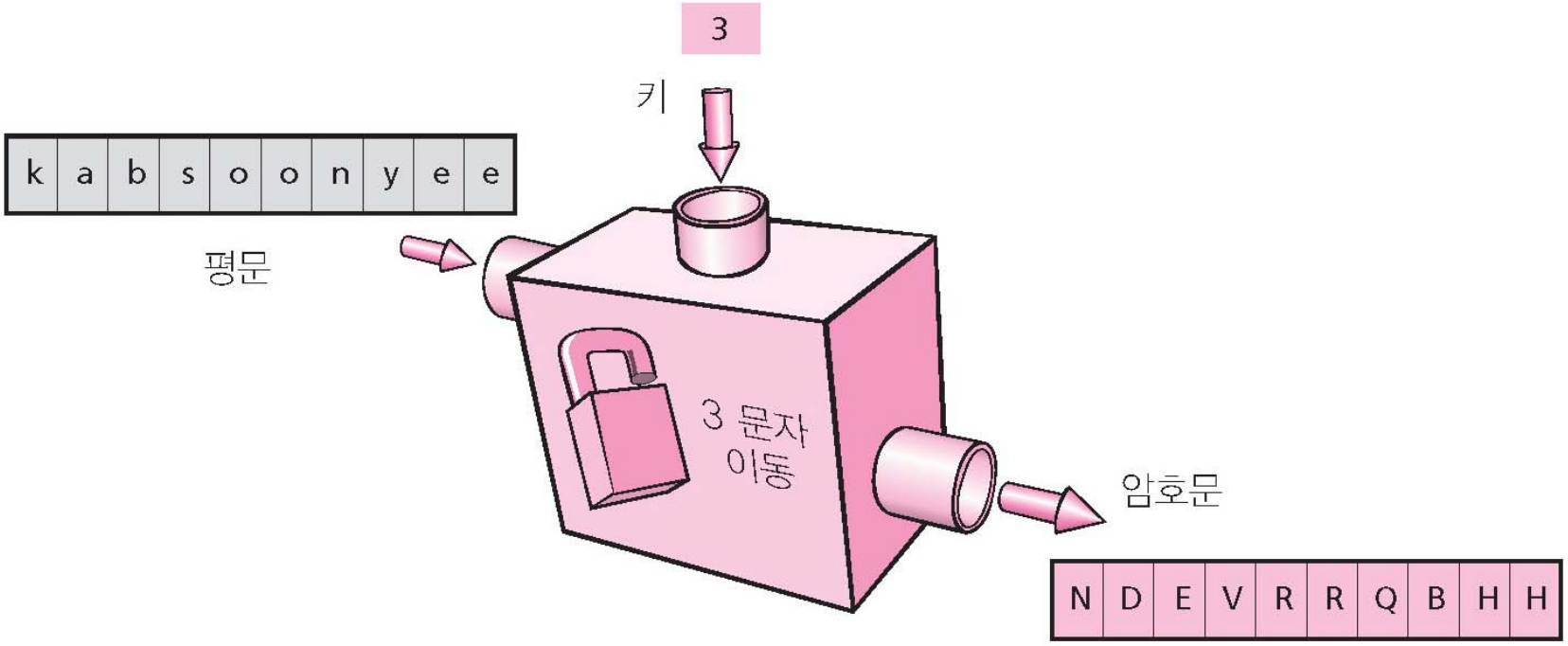
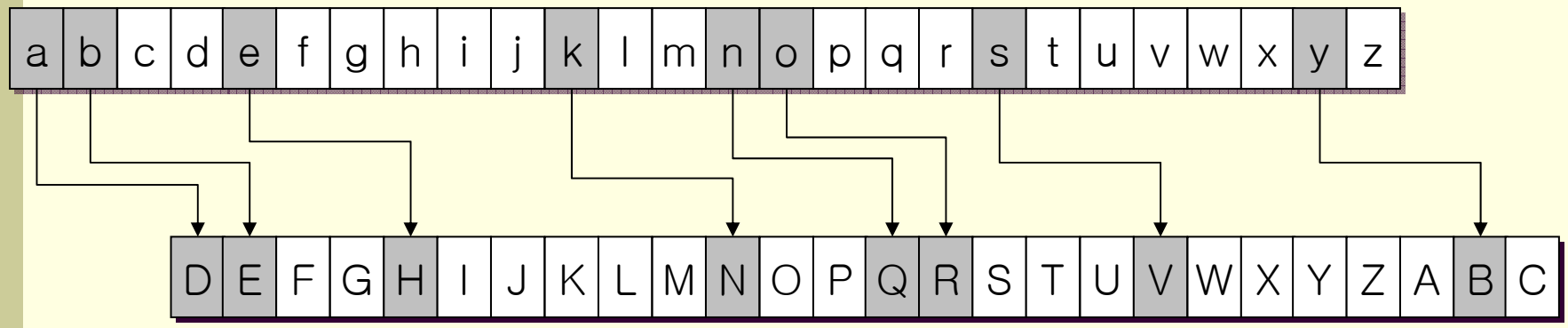
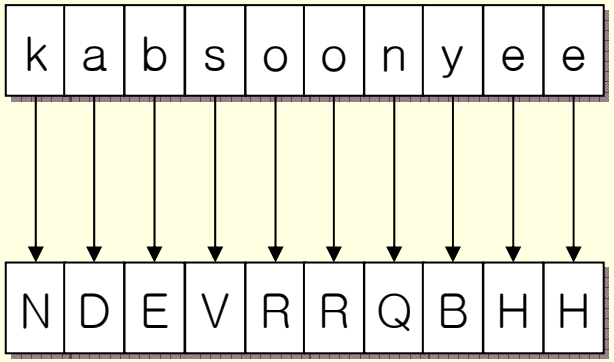


그림 2-2 시저 암호에 의한 암호화(키는 3)

시저 암호 암호화


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
D E F G H I J K L M N O P Q R S T U V W X Y Z A B C


k a b s o o n y e e
N D E V R R Q B H H

시저 암호

- 시저 암호에서는 「알파벳 문자를 평행 이동시킨다」는 조작이 「암호화 알고리즘」에 해당
- 평행 이동시키는 문자수가 「키」에 해당한다.

2.1.3 시저 암호의 복호화

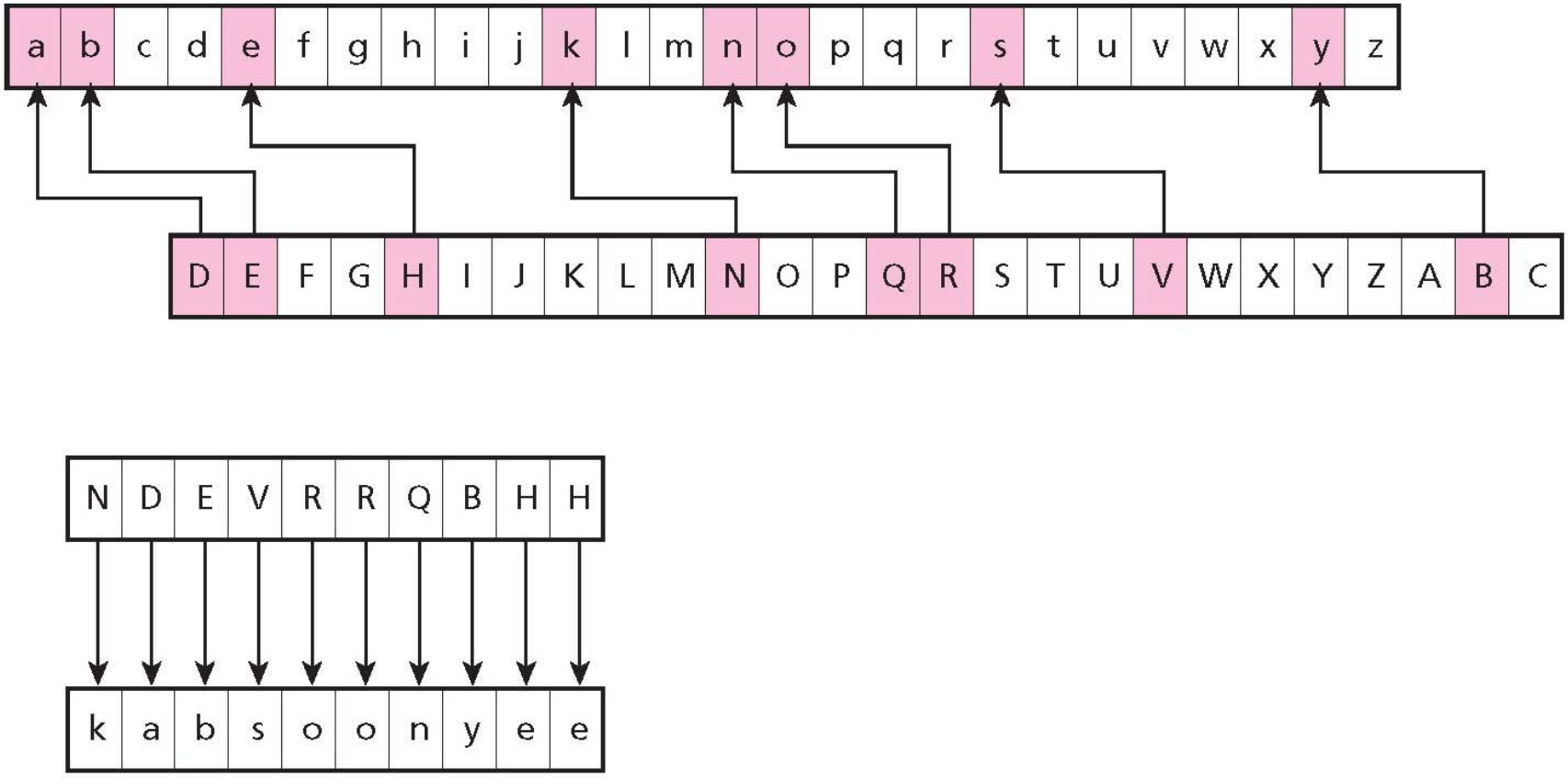


그림 2-3 시저 암호에 의한 복호화는 역방향 평행이동이다(키는 −3)

시저 암호에 의한 복호화(키는 -3)

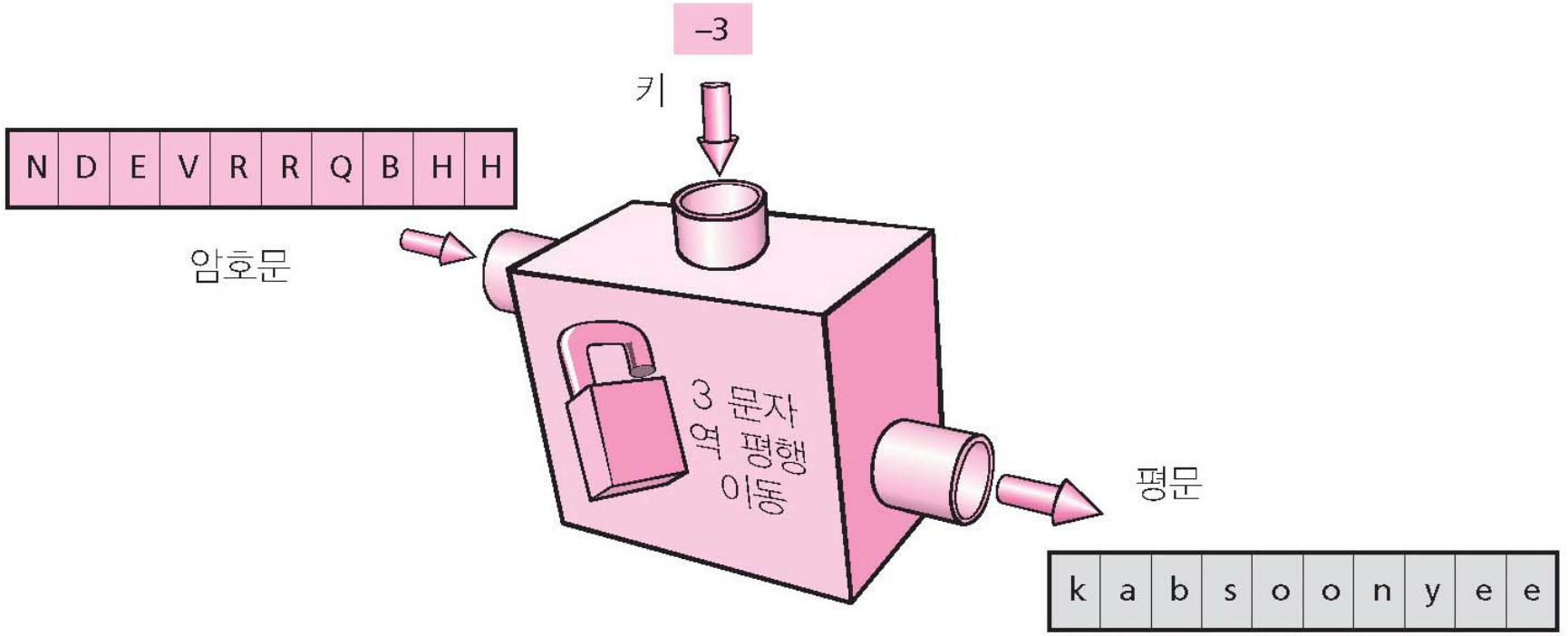


그림 2-4 시저 암호에 의한 복호화(키는 -3)

