

# 핵심요약노트

- PART 01 시스템 보안
- PART 02 네트워크 보안
- PART 03 애플리케이션 보안
- PART 04 정보보안 일반
- PART 05 정보보안 관리 및 법규

## 01

## 시스템 보안

## 1 기억장치의 구성요소

코드 영역 (Code Space)	<ul style="list-style-type: none"> <li>중앙처리장치에 의하여 실행되는 시스템 코드가 있는 영역으로, EIP(Extended Instruction Pointer)가 다음에 실행할 명령을 가리킴</li> <li>프로그램 코드 자체를 구성하는 명령이나 기계어 명령을 위한 메모리 영역</li> </ul>
데이터 영역 (Data Space)	<ul style="list-style-type: none"> <li>초기화된 데이터 세그먼트(Initialized Data Segment)라고도 하며, 프로그램이 실행될 때 0이나 NULL 포인터로 초기화되는 영역</li> <li>프로그램의 전역변수나 정적변수의 할당을 위하여 존재하는 영역</li> </ul>
힙 영역 (Heap Space)	<ul style="list-style-type: none"> <li>프로그램이 실행될 때까지 가변적인 양의 데이터를 저장하기 위해 프로세스가 사용할 수 있도록 예약되어 있는 메모리 영역</li> <li>프로그램에서 사용하는 일시적이고 동적인 메모리를 할당하기 위하여 사용되는 메모리 영역</li> </ul>
스택 영역 (Stack Space)	<ul style="list-style-type: none"> <li>후입선출(LIFO) 방식에 의하여 데이터를 관리하는 구조</li> <li>Top이라고 불리는 스택의 끝부분에서 데이터의 삽입과 삭제가 일어남</li> <li>지역변수, 매개변수, 복귀변지, 함수호출 시 전달되는 인수값 저장을 위한 메모리 영역</li> </ul>

## 2 신뢰 플랫폼 모듈(TPM)

개념	<ul style="list-style-type: none"> <li>신뢰컴퓨팅을 구축하기 위해서 필요한 여러 하위기능을 제공하는 모듈</li> <li>주로 암호화 키를 포함하여 기본 보안 관련 기능을 제공하도록 디자인된 마이크로칩이며, 일반적으로 컴퓨터의 메인보드에 설치되어 있으며, 하드웨어 버스를 사용하여 시스템의 나머지 부분과 통신</li> </ul>
기능	<ul style="list-style-type: none"> <li>암호화 키의 생성과 저장</li> <li>무결성 검증을 위한 측정값의 저장</li> <li>디지털 인증서 관련 신뢰 연산의 제공</li> </ul>

## 3 보안 운영체제 보호기법

물리적 분리	서로 다른 보안 수준을 요구하는 프로세스별로 분리
시간적 분리	서로 다른 보안 수준을 요구하는 프로세스를 서로 다른 시간에 운영
논리적 분리	프로그램의 접근을 제한하여 허용된 영역 밖의 객체에 대해 접근제한
암호적 분리	다른 프로세스가 인식할 수 없는 방법으로 자신의 데이터를 숨김

#### 4 Windows 공유 폴더

ADMIN\$	모든 Windows를 대상으로 어떤 파일을 복사하거나 변경할 필요가 있을 때 주로 사용
C\$, D\$	<ul style="list-style-type: none"> <li>• C 또는 D 드라이브에 대한 공유 폴더로, 하드디스크 수만큼 공유되어 있음</li> <li>• 드라이브 안에 있는 파일/폴더의 삭제나 생성이 쉬우므로, 임의로 파일을 복사하거나 중요한 파일을 삭제할 수도 있음</li> </ul>
IPC\$	<ul style="list-style-type: none"> <li>• 컴퓨터 간 필요한 정보를 통신하기 위하여 사용하며, 네트워크에서 프로세스 간 통신을 위하여 사용</li> <li>• Null Session Share에 취약하여 레지스트리 수정을 통한 변경 필요</li> </ul>

