Chapter

13

네트워크 보안

* 학습목표

- 암호화 원리를 바탕으로 대체 암호화와 위치 암호화를 알아본다.
- 암호화 알고리즘인 DES, RSA의 구조를 이해한다.
- 전자 서명의 필요성과 방법을 이해한다.
- 네트워크 보안의 개념과 관련 이슈를 살펴본다.
- 라우터와 프록시로 구현한 방화벽의 원리를 이해한다.

Preview

컴퓨터 네트워크의 사용자 환경이 과거처럼 단순히 정보를 검색하고 활용하는 데 머무르지 않고, 홈뱅킹, 전자 상거래 같은 상업 분야로 확대되었다. 이에 따라 온라인의 불법 침입자가 중요 정보에 접근할 수 없도록 차단하는 네트워크 보안(Network Security)의 중요성이 강조되고 있다.

보안의 필요성은 두 가지로 생각할 수 있다. 첫째, 허가 받지 않은 외부 침입자에게 정보가 유출되지 않도록 하기 위해서다. 보안 데이터가 전달되는 네트워크 경로에서 데이터를 불법적으로 훔쳐내 악용하는 경우가 이에 해당한다. 둘째, 외부 침입자가 보안 데이터의 내용을 조작하지 않도록 보호하기 위해서다. 데이터 조작은 통신하는 사람들 간의 정상적인 데이터 통신을 방해한다.

Section

01 암호화의 이해

문서의 내용을 **암호화(Encryption)**하여 목적지에 전송함으로써, 외부 침입자로부터 문서를 보호하는 방법은 컴퓨터 네트워크가 보급되기 전부터 사용하던 방식이다. 이때 문서의 송수신자는 암호문을 작성하고 해석하는 과정에서 자신들만 아는 비밀키를 사용한다.

컴퓨터 보안은 일반인도 중요성을 인식하는 분야지만, 네트워크에서는 이론적으로 간단하지 않은 분야에 속한다. 따라서 관련 용어를 살펴봄으로써 전체 시스템을 개괄적으로 이해해보자.

1. 암호화 관련 용어

컴퓨터 네트워크에서 송신자와 수신자는 중간 전송 매체를 통해 메시지를 송수신한다. 네트워크는 기본적으로 개방형 시스템이므로 제삼자가 임의로 접근할 수 있고, 전송 메시지는 외부 접근에 노출되어 있다. 따라서 전송 과정에서 비권한적 접근을 시도하는 침입자(Intruder)로부터 메시지를 안전하게 보호해야 한다.

외부 침입자가 전송 메시지에 가하는 위해(危害) 행동은 다음의 세 가지로 나눌 수 있다.

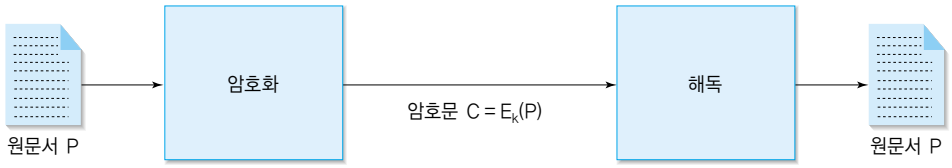
- **메시지 읽기** : 전송 선로를 흐르는 신호를 도청하여 메시지의 내용을 읽는다. 인터넷에서 신호 도청을 차단하기는 쉽지 않으므로 암호화 기법으로 해결해야 한다.
- **전송 방해** : 전송 메시지가 수신자에게 도착하지 못하게 함으로써 송수신자 간의 통신을 방해한다. 인터넷에서 방화벽 기능을 통해 불법 사이트에 접속하지 못하도록 차단하는 것이 이에 해당한다.
- **메시지 수정** : 전송되는 메시지의 내용을 수정하는 것으로, 송수신자가 교환하는 메시지의 의미를 왜곡시킨다.

암호화 용어

외부 침입자가 송수신자 사이에 전송되는 메시지를 읽거나 수정하는 등의 위해 행위를 막기 위해 컴퓨터 네트워크에서 사용하는 일반 기법은 암호화다.

암호화는 메시지의 내용을 변형하여 원래의 의미를 알아볼 수 없도록 변형하는 작업이다. 메시지의 의미는 언어로 표현(Encoding)되기 때문에 송수신자만 해독할 수 있는 표현 방식을 사용해 침입자가 메시지의 내용을 알아볼 수 없게 만들어 전송해야 한다. 암호화된 문서를 원래 언어로 변형하여 수신자가 알아보기 위해서는 **해독(Decryption)** 과정을 거쳐야 한다.

[그림 13-1]은 왼쪽에서 오른쪽으로 메시지를 전송하는 과정에서 암호화와 해독 기능을 설명한다. 암호화 전의 원본 문서를 **원문서(Plaintext)**라고 하며, 이를 임의의 형태로 암호화한 문서를 **암호문(Ciphertext)**이라고 한다. 암호문은 미리 정해진 **암호키** k를 이용하므로 $C = E_k(P)$의 형식으로 암호화한다.

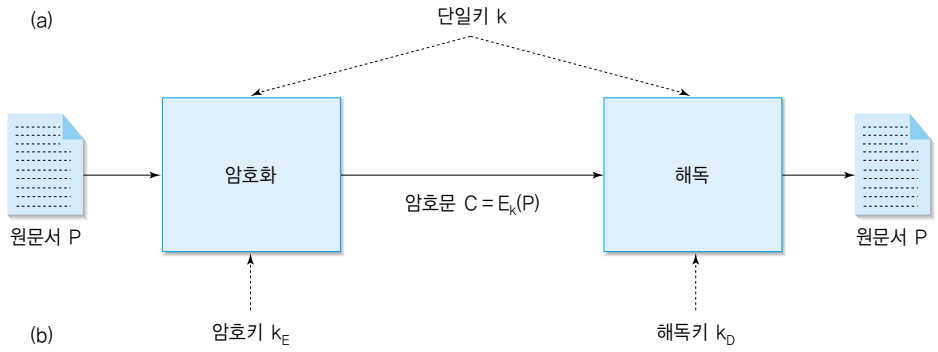


[그림 13-1] **암호화 과정과 용어**

암호화 과정은 송신 호스트에서 메시지를 전송하기 전에 이루어지며, 해독 과정은 수신 호스트에서 암호문을 수신하는 과정에서 처리된다. 따라서 송수신자 사이의 전송 매체에 의하여 외부에 노출될 가능성이 있는 경우는 암호문 상태로 전송될 때다. 외부 침입자가 암호문을 해독할 능력이 없으면 메시지의 내용을 읽거나 수정하는 등의 위해 행동을 할 수 없다.