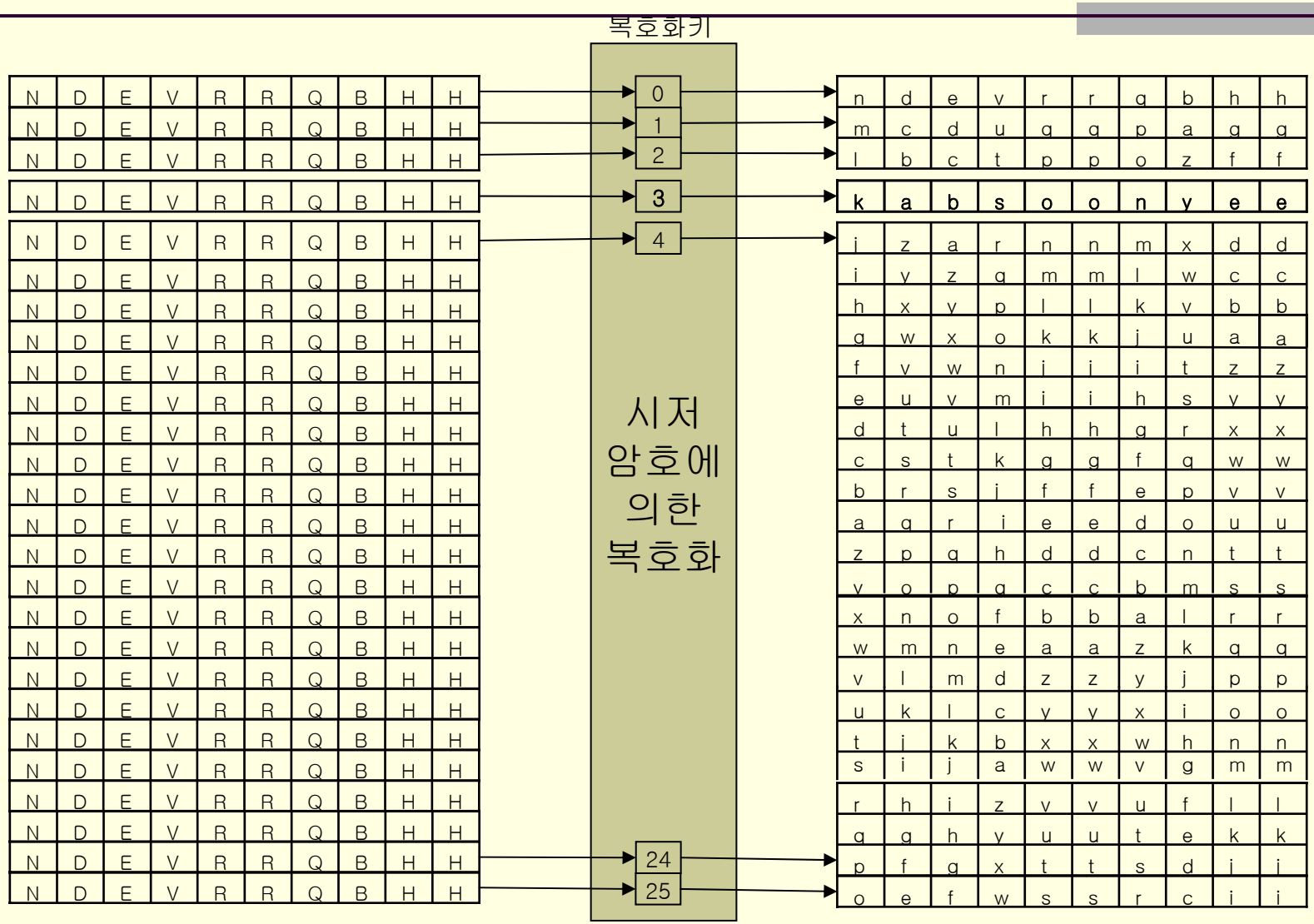
2.1.4 전사공격에 의한 해독

- 수신자 이외의 사람(3이라는 키를 모르는 사람)이 암호문 NDEVRRQBHH 을 보고 다른 정보 없이도 kabsoonyee 라는 메시지를 맞출 수는 없을까?
- 다시 말해, 시저 암호를 해독할 수 없을까?

시저 암호의 방법

- 시저 암호에서는 알파벳을 평행 이동시키는 문자 수가 키가 된다. 알파벳은 26 문자이므로 암호화 키는 0에서 25까지 26가지밖에 없다
- 그럼 이 26 가지 키를 순서대로 사용해서 복호화를 해보자

시저 암호문에 대한 전사공격



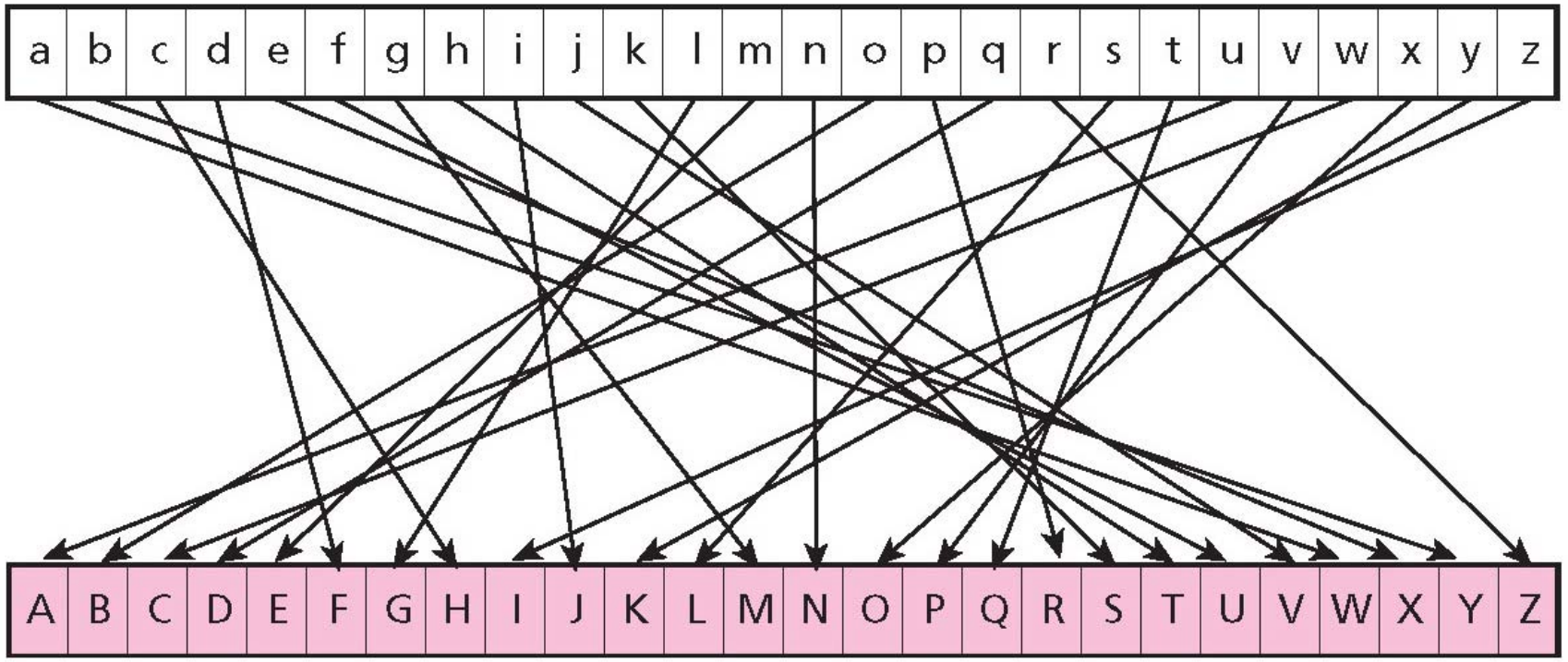
2.2 단일 치환 암호

- 알파벳 26문자를 무작위 순서로 나열하여 얻게 되는 집합을 생각해보자.
- 이 무작위로 만든 집합과 원래 순서대로 된 알파벳 집합 {a,b,c,...,z}은 일대일 대응관계가 되며 이 대응관계를 이용하면 하나의 암호를 만들 수 있게 된다.

2.2.1 단일 치환 암호

- **단일 치환 암호**(simple substitution cipher) :
  알파벳의 대응관계를 이용하여 평문을 구성하는 알파벳을 다른 알파벳으로 변환하는 암호

단일 치환 암호의 치환 표(예)



대응관계를 보기 쉽게 한 치환표 (내용은 같음)