#### 5 Windows 백업

일반 백업	<ul style="list-style-type: none"> <li>• 일반적으로 전체 백업을 할 때 사용</li> <li>• 아카이브 비트(Archive Bit)와 관계없이 모든 파일을 백업 <ul style="list-style-type: none"> <li>- 백업 후 모든 파일의 아카이브 비트를 리셋함</li> </ul> </li> <li>• 각 파일의 백업 상태를 표시함</li> </ul>
복사본 백업	<ul style="list-style-type: none"> <li>• 일반 백업과 같지만, 백업 후 아카이브 비트를 리셋하지 않음</li> <li>• 각 파일의 백업 상태를 표시하지 않음</li> </ul>
증분 백업 (Incremental Backup)	<ul style="list-style-type: none"> <li>• 전체 백업 또는 증분 백업이 수행된 후 변경사항을 선택적으로 백업하는 방식</li> <li>• 마지막 일반 백업 또는 증분 백업 이후 생성되거나 변경된 파일만 복사하는 백업</li> <li>• 아카이브 비트가 셋팅된 파일만 백업, 백업 후 아카이브 비트를 리셋함</li> <li>• 각 파일의 백업 상태를 표시함</li> <li>• 장점 : 전체 백업보다 백업 데이터양이 적고, 백업 소요시간 짧음</li> <li>• 단점 : 전체 백업에 종속적이고, 증분 백업이 많을수록 복구시간 많이 소요</li> </ul>
차등 백업 (Differential Backup)	<ul style="list-style-type: none"> <li>• 이미 수행된 차등 백업과 무관하게 전체 백업 이후의 모든 변경사항을 백업하는 방식</li> <li>• 일반 백업이나 증분 백업을 마지막으로 수행한 이후 생성되거나 변경된 파일을 복사하는 백업</li> <li>• 증분 백업과 같지만, 백업 후 아카이브 비트를 리셋하지 않음</li> <li>• 각 파일의 백업 상태를 표시하지 않음</li> <li>• 장점 : 복구 시 전체 백업본과 차등 백업본 각각 하나씩만을 필요로 하므로, 빠른 시간 내에 복구 가능</li> <li>• 단점 : 증분 백업보다 백업 데이터양이 많음</li> </ul>

## 6 바이러스 세대별 분류

1세대	원시형	<ul style="list-style-type: none"> <li>• 프로그램 구조가 단순하고 분석이 상대적으로 쉬운 바이러스</li> <li>• 기존의 DOS용 바이러스 대부분이 이에 해당</li> <li>• 종류 : 돌(Stoned) 바이러스, 예루살렘(Jerusalem) 바이러스 등</li> </ul>
2세대	암호형	<ul style="list-style-type: none"> <li>• 백신 프로그램이 진단할 수 없도록 바이러스 프로그램의 일부 또는 대부분을 암호화시켜 저장</li> <li>• 종류 : 폭포(Cascade) 바이러스, 느림보(Slow) 바이러스 등</li> </ul>
3세대	은폐형	<ul style="list-style-type: none"> <li>• 자신을 은폐하고 사용자나 백신 프로그램에 거짓 정보를 제공하기 위한 다양한 기법 사용</li> <li>• 기억장소에 존재하면서 감염된 파일의 길이가 증가하지 않은 것처럼 보이게 함</li> <li>• 백신 프로그램이 감염된 부분을 읽으려고 할 때 감염되기 전의 내용을 보여줌으로써, 바이러스가 없는 것처럼 백신 프로그램이나 사용자를 속임</li> <li>• 종류 : 조시(Joshi) 바이러스, 프로도(Frodo) 바이러스 등</li> </ul>
4세대	감염형	<ul style="list-style-type: none"> <li>• 초기 2세대, 3세대 바이러스는 백신 프로그램이 바이러스를 진단하기 어렵게 하는 것이 목표였으나, 백신 프로그램의 발달로 목표를 이뤄내기가 불가능하자 바이러스 개발자는 백신 프로그램 개발자를 공격 대상으로 백신 프로그램으로부터 숨기보다는 여러 단계의 암호화와 다양한 기법을 동원하여 바이러스 분석을 어렵게 하고 백신 프로그램 개발을 지연시킴</li> <li>• 종류 : 다형성(Polymorphic) 바이러스 등</li> </ul>
5세대	매크로형	<ul style="list-style-type: none"> <li>• 운영체제와 관계없이 응용 프로그램 내부에서 동작하는 것이 가장 큰 특징</li> <li>• 매크로 기능이 있는 마이크로소프트(Microsoft)사 오피스 제품군(워드, 엑셀, 파워 포인트) 이외에 비지오(Visio), 오토캐드(AutoCAD)</li> <li>• VBScript(VBScript)를 지원하는 다양한 프로그램에서 활동하기 때문에 현재 등장하고 있는 바이러스 중에서 가장 높은 비중 차지</li> </ul>