암호화 알고리즘

암호화와 해독 과정에서는 **키(Key)**를 사용하는데, 키 종류에는 암호키와 해독키가 있다. [그림 13-2]에서 (a)는 암호화와 해독 과정에서 동일한 하나의 키를 사용하는 경우다. 그리고 (b)는 암호화 과정에서 사용하는 암호키와 해독 과정에서 사용하는 해독키가 서로 다른 경우다.



[그림 13-2] 키의 종류

대칭키(Symmetric Key)라고도 하는 단일키(Single Key) 방식을 사용할 때는 송수신자 외의 제삼자가 키 값을 알지 못하도록 하는 것이 중요하다. 그런데 기본적으로 둘 이상의 호스트가 키 값을 알기 때문에 보안을 유지하기가 쉽지 않다. 암호키 관리를 위해 송수신자가 주기적으로 키 값을 변경할 수 있는데, 이를 처리하는 과정에서 특히 외부로 유출되지 않도록 주의해야 한다.

[그림 13-2(b)]처럼 송신자와 수신자가 서로 다른 키를 사용하는 것을 **비대칭키(Asymmetric Key)** 방식이라 한다. 이 방식에서는 보통 키 하나는 공개되므로 공개되지 않는 나머지 키를 송신 호스트, 혹은 수신 호스트 스스로가 보안에 주의해야 한다.

2. 대체 암호화

암호문을 작성하는 방법은 사용하는 알고리즘에 따라 다양하다. 가장 간단한 방식은 특정 문자를 다른 문자로 1:1 대체하는 것인데, 이때는 문자와 대체 문자를 나열한 표가 암호키와 해독키가 된다. 이와 같이 임의의 문자를 다른 문자로 대체하는 암호화를 **대체 암호화(Substitution Cipher)**라고 한다. 대체 암호화 방식의 예로는 시저 암호화, 키워드 암호화, 복수 개의 문자표 사용 방식 등이 있다.

시저 암호화

시저 암호화(Caesar Cipher)는 줄리어스 시저가 처음 사용했을 것이라는 의미에서 붙은 이름이다. 알파벳 문자를 순차적으로 세 문자씩 오른쪽으로 이동하면서 대체 문자를 사용하는

방식이다. 시저 암호화에서 암호키에 해당하는 문자 변환표는 다음과 같다. 편의상 원문서의 문자는 대문자로, 암호문의 문자는 소문자로 표기하였다.

원문	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
암호문	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

위의 대체 문자표를 이용해 NETWORK TECHNOLOGY를 암호화하면 qhwzrun whfkqrorjb가 된다. 시저 암호화 방식은 가장 초보적인 알고리즘임에도 불구하고, 원문서 없이 암호문만으로 원래의 문자를 찾기가 쉽지 않다.

N	E	T	W	O	R	K		T	E	C	H	N	O	L	O	G	Y
q	h	w	z	r	u	n		w	h	f	k	q	r	o	r	j	b

시저 암호화 방식의 장점은 단순함이다. 특히, 세 문자 간격으로 이동된 암호키를 쉽게 기억할 수 있으므로 암호문의 작성과 해독이 간단한 수작업만으로도 가능하다. 그러나 이런 단순함은 외부 침입자도 쉽게 해독할 수 있다는 단점도 된다.

키워드 암호화

시저 암호화 방식의 단점을 부분적으로 보완한 대체 암호화로 **키워드 암호화(Keyword Cipher)**가 있다. 키워드 암호화는 키워드로 지정한 단어를 암호문 앞줄에 먼저 적고, 키워드에 표시된 문자를 뺀 나머지 문자를 알파벳 순으로 기술한다. 예를 들어, seoul을 키워드로 사용하면 대체 문자표는 다음과 같다.

원문	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
암호문	s	e	o	u	l	a	b	c	d	f	g	h	i	j	k	m	n	p	q	r	t	v	w	x	y	z
	키워드					s, e, o, u, l 을 제외한 문자를 알파벳 순서로 배치																				

키워드를 모르면 시저 암호화보다 대체 문자표를 추측하기가 훨씬 어렵다. 그러나 대체 문자표의 오른쪽으로 갈수록 원문과 암호문의 문자표가 같아질 확률이 높아 시저 암호화보다 나쁜 결과를 초래할 수 있다. 위 예에서도 v 문자부터 시작하여 w, x, y, z 다섯 문자는 대체 문자가 자신과 같다. 다행히 오른쪽에 위치한 알파벳 문자들은 일반 영어 문장에서 많이 사용되지 않는다.

복수 개의 문자표

앞서 소개한 방식은 대체 문자표가 하나므로 침입자가 해독할 가능성이 높다. 이를 보완하려고 문자표를 둘 이상 사용할 수 있다. 예를 들어, 앞서 설명한 두 개의 대체 문자표를 모두 활용할 수 있다. 즉, 첫 번째 문자표는 홀수 위치에 있는 문자를 암호화하는 데 사용하고, 두 번째 문자표는 짝수 위치에 있는 문자를 암호화하는 데 사용한다.

홀수 위치에 있는 문자

원문	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
암호문	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

짝수 위치에 있는 문자

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

위의 두 대체 문자표를 사용해 NETWORK TECHNOLOGY를 암호화하면 qlwwrpn rhukjrhrbb가 된다. 이 방식의 장점은 원문서의 동일 단어가 암호문에서는 다르게 암호화되므로 해독을 더 어렵게 한다는 것이다. 예를 들어, NETWORK에 있는 T는 w로 대체되고 TECHNOLOGY의 T는 r로 대체되었다. 이는 문자의 위치에 따라 대체되는 문자표가 다르기 때문이다.

1	2	3	4	5	6	7		8	9	10	11	12	13	14	15	16	17
N	E	T	W	O	R	K		T	E	C	H	N	O	L	O	G	Y
q	l	w	w	r	p	n		r	h	u	k	j	r	h	r	b	b

3. 위치 암호화

대체 암호화 방식에서는 원문서에 포함된 각 문자의 배열 순서가 일정하게 유지된다. 대신 특정 문자를 다른 문자로 대체함으로써 암호문을 해독하기 어렵게 한다.