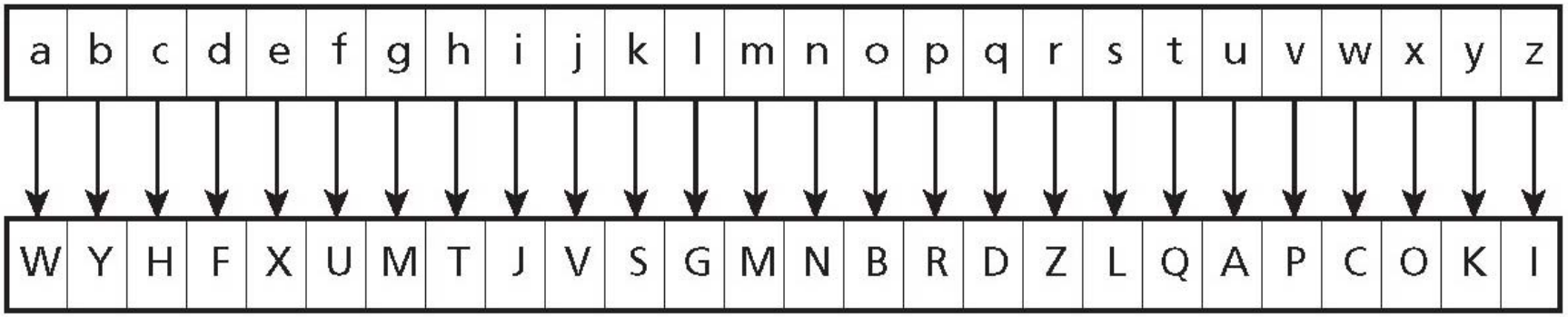


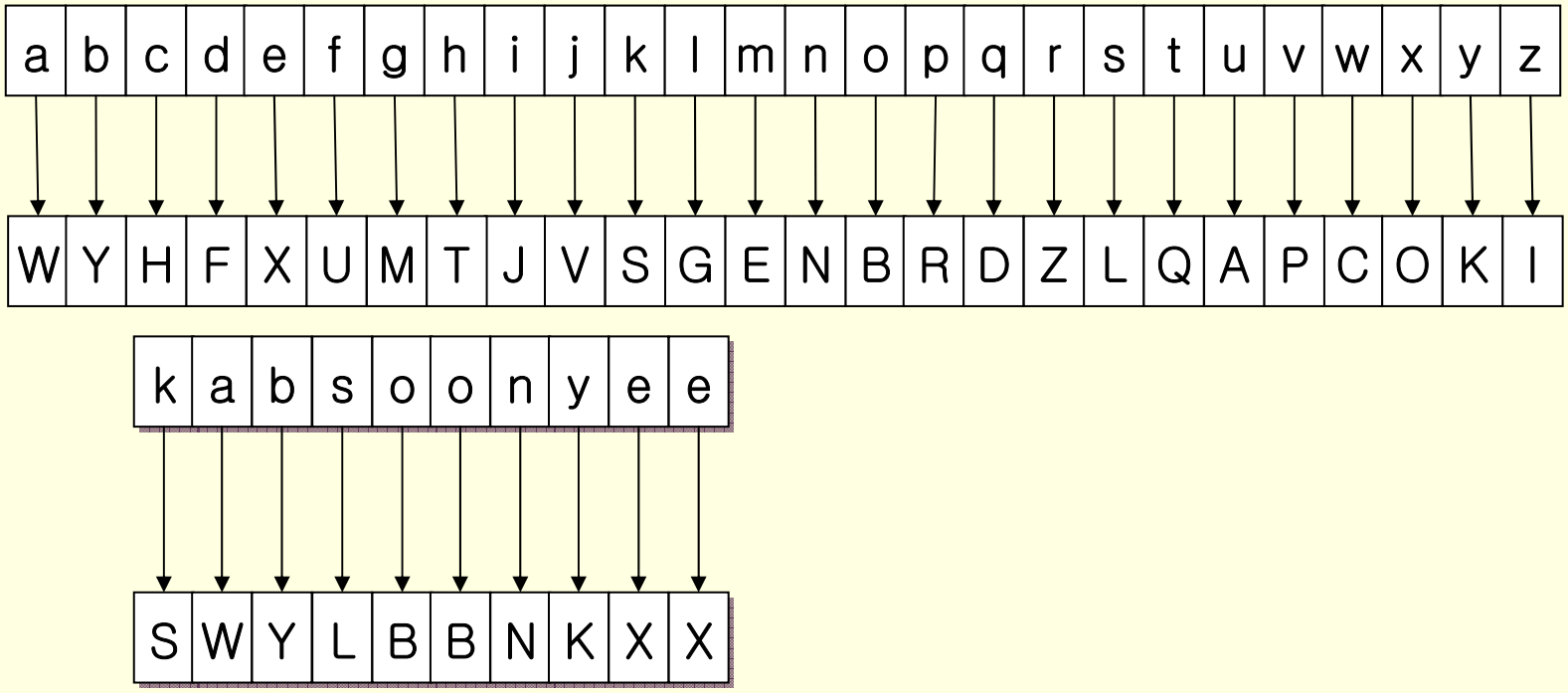
그림 2-6 단일 치환 암호의 치환 표(예)

2.2.2 단일 치환 암호의 암호화

- 단일 치환 암호의 암호화는 평문을 구성하는 알파벳을 한 문자씩 치환표를 참조하여 변환해 가는 작업의 반복이다

단일 치환 암호의 암호화(예)

- 평문: kabsoonyee



- 암호문: SWYLBBNKXX

2.2.3 단일 치환 암호의 복호화

- 암호화 때에 사용한 치환표를 써서 암호화의 역변환을 행하면 단일 치환 암호에 의한 복호화를 할 수 있다.
- 단일 치환 암호를 이용하여 암호화된 암호문을 복호화 하기 위해서는 암호화 때에 사용한 치환표가 필요하므로 송신자와 수신자는 치환표를 공유하고 있어야 한다.
- 이 치환표가 단일 치환 암호의 「키」가 된다.

2.2.4 단일 치환 암호의 키 공간

- kabsoonyee는 시저 암호(키는 3)로 암호화하면 NDEVRRQBHH이 된다.
- 한편, 단일 치환 암호(키는 그림 2-6 참조)로 암호화하면 SWYLBBNKXX가 된다.
- NDEVRRQBHH와 SWYLBBNKXX은 모두 의미를 알 수 없다는 점에서는 비슷한 문자열이다.
- 이 암호문만을 보아서는 시저 암호와 단일 치환 암호 중 어느 쪽이 해독하기 어려운가를 판단할 수는 없다.

단일치환 암호와 시저 암호

- 시저 암호는 전사공격으로 해독할 수 있다. 그러나 **단일 치환 암호는 전사공격으로 해독하는 것이 어렵다**
- 단일 치환 암호가 시저 암호에 비해 훨씬 많은 키의 후보를 가질 수 있다

키 공간(key space)

- 키 공간(key space)
  - 어떤 암호로 사용할 수 있는 「모든 키의 집합」
- 키 공간의 크기:
  - 이 키 공간에 속하는 가능한 키의 총수
  - 키 공간이 크면 클수록 전사공격은 어려워지게 된다.

단일 치환 암호의 키의 총수

- 단일 치환 암호의 키의 총수는

$$26 \times 25 \times 24 \times 23 \times \ldots \times 1 =$$

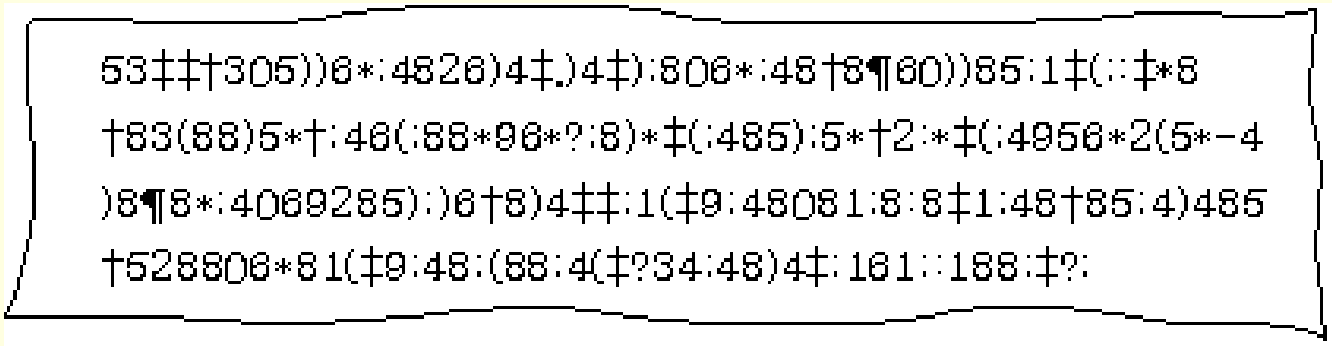
$$26! = 403291461126605635584000000$$

가 된다.

전사공격의 어려움

- 이 정도로 키가 많으면 전사공격으로 조사하는 것은 매우 어렵다.
- 왜냐 하면, 아무리 빨리 키를 적용해 본다고 해도 그 적용시간이 있기 때문에 모든 키를 적용해보는 데에는 상당한 시간이 필요하다.
- 예를 들면 1초에 10억 개의 키를 적용하는 속도로 조사한다고 해도, 모든 키를 조사하는 데 120억년 이상의 시간이 걸리기 때문이다.
- 바른 키를 찾기까지의 평균 시간은 약 60억년

2.2.5 빈도 분석에 의한 해독



A good glass in the bishop's hostel in the devil's seat forty-one degrees and thirteen minutes northeast and by north main branch seventh limb east side shoot from the left eye of the death's-head a bee line from the tree through the shot fifty feet out. - 애드가 앨런 포우『황금벌레』

소설에 등장한 빈도분석

- 빈도분석 방법은 소설에 등장했는데
  - 애드가 앨렌 포우의 '황금벌레'라는 소설과
  - 아서 코난 도일의 '셜록홈즈 이야기'인 '춤추는 남자의 모험'
- 이 장의 맨 앞에 나와 있는 내용이 바로 알렌 포우의 소설에 등장했던 암호문과 해독된 내용이다.

소설 속의 암호

그림 2-7 '셜록홈즈 이야기'인 '춤추는 남자의 모험'에 나오는 암호

최초의 빈도분석에 대한 자료

- 최초의 기록으로 남아 있는 빈도분석에 대한 내용은
- 9세기'암호문 해독에 관한 논고'에 등장하는 아랍의 현학자 알킨디(al-Kindi)에 의해 제안된 것이다

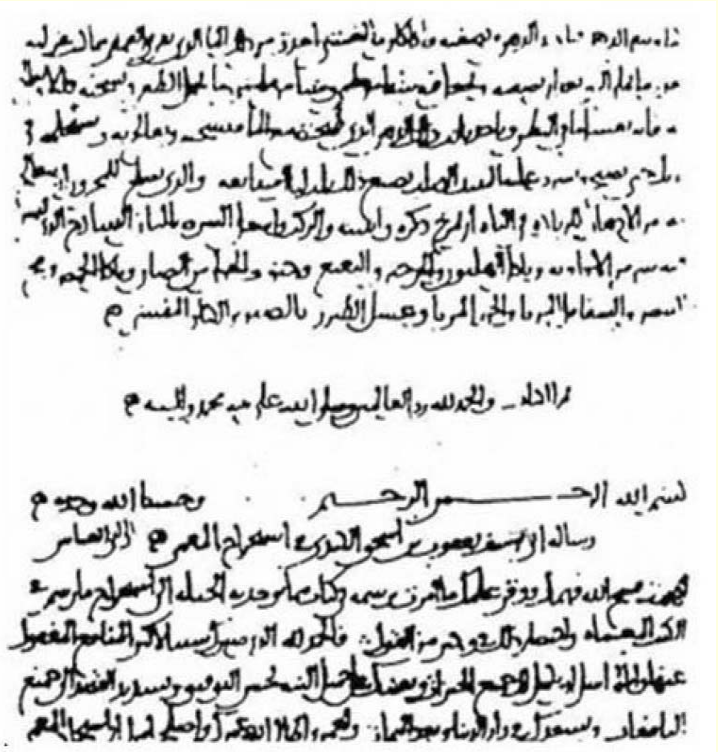


그림 2-8 알킨디의 '암호문 해독에 관한 논고' 첫 페이지

빈도분석

- 전사공격으로 단일 치환 암호를 해독하는 것은 어렵지만
- **빈도분석이라는 암호 해독법을 사용하면 단일 치환 암호도 해독할 수가 있다.**
- 빈도분석에서는 평문에 등장하는 문자의 빈도와 암호문에 나오는 문자의 빈도가 일치하는 것을 이용하는 것이다.