## 7 악성 프로그램의 종류 및 특징

일반 바이러스	<ul style="list-style-type: none"> <li>• 자신 또는 자신의 변형 코드를 실행하는 프로그램</li> <li>• 실행 가능한 시스템 영역 등에 복제하는 프로그램</li> <li>• 기생할 숙주가 반드시 필요함</li> </ul>
웜	네트워크 및 메일을 통하여 자신을 복제하고 유포시키는 프로그램
트로이목마	<ul style="list-style-type: none"> <li>• 자기복제 능력이 없으며, 유틸리티 프로그램 내에 악의적 기능을 가지는 코드를 내장하여 배포하거나, 그 자체를 유틸리티 프로그램으로 위장하여 유포</li> <li>• 특정한 환경 또는 조건에서 배포자의 의도에 따라 사용자의 정보 유출이나 자료를 파괴</li> </ul>
스파이웨어	다른 사용자의 컴퓨터에 잠입하여 중요한 개인정보를 유출하는 프로그램
논리폭탄	조건이 충족되면 트리거가 작동하여 실행되는 프로그램
백도어(트랩도어)	허가받지 않고 접근할 수 있는 비밀 통로
익스플로잇	하나 혹은 여러 개의 취약점을 공격하는 코드
플러더(Flooder)	서비스거부 공격을 하기 위해 사용하는 코드
키로거	사용자의 키 입력 정보를 탈취하여 공격자에게 전송하는 프로그램
루트킷	시스템에 침투하여 루트 권한을 획득하기 위한 공격 도구 모음
좀비(Zombie)	악성 코드에 감염된 컴퓨터로, 다른 컴퓨터를 공격하기 위해 사용
브라우저 하이재커	브라우저를 하이재킹하여 홈페이지, 검색 페이지, 톨바 등을 통제하고 조작하는 프로그램

조크(Joke)	사용자에게 심리적인 위협이나 불안감을 조장하는 프로그램
혹스(Hoax)	사용자를 속이거나 장난을 목적으로 전송하는 가짜 바이러스
스파이웨어	컴퓨터에서 정보를 수집하여 다른 컴퓨터에 전송하는 프로그램
애드웨어	일반 프로그램에 내장된 광고, 일정한 규정보다 지키면 적법함

## 8 악성 프로그램의 비교

	바이러스	웜	트로이목마
유형	파일이나 부트 섹터에 감염	독립적으로 존재	정상적인 프로그램으로 위장하거나 코드 형태로 삽입
자기복제 능력	거의 없음	매우 강함	없음
전파 경로	감염된 파일 복사	네트워크를 통해 전파	파일 다운로드 시 포함
증상	파일 손상	네트워크 성능 저하	컴퓨터 성능 저하