이와 달리 문자의 배열 순서를 변경해 암호화하는 **위치 암호화**(Transposition Cipher) 방식이 있다. 위치 암호화는 각 문자의 모양은 그대로 유지한 채, 문자의 배열 위치를 임의로 변경하여 암호화한다.

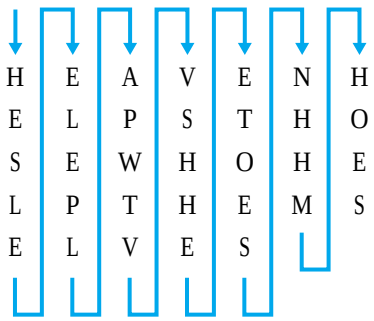
위치 암호화 방식의 예로는 컬럼 암호화, 키워드 암호화 등이 있다.

컬럼 암호화

위치 암호화의 가장 간단한 방식 중 하나인 **컬럼 암호화(Column Cipher)**는 첫 번째부터 마지막 컬럼까지 전체 문장을 컬럼을 기준으로 다시 배치한다. 예를 들어, 컬럼의 길이가 7인 컬럼 암호화 방식에서 Heaven helps those who help themselves라는 문장을 생각해보자.

원문서 HEAVEN HELPS THOSE WHO HELP THEMSELVES

위 문장을 일반적인 문장 표기 방식에 따라 한 줄씩 오른쪽으로 적으면 다음과 같다. 단, 컬럼의 길이가 7로 제한되어 있으므로 한 줄에는 최대 7문자만 표시할 수 있다.



그런 다음, 맨 왼쪽의 첫 번째 컬럼부터 시작해 오른쪽으로 이동하면서 표기하여 암화화하면 다음과 같다.

암호문 1 hesle elepl apwtv vshhe etoes nhhm hoes

위의 예에서는 6번째 컬럼과 7번째 컬럼의 마지막에는 문자가 존재하지 않는다. 일반적으로 컬럼 암화화 방식은 맨 마지막 줄에도 데이터를 채우기 때문에 빈 공간이 있으면 임의의 문자를 채워 암호화한다. 예를 들어, 공백을 채우는 문자를 Z라고 가정하면 최종 암호문은 다음과 같다.

암호문 2 hesle elepl apwtv vshhe etoes nhhmz heosz

컬럼 암호화를 두 번 수행하는 이중 컬럼 암호화 방식은 해독을 더 어렵게 할 수 있다. 이때 첫 번째 컬럼 암호화와 두 번째 컬럼 암호화의 컬럼 길이를 다르게 하는 것이 유리하다.

키워드 암호화

일반적으로 위치 암호화 방식에서는 중복된 문자를 포함하지 않는 임의의 단어를 암호키로 제공하고 **키워드 암호화(Keyword Cipher)**가 사용된다. 예를 들어, NETWORK라는 단어를 키워드로 사용해 위치 암호화를 하는 과정을 살펴보자.

먼저 키워드 NETWORK의 각 문자 N, E, T, W, O, R, K에 알파벳 순으로 일련번호를 부여하고 그 아래에 원문서의 문자를 차례대로 적는다. 이 경우 컬럼의 길이는 키워드 단어의 문자 길이와 동일하므로 원문서 Heaven helps those who help themselves를 배치하면 다음 구조가 된다.

키워드	N	E	T	W	O	R	K
순서	3	1	6	7	4	5	2
	H	E	A	V	E	N	H
	E	L	P	S	T	H	O
	S	E	W	H	O	H	E
	L	P	T	H	E	M	S
	E	L	V	E	S	Z	Z

암호문을 만드는 과정은 맨 왼쪽 컬럼부터 시작했던 컬럼 암호화와 다르게 키워드의 알파벳 순을 따른다. 즉, 먼저 순서 번호 1인 E 문자의 열에 포함된 elepl을 얻어낸다. 다음에는 순서 번호 2인 K 문자의 열에 포함된 hoesz를 표기하고, 계속해서 순서 번호 3, 4, 5, 6, 7열에 포함된 문자를 차례로 표시한다. 최종 암호문은 다음과 같다.

원문서 HEAVEN HELPS THOSE WHO HELP THEMSELVES

암호문 elepl hoesz hesle etoes nhhmz apwtv vshhe

Section

02 암호화 시스템

암호문은 기본적으로 대체 암호화와 위치 암호화 방법을 적절히 조합하여 작성한다. 컴퓨터가 보급되기 전에는 일반적으로 수작업으로 암호화하였다. 그래서 암호화 알고리즘이 간단한 대신 암호키에 문자를 많이 사용해 해독을 어렵게 만들었다.

최근에는 고성능 컴퓨터가 보급됨에 따라 연산 속도가 빨라져 알고리즘의 복잡도를 증가시키는 방식으로 암호화를 한다. 대표적인 예로는 DES와 RSA 알고리즘이 있다.