빈도분석의 예

- 암호문

MEYLGVIWAMEYOPINYZGWYEGMZRUUYPZAIXILGVSIZZMP
GKKDWOMEPGROEIWGPCEIPAMDKKEYCIUYMGIFRWCEGL
OPINYZHRZMPDNYWDWOGWITDWYSEDCEEIAFYYWMPIDW
YAGTYPIKGLMXFPIWCEHRZMMEYMEDWOMGQRYWCEUXM
EDPZMQRGMEEYAPISDWOFICJILYSNICYZEYMGGJIPRWIWA
IHRUNIWAHRZMUDZZYAMEYFRWCEMRPWDWOPGRWAIOID
WSDMEIGWYMSGMEPYYEYHRUNYARNFRMSDMEWGOPYIM
YPZRCCYZZIOIDWIWAIOIDWEYMPDYAILMYPMEYMWUNMD
WOUGPZYKFRMIMKIZMEIAMGODTYDMRNIWASIKJYAISIXSD
MEEDZWGZYDWMEYIDPZIXDWODIUZRPYMEYXIPYZGRPDM
DZYIZXMGAYZNDZYSEIMXGRCIWWGMOYM

빈도분석의 예

빈도조사 결과

표 2-1 암호문 속의 문자 빈도표

문자	개수	문자	개수	문자	개수	문자	개수	문자	개수
I	47	G	27	C	12	F	7	V	2
Y	47	Z	27	S	11	L	6	B	0
M	45	P	26	N	10	H	5		
W	35	R	22	U	10	J	3		
E	33	A	17	K	8	T	3		
D	30	O	16	X	8	Q	2		

문자 출현 빈도

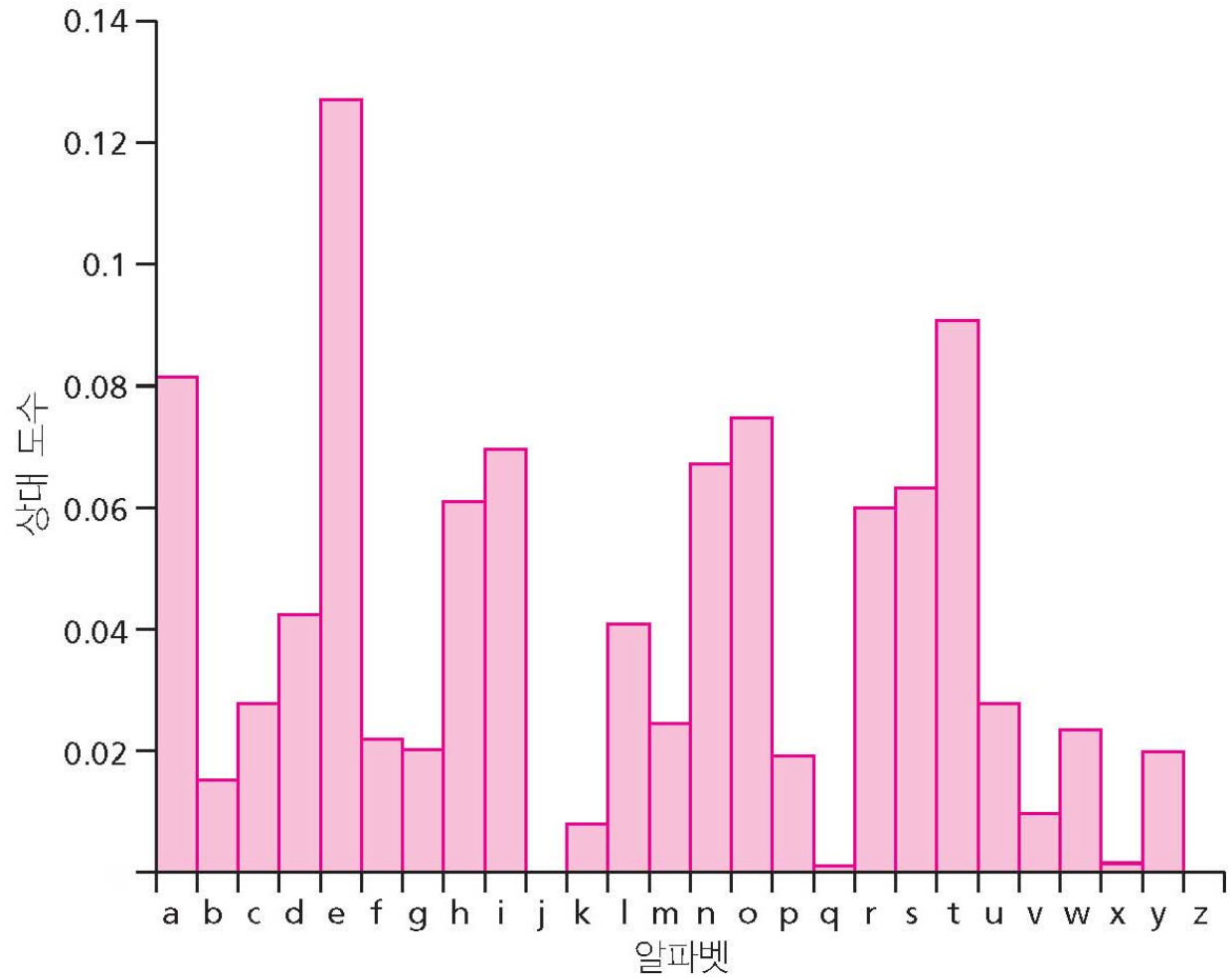


그림 2-9 문자 출현 빈도

최빈도를 갖는 문자를 e로 변환

- 가장 빈도가 높은 I와 Y 중 하나를 e라고 가정한다. Y → e 라고 가정

MEeLGVIWAMEeOPINeZGWeEGMZRUUePZAIXILGVSIZZMPGKKDW
OMEPGROEIWGPCEIPAMDKKEeCIUeMGIFRWCEGLOPINeZHRZMPD
NeWDWOGWITDWeSEDCEEIAFeeWMPIDWeAGTePIKGLMXFPIWCE
HRZMMEeMEDWOMGQReWCEUXMEDPZMQRGMEEeAPISDWOFICJI
LeSNICeZEeMGGJIPRWIWAIHRUNIWAHRZMUDZZeAMEeFRWCEMR
PWDWOPGRWAIOIDWSDMEIGWeMSGMEPeeEeHRUNeARNFRMSD
MEWGOPeIMePZRCCeZZIOIDWIWAIOIDWEeMPDeAILMePMEeMWU
NMDWOUGPZeKFRMIMKIZMEIAMGODTeDMRNIWASIKJeAISIXSDME
EDZWGZeDWMEeIDPZIXDWODIUZRPeMEeXIPeZGRPDMDZeIZXMG
AeZNDZeSEIMXGRCIWWGMOeM

영어의'the' 점검

- 영어에서 가장 자주 등장하는 단어는 "the"이다. 그러면, e로 끝나는 3문자의 패턴을 찾아보자.
- 그랬더니, MEe라는 3문자의 조합이 자주 나온다는 것을 알게 되었다.
- 게다가 MEe는 암호문의 맨 처음에도 나타나 있다.
- MEe가 the일 가능성이 높은 것 같다.
- 그러므로 M→t, E→h라고 가정해 보자.

점검 이후 익숙한 단어 추측

theLGVIWAtheOPINeZGWehGtZRUUePZAIXILGVSIZZt
PGKKDWOthPGROhIWGPChIPAtDKKheCIUetGIFRWC
hGLOPINeZHRZtPDNeWDWOGWITDWeShDChhIAFee
WtPIDWeAGTePIKGLtXFPIWChHRZtthethDWOtGQRe
WChUXthDPZtQRGthheAPISDWOFICJILeSNICeZhetG
GJIPRWIWAIHRUNIWAHRZtUDZZeAtheFRWChtRPWD
WOPGRWAIOIDWSDthIGWetSGthPeeheHRUNeARNF
RtSDthWGOPeltePZRCCeZZIOIDWIWAIOIDWhetPDe
AILtePthetWUNtDWOUGPZeKFRtItKIZthIAtGODTeDtR
NIWASIKJeAISIXSDthhDZWGZeDWtheIDPZIXDWODI
UZRPetheXIPeZGRPDtDZeIZXtGAeZNDZeShItXGRCI
WWGtOet

아는 영어 단어 총동원

- 자신의 영어단어 지식을 총동원해서 위에서 있을 법한 패턴을 찾아본다.
- 중간쯤에 있는 thPee가 눈에 띈다. 이것은 three가 아닐까(P→r)?
- 이 문자열을 보고 있으면 여기저기에 he, re, re, ter와 같은 철자가 보이므로 단편적인 정보로 P→r은 바른 대응이라는 것을 알 수 있다
- 다음에 암호문의 마지막으로 눈을 돌려 보자.
- 제일 마지막의 Oet는 bet, get, set, ... 중의 하나일 것이다.
- 자주 사용되는 단어인 get라고 가정해 본다(O→g).

단어 패턴

- 그 다음 발견한 패턴들과 가정한 대응 알파벳을 써 나가 보자.
- thethDWg라는 패턴이 보인다. 이것은 the thing일지도 모른다(D→i, W→n).
- grINe라는 패턴이 보인다.
- 사전을 찾아보았더니, grace, grade, grape, grate, grave, gripe, grofe, ...처럼 많은 후보가 있다.
- 이것으로는 결정을 할 수 없다.
- I→a를 가정해 보면 greater라는 패턴이 나오므로 I→a는 맞는 것 같다.
- 하지만, N→c를 가정하면 tricening라는 패턴이 나왔다.
- 이런 단어는 영어 단어에 없는 것 같다.
- 따라서 N→c는 잘못일지도 모른다.

빈도 추측

- 영어에서 빈도가 높은 문자 중 아직 가정에 등장하지 않은 문자는 o 이다.
- 한편 암호문 중에 등장하는 빈도가 높은 문자로서 아직 모르는 것은 G와 Z이다.
- 여기서 G→o를 가정해 보자.
- 여기까지의 가정을 써서 암호문을 다시 읽는다.

지금까지 정리하면

theLoVanAthegraNeZonehotZRUUerZAaXaLoVSaZZtr
oKKingthroRghanorCharAtiKKheCaUetoaFRnChoLgra
NeZHRZtriNeningonaTineShiChhaAFeentraineAoTera
KoLtXFranChHRZtthethingtoQRenChUXthirZtQRothhe
AraSingFaCJaLeSNaCeZhetooJarRnanAaHRUNanAH
RZtUiZZeAtheFRnChtRrningroRnAagainSithaonetSoth
reeheHRUNeARNFRtSithnogreaterZRCCeZZagainanA
againhetrieAaLterthetnUNtingUorZeKFRtatKaZthaAto
giTeitRNanASaKJeAaSaXSithhiZnoZeintheairZaXingia
UZRretheXareZoRritiZeaZXtoAeZNiZeShatXoRCannot
get

패턴을 더 들여다 본다

- 끝에 Cannotget이라는 패턴이 등장했다.
- C→c가 틀림없다.
- C→c라는 것을 통해 조금 전에 생각한 N→c는 역시 잘못이라는 것을 알 수 있다.

지금까지 내용을 정리해보자

theLoVanAthegraNeZonehotZRUUerZAaXaLoVSaZZtr
oKKingthroRghanorcharAtiKKhecaUetoaFRnchoLgra
NeZHRZtriNeningonaTineShichhaAFeentraineAoTera
KoLtXFranchHRZtthethingtoQRenchUXthirZtQRothhe
AraSingFacJaLeSNaceZhetooJarRnanAaHRUNanAHR
ZtUiZZeAtheFRnchtRrningroRnAagainSithaonetSothre
eheHRUNeARNFRtSithnogreaterZRcceZZagainanAag
ainhetrieAaLterthetnUNtingUorZeKFRtatKaZthaAtogiT
eitRNanASaKJeAaSaXSithhiZnoZeintheairZaXingiaUZ
RretheXareZoRritiZeaZXtoAeZNiZeShatXoRcannotget

Which일 것이다
그래서 S→w

빈도가 낮은 문자 추측

- 빈도가 높은 문자뿐만 아니라 암호문 중에서 빈도가 낮은 문자인 Q를 포함하는 패턴을 찾아보자.
- thethingtoQRench라는 패턴이 찾아졌다.
- 이것은 분명히 the thing to QRench이다. 사전을 찾아보니 quench라는 단어가 있었다(Q→q, R→u). quench라는 것은 「갈증을 해소하다」라는 의미이다. 마시는 것에 관한 이야기가 아닐까?
- hotZuUUer라는 패턴이 찾아졌다. 이것은 hot summer일 것이다(Z→s, U→m). U가 두 개 연속해 있다는 것이 큰 실마리였다. 「갈증을 해소하다」라는 문맥과도 일치한다.