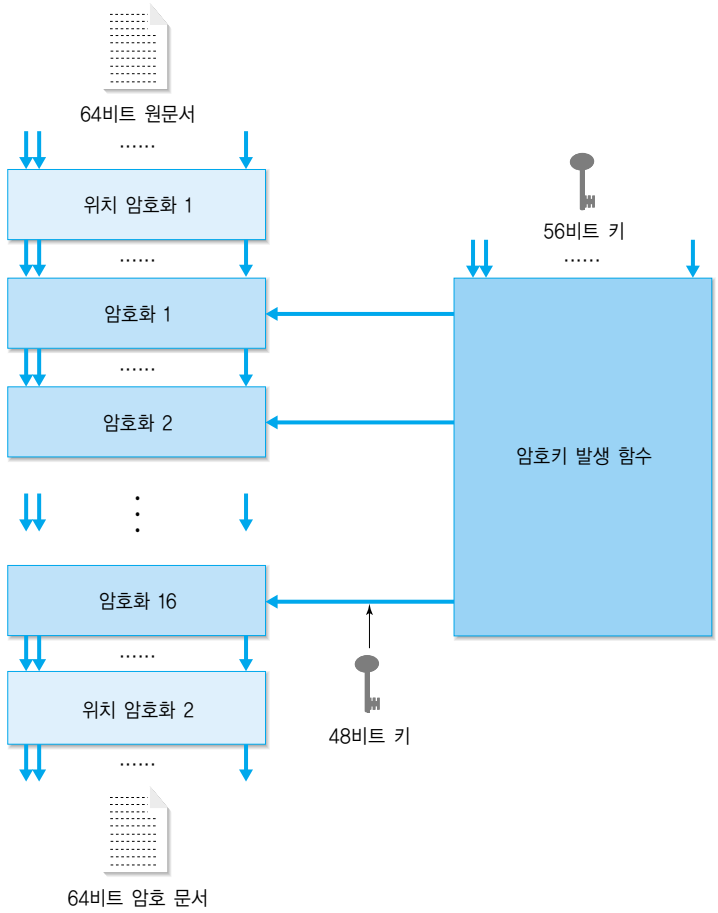
1. DES 알고리즘

DES 알고리즘은 암호문을 작성할 때 사용하는 암호키와 암호문을 해독할 때 사용하는 해독키가 같다. 따라서 이 키는 절대로 외부에 유출되지 않도록 관리해야 하여 비밀키(Secret Key)라고 부른다. 또한 양쪽이 동일 키를 사용한다고 해서 대칭키(Symmetric Key)라고도 한다. 이처럼 외부 사용자에게 노출되지 않아야 하는 암호키로 암호화하는 알고리즘을 **비공개키 알고리즘**이라고 한다.

비공개키 방식의 **DES(Data Encryption Standard)** 알고리즘은 미국 정부가 개발하여 여러 하드웨어와 소프트웨어에서 사용되어 왔다. 대체 암호화와 위치 암호화를 복잡하게 조합하여 개발한 DES 알고리즘은 암호화를 64비트 단위로 수행하며, 암호키의 크기는 56비트다.

동작 방식

DES 알고리즘은 크기가 64비트인 데이터 블록을 32비트씩 둘로 나누어 독립적으로 처리한다. 32비트 블록 하나를 암호키로 암화화한 후에, 두 블록의 위치를 맞바꾸는 과정을 16번 반복하는데 이 과정이 [그림 13-3]의 중간에 표기한 16번의 암호화 과정이다.

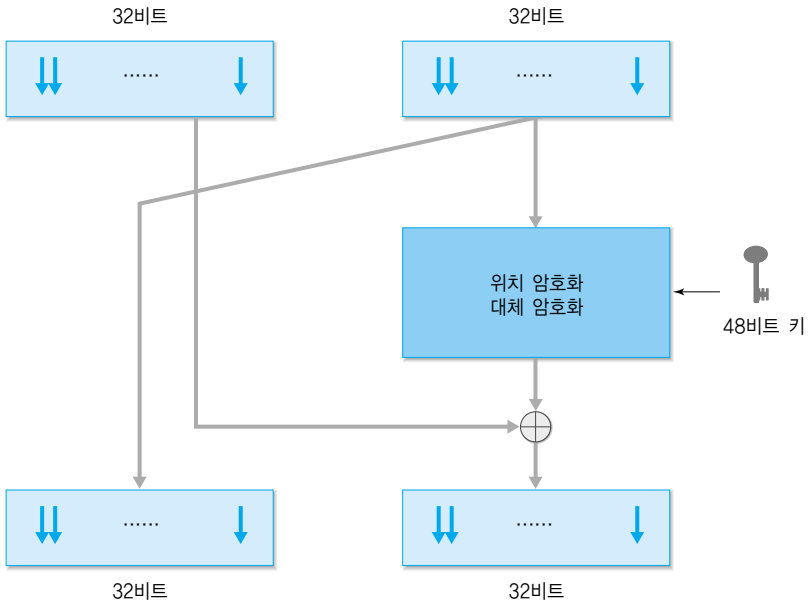


[그림 13-3] DES 알고리즘 동작 과정

DES 알고리즘의 동작 과정에서 위치 암호화는 시작과 끝에서 2번 수행된다. 즉, [그림 13-3]에서 왼쪽 과정은 총 18번의 단계로 구성되는데, 처음과 마지막에는 위치 암호화, 중간에는 대체 암호화 16번이 진행된다. 이때 위치 암호화 1과 위치 암호화 2는 서로 반대의 위치 변환을 수행한다.

16단계의 암호화

위치 암호화의 중간 단계에는 동일한 암호화 알고리즘을 16번 반복하는데, 각 단계에서 수행하는 기능은 [그림 13-4]와 같다. 먼저 이전 단계에서 주어진 입력 데이터의 오른쪽 32비트는 아무 작업 없이 출력 데이터의 왼쪽 32비트로 바로 나가 다음 단계의 입력으로 전달된다.



[그림 13-4] [그림 13-3]의 16단계 암호화 알고리즘

출력의 오른쪽 32비트 계산은 조금 복잡하다. 암호화를 위해 48비트의 암호키와 64비트 입력을 이용한다. 먼저 임의의 함수 F(오른쪽 32비트, 48비트 암호키)를 이용해 32비트의 결과를 얻는다. 이 값과 입력 데이터의 왼쪽 32비트를 배타적 논리합(Exclusive OR)하면 오른쪽 32비트 출력을 얻을 수 있다.

데이터 블록의 오른쪽 32비트는 48비트의 암호키와 연산하기 위해 48비트로 확장하는 작업을 먼저 한다. 반대로 암호키는 원래 56비트기 때문에 48비트로 축소하는 작업을 먼저 한다.

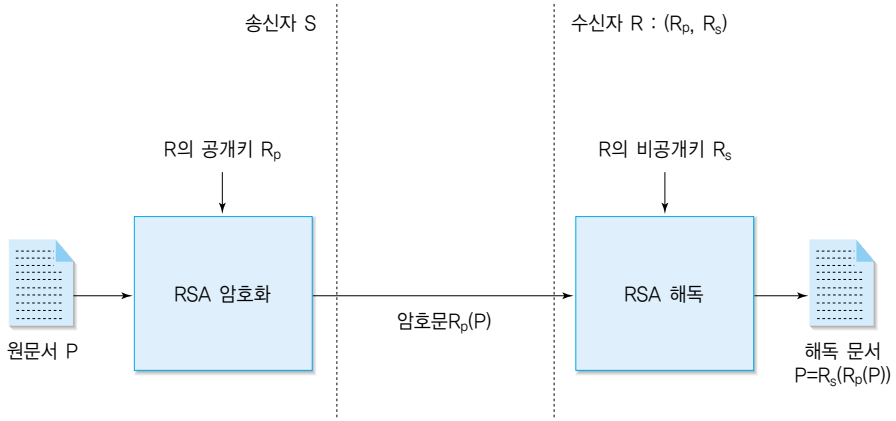
2. RSA 알고리즘

비공개키 알고리즘에서는 비밀키의 보안이 절대적으로 중요하다. 따라서 허가 받은 수신자에게만 비밀키를 분배하도록 관리해야 한다. DES 알고리즘처럼 암호문을 작성하는 암호키와 해독하는 해독키가 동일하면(혹은 하나로 다른 하나를 쉽게 유추할 수 있으면) 이 키를 분배하는 과정에서 키가 외부로 유출되지 않도록 주의해야 한다. 비밀키가 외부에 유출되면 보안을 유지하기가 사실상 불가능하다.

공개키 알고리즘은 암호화하는 키와 해독하는 키가 동일하지 않도록(혹은 하나로 다른 하나를 쉽게 유추할 수 없도록) 고안된 방식이다. 공개키 알고리즘을 이용하면, 암호문을 작성할 때 사용하는 암호키가 외부에 공개되어도 해독키를 모르면 암호문을 해독할 수 없다.

공개키 알고리즘에서는 사용자가 두 개의 암호키(공개키와 비공개키)를 사용하는데, **공개키(Public Key)**는 원문서를 암호화하는 데 사용하므로 원칙적으로 누구에게나 공개된다. 따라서 송신 호스트는 공개키로 원문서를 암호화하여 전송한다. 수신 호스트에서는 암호문을 해독하기 위해 **비공개키(Private Key)**를 사용한다. 비밀키는 공개키와 다른 값을 갖는다.

공개키 알고리즘의 대표 예는 [그림 13-5]의 RSA(알고리즘 발명자인 Rivest, Shamir, Adelman 세 사람 이름의 첫 글자) 알고리즘이다. **RSA 알고리즘**은 (공개키, 비공개키) 조합을 발생시키는 방법을 제시한다.



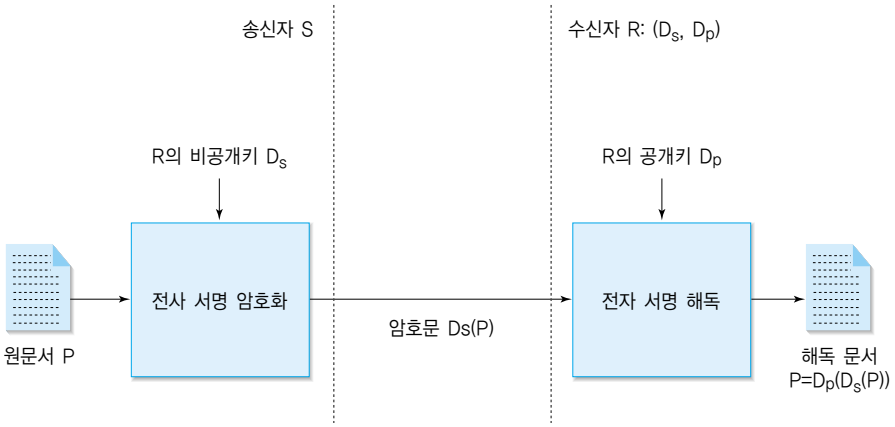
[그림 13-5] RSA 알고리즘

송신자는 공개키를 이용해 암호문을 작성하는데 작성한 암호문은 비공개키로만 해독할 수 있다. 이때 알고리즘으로 얻을 수 있는 (공개키, 비공개키) 조합은 다양한데, 서로 다른 조합에 표시된 공개키를 이용해 비공개키를 유추할 수 없다.

[그림 13-5]에서는 송신자 S가 수신자 R에 데이터를 전송할 때 수신자 R의 암호 체계인 (R_p, R_s) 조합을 사용한다. 송신자 S는 공개키 R_p로 원문서 P를 암호화하여 암호문 $R_p(P)$를 얻을 수 있으며, 이렇게 얻은 암호문을 수신자에게 전송한다. 수신자는 비공개키 R_s로 해독문 $P = R_s(R_p(P))$를 얻을 수 있다.

3. 전자 서명

전자 서명(Digital Signature)은 인터넷 환경에서 특정 사용자를 인증(Authentication)하려고 사용한다. 인증은 특정인이 진짜 그 사람인지를 확인하는 절차다. 이와 비슷한 기능으로 권한이 있고 없음을 확인하는 권한(Authorization)이 있는데, 인증과 다른 특징이 있다. [그림 13-6]은 전자 서명의 원리를 설명한다.

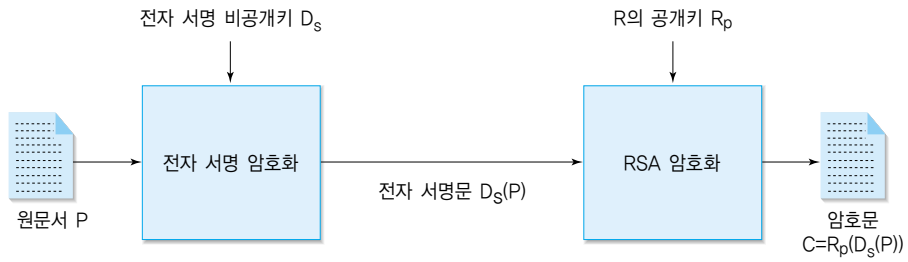


[그림 13-6] **전자 서명의 원리**

일반적으로 전자 서명의 인증 과정은 RSA 알고리즘과는 반대 원리며 비공개키 알고리즘과 공개키 알고리즘의 조합을 사용한다. 전자 서명은 자신을 다수의 타인에게 증명하는 기능이므로, 암호화 과정에서 자신만 아는 비밀키(전자 서명)를 사용한다. 암호화한 전자 서명은 다수의 타인이 확인하므로 해독 과정에서는 공개키를 사용한다.

암호화 과정

일반적으로 전자 서명의 암호화는 [그림 13-7]처럼 두 단계로 이루어진다. 첫 번째 단계는 전자 서명 알고리즘으로 자신을 인증하는 암호화고, 두 번째 단계는 전자 서명의 정보를 전송하기 위해 다시 RSA 알고리즘을 사용하는 암호화다.