다시 정리해보자

theLoVanAthegraNesonehotsummersAaXaLoVwa
sstroKKingthroughanorcharAtiKKhecametoaFunc
hoLgraNesHustriNeningonaTinewhichhaAFeentra
ineAoTeraKoLtXFranchHustthethingtoquenchmXt
hirstquothheArawingFacJaLewNaceshetooJaruna
nAaHumNanAHustmisseAtheFunchturningrounAa
gainwithaonetwothreeheHumNeAuNFutwithnogre
atersuccessagainanAagainhetrieAaLterthetnmNti
ngmorseKFutatKasthaAtogiTeituNanAwaKJeAaw
aXwithhisnoseintheairsaXingiamsuretheXaresouri
tiseasXtoAesNisewhatXoucannotget

단어와 내용 추측

- sucessagainanAagain라는 패턴이 있다. 이것은 success again and again일 것이다(A→d).
- triedaLter라는 패턴이 보인다. 이것은 틀림없이 tried after이다(L→f).
- whatXoucannotget라는 패턴이 보인다. 이것은 what you cannot get일 것이다(X→y).

또 다시 정리하면

the fox and the grapes one hot summer sday(V→x, N→p)

thefoVandthegraNesonehotsummersdayafoVwasstro
KKingthroughanorchardtiKKhecametoaFunchofgraNe
sHustriNeningonaTinewhichhadFeentrainedoTeraKoft
yFranchHustthethingtoquenchmythirstquothhedrawin
gFacJafewNaceshetooJarunandaHumNandHustmiss
edtheFunchturningroundagainwithaonetwothreeheHu
mNeduNFutwithnogreatersuccessagainandagainhetri
edafterthetnmNtingmorseKFutatKasthadtogiTeituNan
dwaKJedawaywithhisnoseintheairsayingiamsuretheya
resouritiseasytodesNisewhatyoucannotget

좀더 정리해보면

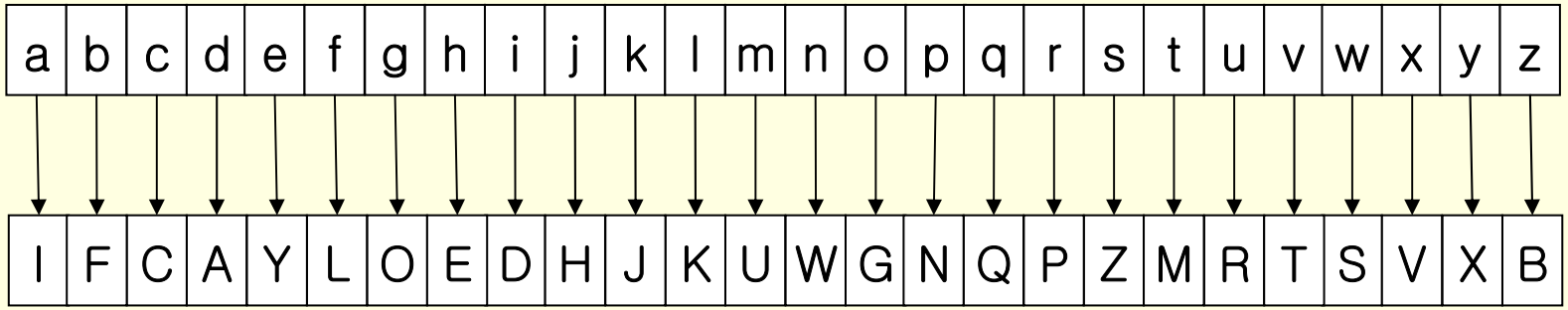
소문자의 비율이 늘어서 이제 거의 해독이 끝났다는 것을 알 수 있다.

thefoxandthegrapesonehotsummersdayafoxwasstroK
Kingthroughanorchardti KKhecametoaFunchofgrapes
HustripeningonaTinewhichhadFeentrainedoTeraKofty
FranchHustthethingtoquenchmythirstquothhedrawing
FacJafewpaceshetooJarunandaHumpandHustmisse
dtheFunchturningroundagainwithaonetwothreeheHu
mpedupFutwithnogreatersuccessagainandagainhetri
edafterthetnmptingmorseKFutatKasthadtogiTeitupan
dwaKJedawaywithhisnoseintheairsayingiamsuretheya
resouritiseasytodespisewhatyoucannotget

남은 사항을 처리하면

- foxwasstroKKing = fox was strolling (K → l)
- hetooJarunandaHumpandHustmissed
  = he took a run and a jump and just missed (H → j), (J → k)
- hejumpedupFutwithnogreatersuccess
  = he jumped up but with no greater success (F → b)
- butatlasthadtogiTeitup
  = but at last had to give it up (T → v)
- 이 암호문에 나오지 않은 마지막 1문자 (B → z)
- 이것으로 전부 해독이 되었다!

종합한 암호화 치환 키


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
I F C A Y L O E D H J K U W G N Q P Z M R T S V X B

해독된 평문

thefoxandthegrapesonehotsummersdayafoxw
asstrollingthroughanorchardtilhecametoabunc
hofgrapesjustripeningonavinewhichhadbeentr
ainedoveraloftybranchjustthetoquenchmythirst
quothhedrawingbackafewpaceshetookarunan
dajumpandjustmissedthebunchturningroundag
ainwithonetwothreehejumpedupbutwithnogrea
tersuccessagainandagainhetriedafterthetempti
ngmorselbutatlasthadtogiveitupandwalkedawa
ywithhisnoseintheairsayingiamsuretheyaresour
itiseasytodespisewhatyoucannotget

띄어쓰기가 끝난 평문

"The Fox and the Grapes"

One hot summer's day, a Fox was strolling through an orchard till he came to a bunch f grapes just ripening on a vine which had been trained over a lofty branch. "Just the to quench my thirst, "quoth he. Drawing back a few paces, he took a run and a jump, and just missed the bunch. Turning round again with one , two, three, he jumped up, but with no greater success. Again and again he tried after the tempting morsel, but at last had to give it up, and walked away with his nose in the air, saying: "I am sure they are sour." It is easy to despise what you cannot get.

해독작업의 결과

- 빈도가 높은 문자뿐만 아니라 빈도가 낮은 문자도 단서가 된다.
- 처음과 끝을 아는 것은 단서가 된다. 단어의 단락을 알면 그것도 단서가 될 수 있다.
- 암호문이 길면 해독이 쉬워진다.
- 같은 문자가 연속해서 나타나면 그것은 단서가 된다
- 해독의 속도가 점점 빨라진다.

2.3 다중 치환암호

- 단일 치환암호의 약점
  - 평문과 암호문간의 단순 대응을 사용하기 때문에 평문의 단일 문자에 대한 빈도가 그대로 암호문에 반영된다.
- 따라서 암호해독자로 하여금 빈도분석을 어렵게 하기 위해서는 암호문에 나타나는 문자들의 빈도를 거의 균등하게 만드는 암호를 이용하는 것이 바람직하다.

다중 치환암호(Polyalphabetic Substitution Cipher)

- 에서는 다중 치환을 이용하여 문자의 발생빈도를 균일화 한다.
- 다중 치환암호 방법의 대표적인 예
  - 비제네르(Vigenere) 암호
  - 힐(Hill) 암호.

치환 기법

- Hill 암호 기법
  - 각 문자에 정수 값을 부여하고 m개의 문자를 치환
    - M=3개는 3개의 문자를 치환하는 방법

$$C1 = (k11\ p1 + k12\ p2 + k13\ p3) \bmod 26$$
$$C2 = (k21\ p1 + k22\ p2 + k23\ p3) \bmod 26$$
$$C3 = (k31\ p1 + k32\ p2 + k33\ p3) \bmod 26$$

C: 암호문
P: 평문
k: 키

치환 기법

- 암호문 형식을 열 벡터와 행렬로 표현

$$\begin{vmatrix} C1 \\ C2 \\ C3 \end{vmatrix} = \begin{vmatrix} k11 & k12 & k13 \\ k21 & k22 & k23 \\ k31 & k32 & k33 \end{vmatrix} \begin{vmatrix} P1 \\ P2 \\ P3 \end{vmatrix}$$

- 암호화 사례
  - 평문: PAYMOREMONEY
  - 암호 키

$$K = \begin{vmatrix} 17 & 17 & 5 \\ 21 & 18 & 21 \\ 2 & 2 & 19 \end{vmatrix}$$

치환 기법

- 암호문 계산

$$\begin{vmatrix} C1 \\ C2 \\ C3 \end{vmatrix} = \begin{vmatrix} k11 & k12 & k13 \\ k21 & k22 & k23 \\ k31 & k32 & k33 \end{vmatrix} \begin{vmatrix} P1 \\ P2 \\ P3 \end{vmatrix}$$

- 평문을 숫자변환➔ PAYMOREMONEY: P ➔15, A ➔ 0, Y ➔ 24, …
- 숫자 대입 암호문 치환

$$\begin{vmatrix} C1 \\ C2 \\ C3 \end{vmatrix} = \begin{vmatrix} 17 & 17 & 5 \\ 21 & 18 & 21 \\ 2 & 2 & 19 \end{vmatrix} \begin{vmatrix} 15 \\ 0 \\ 24 \end{vmatrix} \text{ mod } 26 \rightarrow \begin{vmatrix} 11 \\ 13 \\ 18 \end{vmatrix} \begin{vmatrix} L \\ N \\ S \end{vmatrix}$$

- K(15 0 24) + (375 819 486) mod 26 = (11 13 18) = LNS
  - C1 = 17 x 15 + 17 x 0 + 5 x 24 = 375 mod 26 = 14 … …11
  - C2 = 21 x 15 + 18 x 0 + 21 x 24 = 819 mod 26 = 31 … …13
  - C3 = 2 x 15 + 2 x 0 + 19 x 24 = 486 mod 26 = 18 … …18

치환 기법

- 복호문 계산
  - 암호문 계산 형식 $C = E_K(P) = KP$에서
  - 평문 $P = D_K(C) = K^{-1}C = K^{-1}KP = P$; 여기서, K^{-1}는 역행열: $K^{-1}K = I$

$$\begin{vmatrix} P1 \\ P2 \\ P3 \end{vmatrix} = \begin{vmatrix} 4 & 9 & 15 \\ 15 & 17 & 6 \\ 24 & 0 & 7 \end{vmatrix} \begin{vmatrix} 11 \\ 13 \\ 18 \end{vmatrix} \quad \text{mod } 26 \rightarrow \quad \begin{vmatrix} 15 \\ 0 \\ 24 \end{vmatrix} \begin{vmatrix} P \\ A \\ Y \end{vmatrix}$$

  - 역행렬 계산

$$\begin{vmatrix} 17 & 17 & 15 \\ 21 & 18 & 2 \\ 2 & 2 & 19 \end{vmatrix} \begin{vmatrix} 4 & 9 & 15 \\ 15 & 17 & 6 \\ 24 & 0 & 7 \end{vmatrix} = \begin{vmatrix} 443 & 442 & 442 \\ 858 & 495 & 780 \\ 494 & 52 & 365 \end{vmatrix} \quad \text{mod } 26 \rightarrow \quad \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

치환 기법

- 다중 단일 문자 치환 암호 기법
  - 관련된 단일 문자 치환 규칙들의 집합을 사용함
  - 주어진 변환에 사용될 특정 규칙은 키에 의해 결정됨

- 대표적인 Vigenere 암호 방식
  - 행렬표를 구성
  - 키 문자 x와 평문자 y가 주어지면 암호 문자는 x행 y 열의 암호문 V

키	d	e	c	e	p	t	i	v	e	d	e	c	e	p	t	i	v	e	d	e	c	e	p	t	i	v	e
평문	w	e	a	r	e	d	i	s	c	o	v	e	r	e	d	s	a	v	e	y	o	u	r	s	e	l	f
암호문	Z	I	C	V	T	W	Q	N	G	R	Z	G	V	T	W	A	V	Z	H	C	Q	Y	G	L	M	G	J