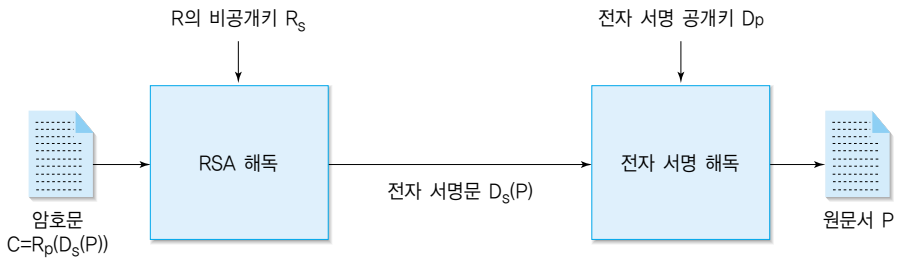


[그림 13-7] **전자 서명 암호화**

[그림 13-7]처럼 송신자는 데이터 전송 과정에서 두 번의 암호화를 수행한다. 먼저 비공개키 D_s로 표현되는 전자 서명을 사용해 원문서 P를 암호화함으로써, 전자 서명문 $D_s(P)$를 얻는다. 이 문서를 다시 공개키 알고리즘인 RSA의 공개키 R_p를 사용해 암호문 $R_p(D_s(P))$로 만들고, 이를 수신자에게 전송한다.

해독 과정

[그림 13-8]은 수신자가 전자 서명된 문서를 해독하는 과정이다. 먼저 수신한 문서를 RSA 알고리즘의 비공개키 R_s로 해독함으로써 전자서명문 $D_s(P)$를 얻는다. 그리고 이 문서를 송신자의 인증에 필요한 전자 서명 공개키인 D_p를 사용해 해독하여 원문서 P를 얻는다.



[그림 13-8] **전자 서명 해독**

전자 서명 과정에서 복잡하게 두 단계로 암호화하는 이유는 다음과 같다. 먼저 RSA 알고리즘을 사용해 암호화하는 과정은 전송 과정에서의 보안 문제를 해결하기 위함이다. 그런데 이렇게 전송 보안 문제를 해결하면 전자 서명의 기본 목적인 인증 문제를 해결해야 하므로 비공개키인 전자 서명을 사용해 암호화하는 과정도 필요하다.

Section

03 보안 프로토콜

인터넷이 일반화됨에 따라 네트워크 보안 문제는 중요 이슈가 되었다. 네트워크의 기본 개념은 서로 연동하여 자원을 공유하는 것이므로 불특정 다수가 네트워크 시스템에 접근함을 전제로 한다. 이런 구조에서 사용자의 접근을 제어하는 일은 간단한 문제가 아니다. 이 절에서는 보안과 관련된 다양한 프로토콜을 살펴본다.

1 보안 프로토콜의 개요

인터넷은 전 세계적으로 연결된 거대한 통신망으로, 송신자가 전송한 데이터가 수신자에게 전달되는 과정에서 여러 호스트와 매체를 통한다. 이 과정에서 여러 가지 보안 문제에 직면할 수 있다. 특히 중간에 위치한 호스트의 보안 등급이 낮게 설정된 경우에는 위험에 노출될 가능성이 더 높다.

이러한 위협 요소에는 전송 데이터를 중간에서 감청하거나 임의로 변경하는 경우, 원격 호스트의 데이터에 위해를 가하는 등 직접적으로 특정 시스템의 내부에 침입하는 경우, 과도한 트래픽을 발생시켜 특정 호스트의 통신을 방해하는 경우 등이 있다.

감청

감청은 허가받지 않은 자가 직간접적인 방법으로 전송 중인 데이터를 얻어내는 것이다. 또한 얻어낸 정보를 변경한 후 이를 통신 과정에 다시 입력함으로써 송수신 호스트의 통신 내용을 왜곡하는 것도 넓은 의미에서 감청에 포함된다.

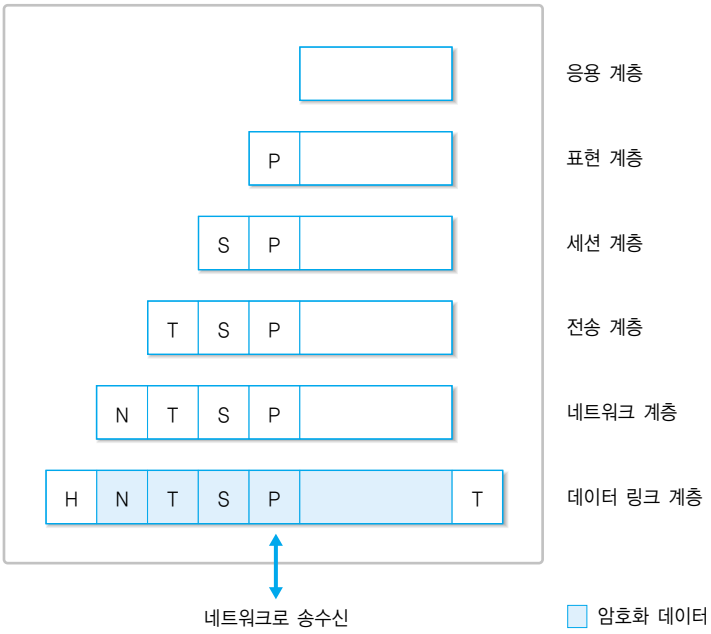
전통적으로 감청의 가장 일반 형태는 유선의 통신 선로에서 이루어지는데, 인터넷에서는 이더넷 선로에 감청 장비를 연결해 패킷을 훔쳐낼 수 있다. 예를 들어, 감청하려는 호스트의 네트워크 보드와 MAC 주소가 같은 장비를 연결하여 전달되는 패킷을 전부 감청할 수 있다.

휴대폰의 무선 데이터는 무선 신호가 넓은 범위로 전파되기 때문에 물리적인 감청이 훨씬 용이하다. 따라서 유선에서도 보안이 중요하지만 무선 신호의 송수신에서는 암호화 과정을 통해 보안을 유지하는 작업이 꼭 필요하다.

암호화

전송 선로에서의 감청 위협으로부터 데이터를 안전하게 보호하는 방법에는 물리 계층에서 데이터를 송신하기 전에 암호화하는 **데이터 링크 계층 암호화**가 있다.