  - 평문 길이 만큼 키 크기가 필요
  - 언어의 특징을 모두 없애지 못함

현대 VIGENRE 표

	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
a	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
b	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	A
c	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
d	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
e	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
f	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
g	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
h	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
i	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
j	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
k	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
l	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
m	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
n	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
o	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
p	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
r	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
s	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
t	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
u	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
v	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
w	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
x	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
z	Z	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

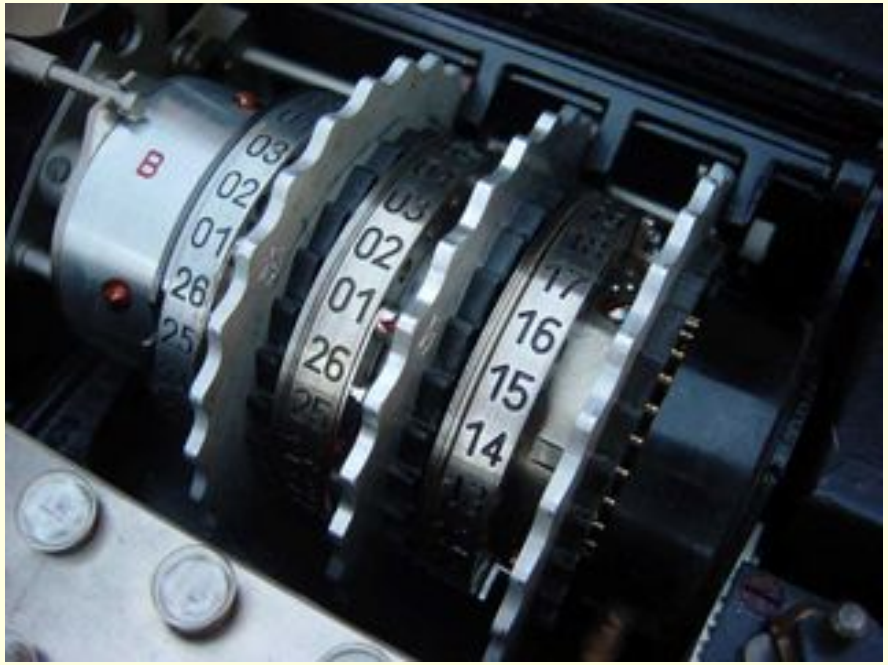
2.4 애니그마

- 제2차 세계대전 중에 독일에서 사용된 암호기계
- 「애니그마」란?

2.4.1 애니그마란

- 독일의 세르비우스에 의해 20세기 초에 발명된, 암호화/복호화를 행하는 기계이다.
- 애니그마라는 이름은 독일어로 「수수께끼」를 의미한다.

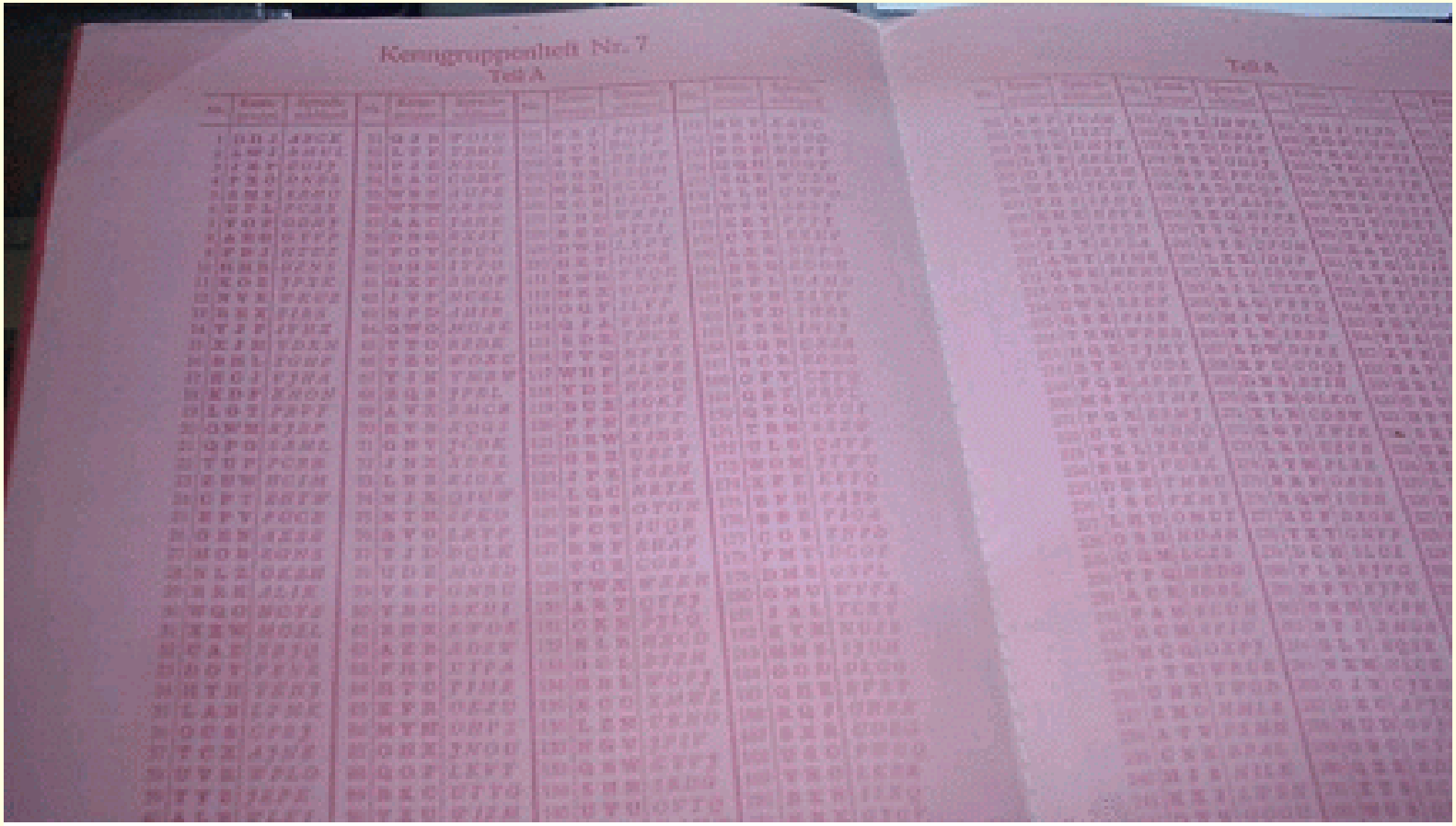
애니그마와 애니그마의 내부에 사용되는 로터



2.4.2 애니그마에 의한 암호통신

- 송신자와 수신자는 애니그마를 1 대씩 가지고 있어야 한다.
- 송신자와 수신자가 같은 키를 사용하지 않으면 암호통신은 할 수 없으므로 송신자와 수신자는사전에 코드북을 가지고 있다
  - 코드북에는 송/수신자가 사용하는 **날짜별 키**가 기록
  - 송신자/수신자는 이 책자의 지시에 따라 애니그마를 설정한다.

애니그마에서 사용된 코드북



2.4.3 애니그마의 구조

- 입력용 키보드의 **키**를 하나 누르면 전기신호가 복잡한 회로를 거쳐 최종적으로 출력용 **램프**가 점등한다.
- 그림 2-13에서는 a의 키를 눌렀을 때 D 램프가 점등하는 모습을 보이고 있다.

애니그마를 사용한 암호통신의 흐름

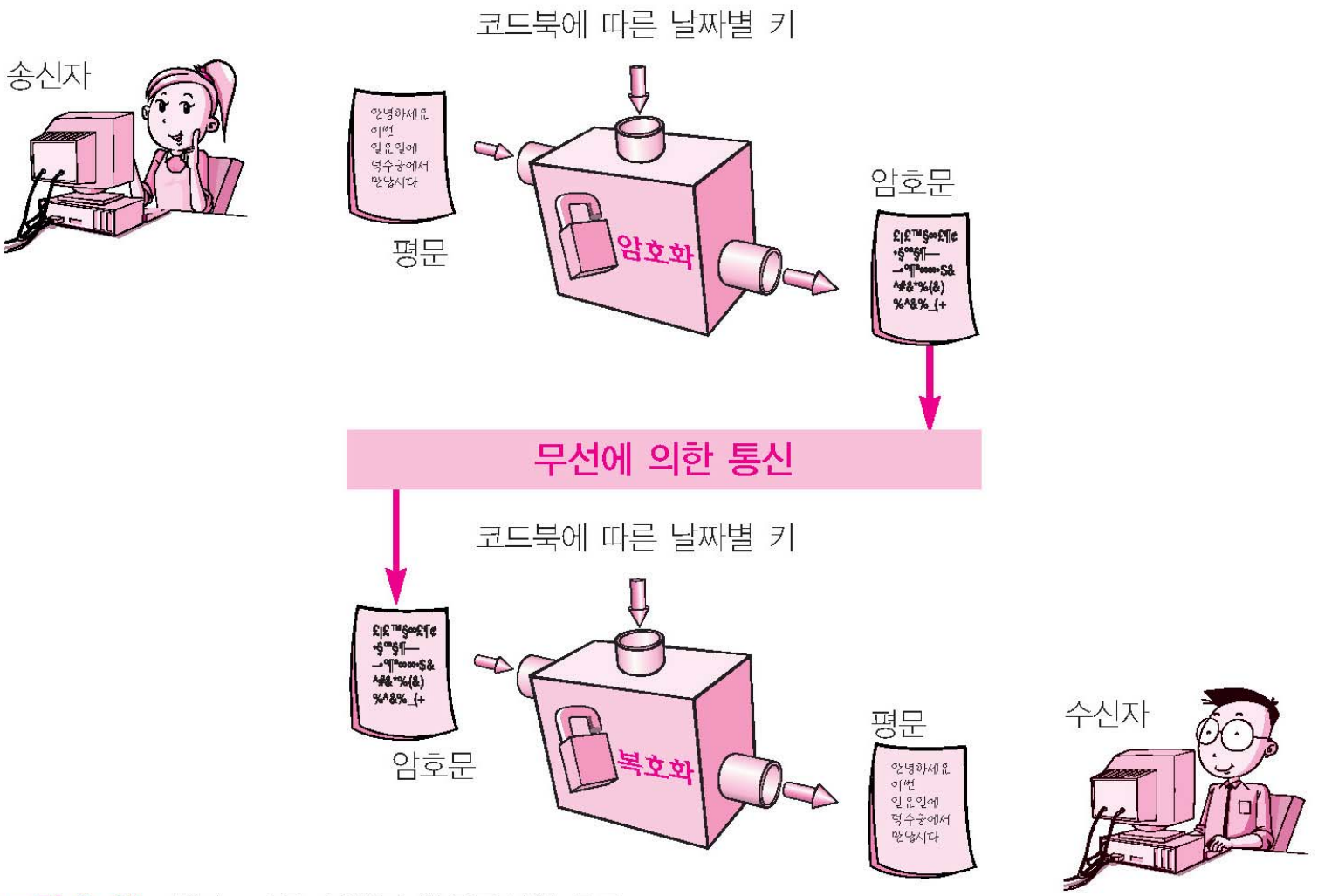


그림 2-12 애니그마를 사용한 암호통신의 흐름

애니그마의 구조
(알파벳의 수를 4문자로 했을 경우)