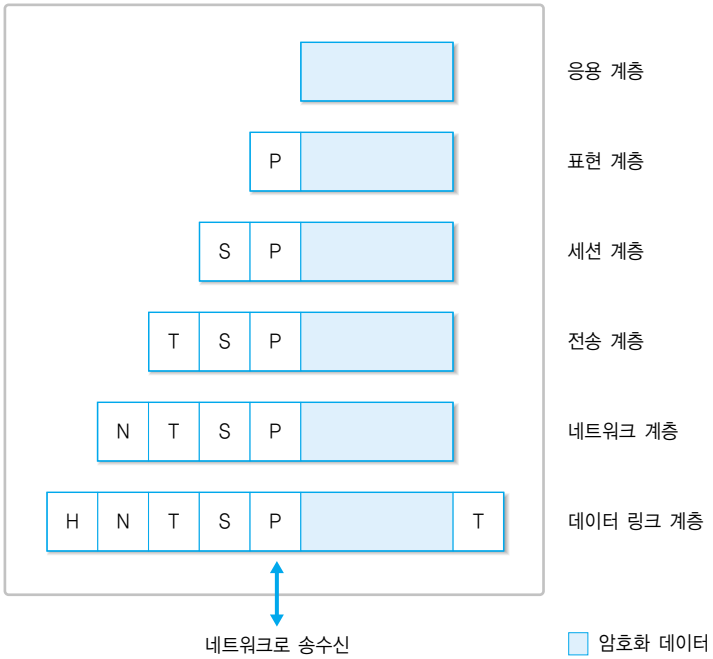
[그림 13-9]처럼 응용 계층부터 네트워크 계층까지 정보는 암호화되어 있지 않은데, 전송 직전인 데이터 링크 계층에서 암호화하여 전송하는 방식이 데이터 링크 계층 암호화다. 수신 호스트에서는 이와 반대로 암호화된 데이터를 수신하기 때문에 데이터 링크 계층에서 해독하여 상위 계층으로 전달해야 한다. 이 방식은 호스트 사이에 있는 전송 선로에서의 감청 위협으로부터 데이터를 보호한다.



[그림 13-9] 데이터 링크 계층 암호화

데이터 링크 계층 암호화의 단점은 네트워크 계층 데이터가 암호화되지 않기 때문에 송수신자 사이의 라우터에서는 보안이 되지 않는다는 것이다. 즉, 라우터를 포함한 전송 호스트 내부에서는 보안을 지원하지 않고, 호스트와 호스트 사이의 전송 과정에서만 보안이 유지된다.

호스트 내부에서 보안을 지원하려면 **응용 계층 암호화** 방식을 사용해야 한다. 이 방식은 송수신 과정의 끝단에 위치한 응용 계층에서 암호화하는 방식으로 형태는 [그림 13-10]과 같다.



[그림 13-10] **응용 계층 암호화**

암호문의 작성이 송신 호스트의 응용 계층에서 이루어졌기 때문에 라우팅을 포함하여 전송의 모든 과정에서 보안을 유지할 수 있다.

트래픽 제어

특정 호스트가 누구와 통신을 많이 하는지에 대한 정보도 네트워크 보안에 포함된다. 예를 들어, 전쟁 중에 특정 부대와의 교신량이 평소보다 증가하면 그 지역에서 모종의 군사 작전이 있을 가능성이 높다. 일반 사회에서도 특정 회사끼리 접촉이 잦으면 모종의 상업적인 협력이 늘고 있음을 암시한다.

외부 침입자의 통신량 분석을 방해하는 간단한 방법은 무의미한 가공 데이터를 여러 호스트에서 주기적으로 발생시킴으로써 통계 자료에 혼선을 주는 것이다. 이러한 자료의 통신량, 송신자, 수신자 등은 랜덤하게 생성된다.

2. 방화벽

인터넷의 확산으로 허가 받지 않은 사용자의 불법적인 사설 망 접근을 방지하는 문제가 중요한 이슈로 인식되고 있다. 따라서 개방적인 공중 인터넷망과 제한된 사용자 그룹에게 허가된 사설 망 사이에 보안 기능이 필요하며, 이를 **방화벽(Firewall)**이라고 한다.

사설 망을 외부로부터 보호하는 가장 간단한 방법은 외부 망을 완전히 끊어 버리는 것이다. 그러나 사설 망의 내부 사용자가 공중 인터넷 망에 접속하면서 보안을 유지하려면 [그림 13-11]과 같은 방화벽의 기능이 필수다.

[그림 13-11] **방화벽**

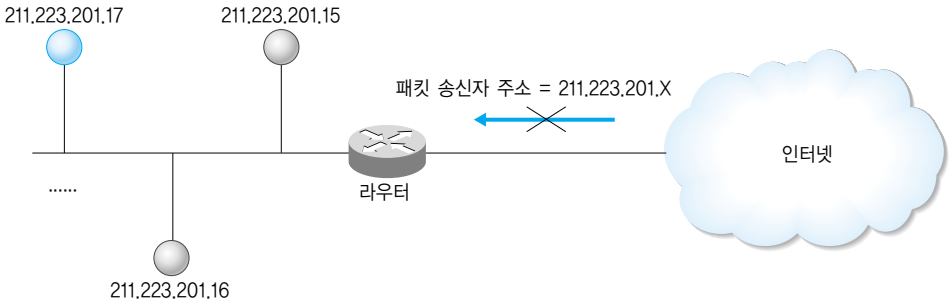
사설 망을 공중 인터넷 망으로부터 보호하는 기술은 크게 두 가지다. 첫째는 패킷 필터링 방법이다. 이 방식에서는 방화벽이 패킷의 헤더를 검색하거나, 필요에 따라서는 내용까지 검색하여 적절하지 못한 패킷을 배제시킨다. 이러한 기능의 구현은 보통 라우터에서 이루어진다. 방화벽의 또 다른 기능은 해커 같이 의심스러운 행위를 하는 사용자를 감시하는 것이다.

라우터를 이용한 방화벽 구현

인터넷에 연결된 모든 호스트들은 외부 통신망과 연결하기 위해서 반드시 라우터의 중개 과정을 거쳐야 한다. 따라서 라우터를 이용한 방화벽 구현은 간단하면서도 매우 효과적이다.

라우터는 자신과 연결된 네트워크로부터 입력된 패킷의 정보를 분석하여, 어느 네트워크로 중개할 것인지를 결정하므로 패킷을 계속 전송할지에 대한 권한도 가지고 있다. 예를 들어,

[그림 13-12]와 같이 외부 네트워크로부터 내부의 호스트 IP 주소를 위장한 송신 호스트가 전송하는 데이터가 내부 네트워크로 전달되는 것을 차단할 수 있다.



[그림 13-12] 위장 IP 주소의 차단

[그림 13-12]에서 211.223.201.×는 라우터 왼쪽에 위치한 내부 네트워크에서 사용하는 주소지만 주소 발신자가 라우터의 오른쪽에 위치한다. 이러한 패킷은 라우터가 차단한다.

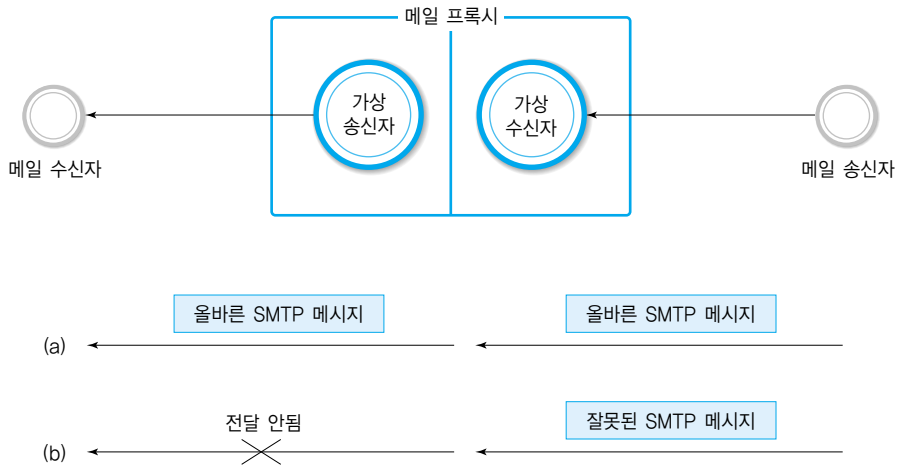
라우터의 차단 기능은 다양하게 사용할 수 있는데, 외부의 특정 호스트가 스팸 메일을 자주 보낼 때는 이 호스트를 발신자로 하는 모든 패킷을 차단할 수 있다. 반대로 내부 사용자가 불법 도박 사이트나 유해한 게임 사이트에 접근하는 것도 차단할 수 있다.

호스트의 IP 주소뿐만 아니라, 포트 번호를 이용한 응용 프로그램의 접근도 차단할 수 있다. FTP, 텔넷, 전자 메일 프로그램은 특정 포트 번호를 사용하기 때문에 이 포트 번호를 송신 주소나 목적지 주소로 갖는 패킷을 차단할 수 있다. 따라서 내부에서 외부로, 혹은 외부에서 내부로 특정 응용 서비스에 접근하는 것을 제어할 수 있다.

프록시를 이용한 방화벽 구현

라우터의 방화벽 기능은 네트워크 계층과 전송 계층의 헤더 정보에 기초하여 이루어진다. 그러므로 메일 내용처럼 패킷 내부에 위치한 응용 데이터는 제어할 수 없다.

프록시(Proxy)는 응용 환경에서 적절하게 처리할 수 있는 정보만 수신하도록 가상의 응용 프로그램을 시뮬레이션하는 방화벽이다. 프록시는 내부 네트워크의 호스트에는 외부 네트워크의 응용 연결처럼 보이고, 외부 네트워크에는 내부 네트워크의 응용 연결처럼 보인다.



[그림 13-13] **메일 프록시**

예를 들어, [그림 13-13]처럼 메일 송신자와 수신자 사이에 메일 프록시를 설치할 수 있다. 메일 수신자는 메일 프록시 내부에 있는 가상 송신자를 실제 메일 송신자로 인식하고, 메일 송신자는 메일 프록시 내부에 있는 가상 수신자를 메일 수신자로 인식하여 SMTP 메시지를 전송한다.

중간의 프록시는 메일 시스템이라는 응용 서비스 기능을 구현하기 때문에 메일 송신자가 전송한 SMTP 메시지를 해석할 수 있다. 따라서 잘못된 메시지는 프록시가 차단하므로 메일 수신자가 올바르게 처리할 수 있는 메시지만 전달된다.

웹 기능이 구현된 웹 프록시의 경우를 가정하면 내부 네트워크 사용자가 어떤 웹 서버를 어느 정도 방문하는 지에 대한 통계 등을 관리할 수 있다. 따라서 자주 방문하는 사이트 정보는 프록시에 저장하여 사용자에게 정보를 더 빠르게 제공할 수 있다.

요약

1 데이터 전송 과정에서 외부 침입자로부터 데이터를 보호하려면 암호화하여 전송해야 한다. 그리고 수신자는 암호화된 데이터를 해독할 수 있어야 한다.

2 임의의 문자를 다른 문자로 대체하는 대체 암호화 방식에는 시저 암호화, 키워드 암호화, 복수 개의 문자표 등이 있다.

3 문자의 위치를 변경하는 위치 암호화 방식에는 컬럼 암호화, 키워드 암호화 등이 있다.

4 DES 알고리즘은 비공개키를 이용한 암호화 방식으로 송수신자가 동일한 비밀키를 사용한다.

5 RSA 알고리즘은 공개키를 이용해 암호문을 작성하고 비공개키를 이용해 암호문을 해독한다.

6 전자 서명은 비공개키를 이용해 암호문을 작성하고 공개키를 이용해 암호문을 해독한다.

7 전송 선로의 감청으로부터 데이터를 보호하려면 데이터 링크 계층 암호화를 사용한다.

8 전송 경로에 있는 호스트 내부에서 데이터를 보호하려면 응용 계층 암호화를 사용한다.

9 방화벽 기능을 이용해 패킷 정보를 해석하여 내부 네트워크를 외부 네트워크의 불법 침입으로부터 보호할 수 있다.

10 프록시 기능을 이용해 응용 계층의 내용을 해석하여 내부 네트워크를 외부 네트워크의 불법 침입으로부터 보호할 수 있다.

연습문제

1 암호화 과정의 필요성을 설명하시오.

2 시저 암호화의 원리와 장단점을 설명하시오.

3 대체 암호화에서 키워드 알고리즘의 원리와 장단점을 설명하시오.

4 컬럼 암호화의 원리와 장단점을 설명하시오.

5 위치 암호화 방식에서 키워드 알고리즘의 원리와 장단점을 설명하시오.

6 DES 알고리즘을 설명하시오.

7 RSA 알고리즘을 설명하시오.

8 전자 서명을 설명하시오.

9 데이터 링크 계층 암호화를 설명하시오.

10 응용 계층 암호화를 설명하시오.

11 방화벽의 필요성을 설명하시오.

12 네트워크 보안과 관련해 라우터에서 수행하는 기능을 설명하시오.

13 네트워크 보완과 관련해 프록시 기능을 설명하시오.