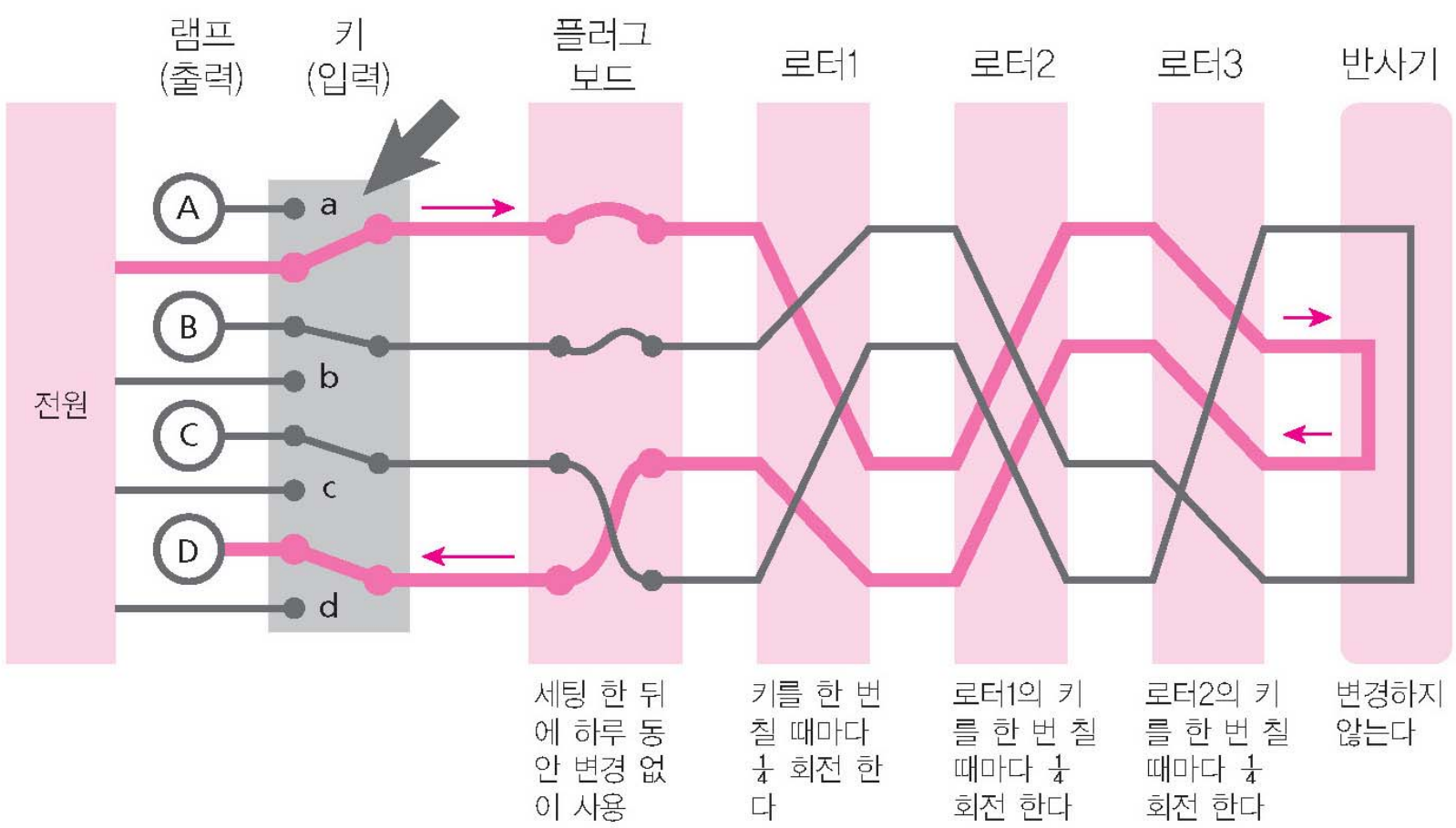
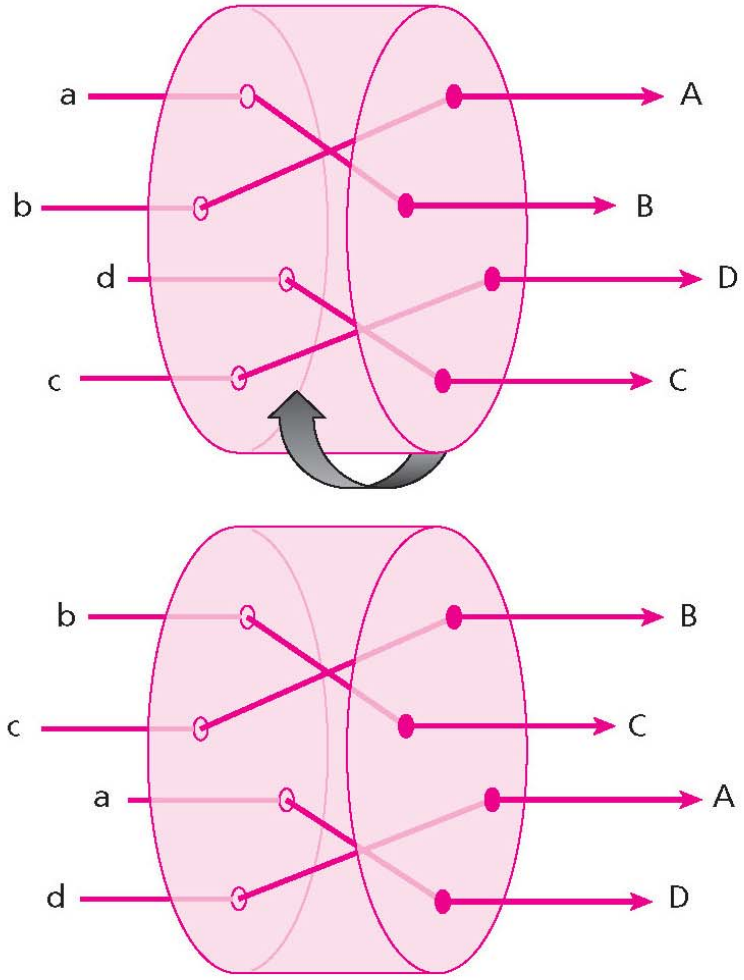


그림 2-13 애니그마의 구조(알파벳의 수를 4문자로 했을 경우)

로터



초기상태 치환표
a → B
b → A
c → D
d → C

1/4 회전 후의 치환표
a → D
b → C
c → B
d → A

그림 2-14 로터(rotor)

2.4.4 애니그마의 암호화

- 예: 송신자가 독일어로 nacht(밤)이라는 5 문자를 암호화해서 송신하는 방법

애니그마의 설정

- 송신자는 코드북을 참조해서 「그 날의 **날짜별 키**」를 조사하여 그 날짜별 키대로 애니그마를 설정한다. 구체적으로는 플러그 보드의 연결선을 연결하고, 3장의 로터의 배열을 바꾸게 된다.

통신키의 암호화

- 송신자는 알파벳 3문자를 정하고 암호화한다. 이 알파벳 3 문자를 **통신 키**라고 한다.
- 통신 키의 암호화는 애니그마를 써서 행한다. 지금 송신자가 고른 통신키가 psv라고 하면, 송신자는 애니그마의 키보드로 통신키를 2회 연속해서 친다.
- 즉, psvpsv라는 6 문자를 치게 된다.
- 한 문자 칠 때마다 로터가 회전하고 램프가 켜진다.
- 송신자는 점등된 램프에 대응한 문자를 메모한다.
- 6 문자를 다 치면 점등한 6 문자가 메모되어 있게 된다. 여기서는 점등된 램프의 6문자가 ATCDVT라고 하자

애니그마의 재설정

- 다음에 송신자는 통신키에 따라 애니그마의 재설정을 행한다.
- 실은 통신키의 알파벳 3문자는 3장의 로터의 설정각도를 나타내고 있다.
- 1장의 로터 주위에는 목표 문자가 기록되어 있고, 문자에 대응한 각도를 설정할 수 있게 되어 있다.
- 통신키 psv는 로터1, 2, 3을 각각 p의 각도, s의 각도, v의 각도로 한다는 것을 의미한다.

메시지의 암호화

- 다음에 송신자는 메시지를 암호화한다.
- 송신자는 메시지(평문)를 한 문자 한 문자 키보드로 입력하고 그 결과를 램프에서 읽어서 메모한다.
- 여기에서는 natch라는 5문자를 키로 입력하고, 결과의 5문자(예를 들면 KXNWP)를 메모하게 된다.

nacht(밤)이라는 5 문자를 암호화해서 송신하는 방법

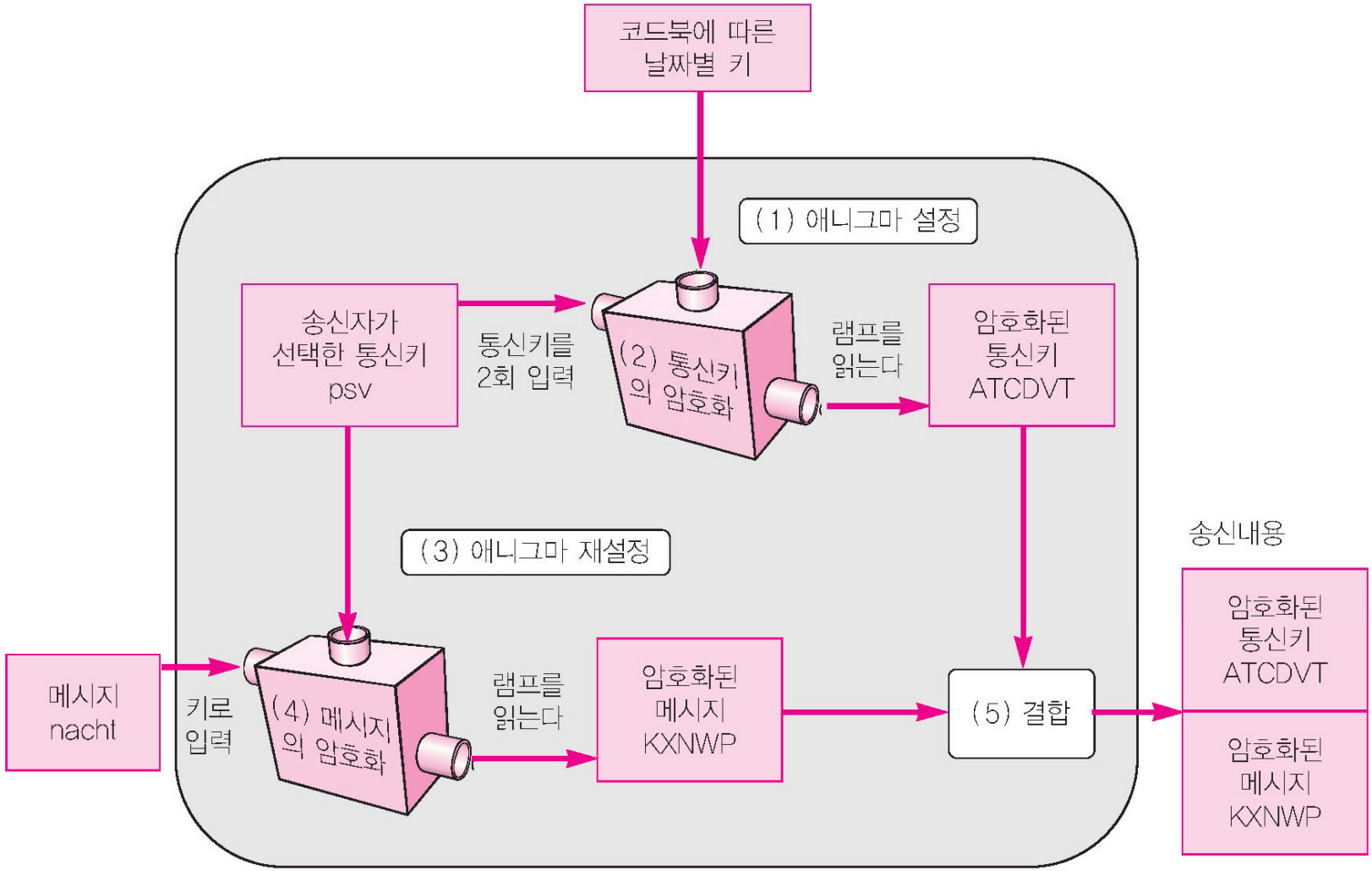


그림 2-15 애니그마를 써서 nacht를 암호화한다

결합

- 마지막으로 송신자는 「암호화된 통신키」 ATCDVT와 「암호화된 메시지」 KXNWP를 결합하여 ATCDVT KXNWP라는 통신문을 무선으로 송신한다.

2.4.5 「날짜별 키」와 「통신키」

- 날짜별 키는 메시지의 암호화가 아니라 통신키의 암호화에 사용되었다.
- 즉, 날짜별 키는 「키를 암호화하기 위한 키」가 된다.
- 이와 같은 키를 가리켜 일반적으로 **키 암호 키** (key encrypting key ; KEK)라고 부른다.

2.4.6 통신 오류의 회피

- 애니그마가 사용된 시대는 무선 기술이 충분히 발달하지 않아서 통신이 제대로 되지 않는 경우가 많이 있었기 때문이다.
- 통신 키가 바르게 보내지지 않으면 수신자가 통신문을 복호화 할 수 없다.
- psvpsv라고 통신키를 2회 연속해서 키를 쳐서 암호화하면 수신자 측에서 통신 키를 검증할 수 있다.

2.4.7 애니그마의 복호화

- 분해
  - 수신한 통신문 처음의 6 문자 ATCDVT와 나머지 문자 KXNWP를 분리
- 애니그마의 설정
  - 코드북을 참조해서 「그 날의 날짜별 키」를 조사하여 그 날짜별 키대로 애니그마를 설정
- 통신키의 복호화
  - 수신자는 암호화된 통신키 ATCDVT를 복호화한다.
  - 수신자는 애니그마의 키보드로 ATCDVT라는 6문자를 쳐서 불이 켜지는 6문자 psvpsv를 메모한다

2.4.7 애니그마의 복호화

- 애니그마의 재설정
  - 수신자는 통신키 psv를 이용하여 애니그마를 재설정
- 메시지의 복호화
  - 메시지를 복호화한다.
  - 통신문의 나머지 KXNWP를 한 문자 한 문자 키보드로 입력하고 그 결과를 램프에서 읽어 메모한다.
  - natch라는 5 문자가 메모되는 것을 알 수 있고 송신자로부터 전송된 메시지의 복호화가 끝나게 된다.

애니그마를 사용한 복호화

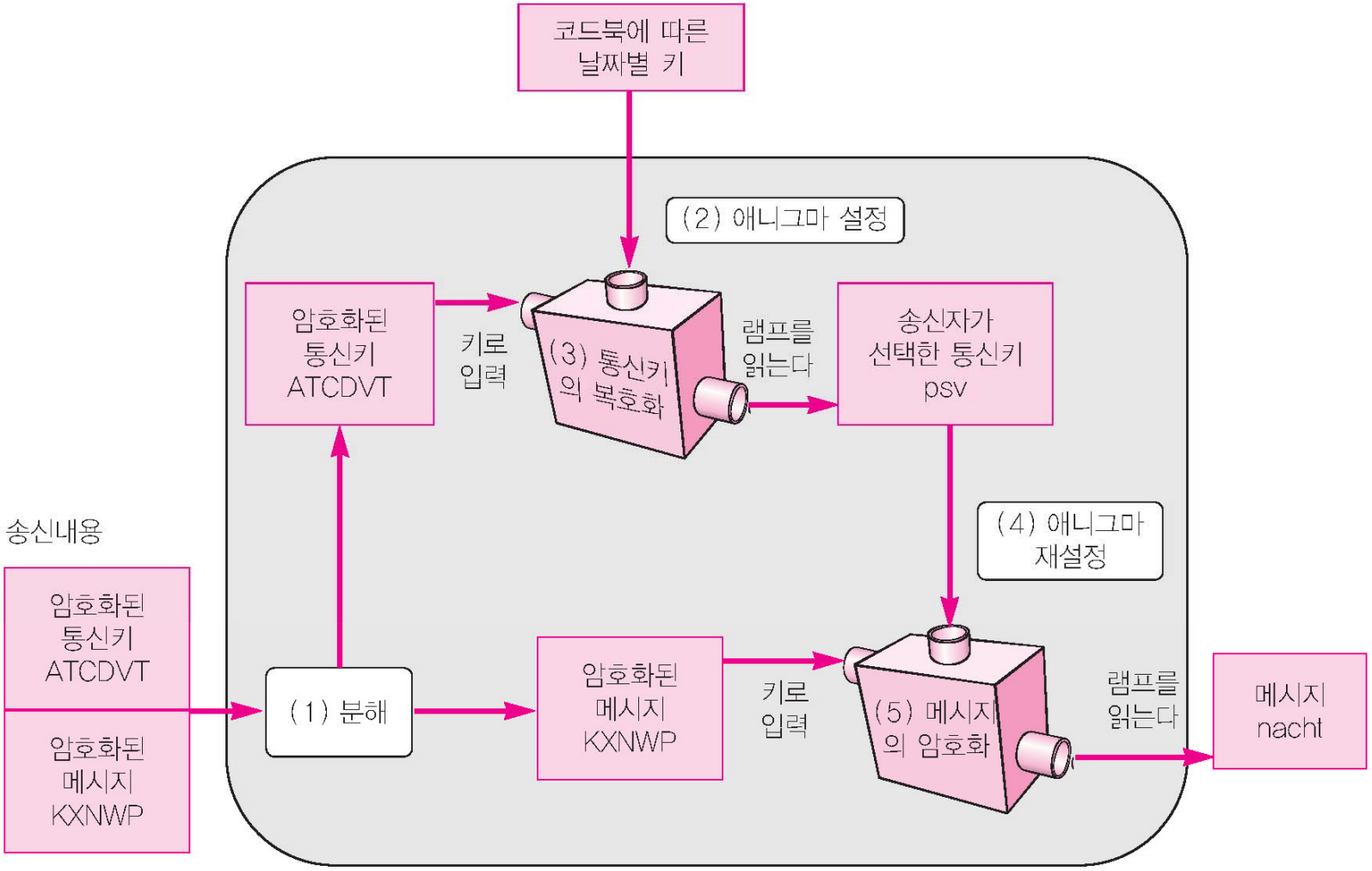


그림 2-16 애니그마를 사용한 복호화

2.4.8 애니그마의 약점

- 「통신 키의 암호화」라는 중요한 처리(처음에 키를 6번 치는 것) 동안 실제로 회전하는 것은 로터1뿐이다.
- 「통신키를 2회 반복한 것을 암호화 한다」
- 「통신키를 선택한 것이 사람이다」
- 「코드북을 배송하지 않으면 안 된다」

2.4.9 애니그마의 해독

- 해독의 시작은 프랑스와 영국의 암호 해독자가 스파이 활동으로 독일군이 사용하고 있는 애니그마의 구조정보를 입수
- 프랑스로부터 정보제공을 받은 폴란드의 암호 해독자 르예프스키였다.
  - 르예프스키는 날짜별 키에 의한 암호문으로부터 날짜별 키를 간파하는 방법을 고안하였다.
  - 대량의 암호문을 근거로 해서 약 2시간 만에 날짜별 키를 찾아낼 수 있었다

2.5 암호 알고리즘과 키

- 지금까지 우리가 본 암호 알고리즘에서는 키를 항상 사용하였는데 왜 키를 사용하는 것일까?

2.5.1 암호 알고리즘과 키의 분리

암호명	암호 알고리즘	키
시저 암호	평문의 각 문자를 '지정한 문자 수'만큼 평행 이동한다.	평행 이동하는 문자 수
단일 치환 암호	치환표에 따라 알파벳을 변환한다.	치환표
애니그마 (통신키의암호화)	애니그마의 기계를 써서 『플러그 보드의 연결선, 3장의 로터의 순서, 각 로터의 설치 각도』에 따라 알파벳을 변환한다.	• 플러그 보드의 연결선 • 3장의 로터 순서 • 각 로터의 설치 각도
애니그마 (통신문의암호화)	플러그 보드의 연결선과 3장의 로터의 순서를 고정한 애니그마 기계를 사용하여 『각 로터의 설치 각도』에 따라 알파벳을 변환한다.	각 로터의 설치 각도

암호알고리즘과 키의 관계

- 이와 같이 「암호 알고리즘」과 「키」의 조합을 잘 조사해 보면 암호 알고리즘 안에는 「변경 가능한 부분」이 반드시 포함되어 있다는 것을 알 수 있다.
- 암호 알고리즘 안의 「변경 가능한 부분」이 「키」에 해당한다.

「암호 알고리즘」과 「키」를 나누어 생각한다

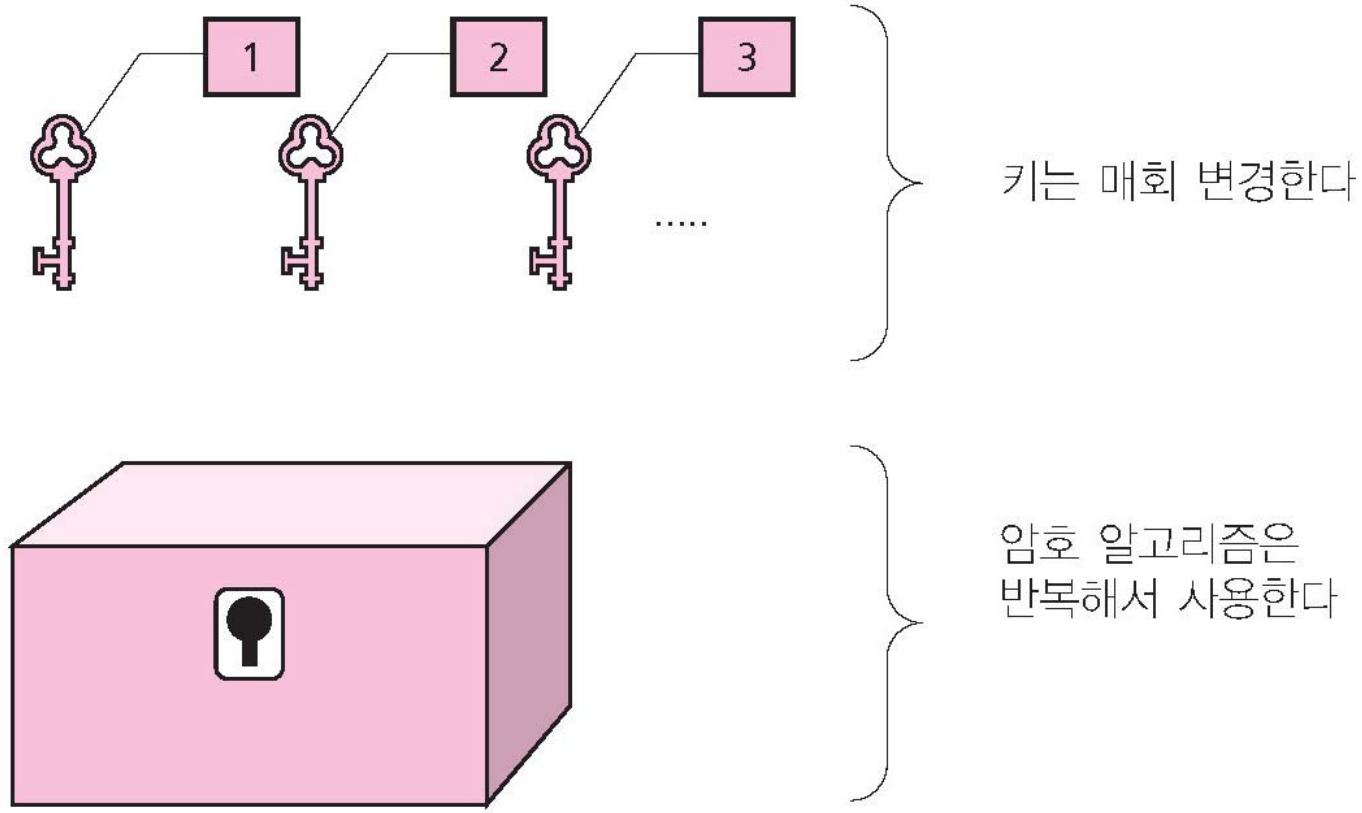


그림 2-17 「암호 알고리즘」과 「키」를 나누어 생각한다

질의 및 응답

– 끝 –