## 9 Windows 레지스트리의 구성 정보

HKEY_CLASSES_ROOT	<ul style="list-style-type: none"> <li>OLE 데이터와 파일의 각 확장자에 대한 정보 저장</li> <li>파일과 프로그램 간 연결에 대한 정보 포함</li> </ul>
HKEY_CURRENT_USER	<ul style="list-style-type: none"> <li>Windows가 설치된 컴퓨터 환경설정에 대한 정보 포함</li> <li>다수의 사용자가 사용할 경우 각 사용자별 프로파일 저장</li> <li>현재 로그인하여 시스템을 사용 중인 사용자의 배경 화면, 디스플레이 설정이나 단축 아이콘, 사용자가 설치한 응용 프로그램의 설정 등의 정보가 기록되어 있음</li> </ul>
HKEY_LOCAL_MACHINE	<ul style="list-style-type: none"> <li>Windows에 설치된 모든 하드웨어와 소프트웨어의 설정정보 포함</li> <li>Windows를 처음 설치할 때 내용이 구성되며, 장치관리자를 이용하여 하드웨어 구성을 변경할 수 있음</li> <li>현재 설치된 하드웨어와 사용 중인 드라이버에 대한 정보부터 프린터, 인터넷 시리얼 포트 설정 등을 저장</li> </ul>
HKEY_USERS	<ul style="list-style-type: none"> <li>컴퓨터에서 사용 중인 각 사용자 프로파일에 대한 HKEY_CURRENT_USER 키에 일치하는 서브키를 저장</li> <li>데스크톱 설정과 네트워크 환경에 대한 정보 포함</li> <li>user.dat에 저장</li> </ul>
HKEY_CURRENT_CONFIG	<ul style="list-style-type: none"> <li>실행시간에 수집한 자료를 저장하고 있으며, 이 키에 저장된 정보는 디스크에 영구적으로 저장되지 않고 시동 시간에만 생성</li> <li>디스플레이와 프린터에 관한 정보 포함</li> <li>레지스트리 중 가장 단순한 키</li> </ul>

## 10 웹브라우저 보안영역

인터넷	<ul style="list-style-type: none"> <li>• 다른 영역에 포함되지 않은 모든 웹 사이트의 보안수준 설정</li> <li>• 기타 범주에 들어가지 않는 모든 사이트가 해당</li> </ul>
로컬 인트라넷	<ul style="list-style-type: none"> <li>• 사용자의 인트라넷에 있는 모든 웹 사이트의 보안수준 설정</li> <li>• 방화벽 뒤에 있는 또는 조직 내의 사이트를 말하며, 이 사이트는 보통 가장 높은 수준의 신뢰를 가짐</li> </ul>
신뢰할 수 있는 사이트	<ul style="list-style-type: none"> <li>• 사용자 컴퓨터나 데이터를 손상시키지 않을 것으로 신뢰되는 웹 사이트의 보안수준 설정</li> <li>• 방화벽 밖에 있지만 가장 높은 신뢰의 사이트</li> </ul>
제한된 사이트	<ul style="list-style-type: none"> <li>• 사용자 컴퓨터나 데이터를 손상시킬 수도 있는 웹 사이트의 보안수준 설정</li> <li>• Windows를 처음 설치했을 때 비어 있음</li> </ul>

## 11 패스워드 크래킹 도구

John The Ripper	<ul style="list-style-type: none"> <li>• 가장 잘 알려진 패스워드 점검 도구</li> <li>• Windows, 리눅스, MAC 등 거의 모든 운영체제에서 사용 가능</li> </ul>
Pwdump	Windows에서 패스워드를 덤프할 수 있는 도구
LophCrack	<ul style="list-style-type: none"> <li>• 패스워드 취약성 점검 도구</li> <li>• 원격 또는 로컬 서버나 컴퓨터의 패스워드를 점검하는 데 유용</li> <li>• SAM 파일의 해시 정보를 이용하거나 무차별 대입 공격을 통한 패스워드 취약점 점검</li> </ul>
lpcCrack	<ul style="list-style-type: none"> <li>• 사용자 계정 및 패스워드를 원격지에서 추측하여 취약성을 점검하는 도구</li> <li>• 사전 공격을 이용한 점검</li> </ul>
Chntpw	물리적 접근이 가능한 시스템에서 패스워드를 초기화시키는 프로그램
ERD Commander	Windows 20XX 서버의 관리자 패스워드 분실 시 복구를 위해 사용

## 12 패스워드 공격기법

무차별 대입 공격 (Brute Force Attack)	암호를 찾기 위하여 가능한 모든 조합을 시도하는 공격기법
사전 공격 (Dictionary Attack)	사전 파일(단어 리스트 파일)을 이용하여 공격하는 기법
암호 추측 공격 (Password Guessing)	사용자의 개인정보를 이용하여 암호를 추측하고 대입하는 공격 기법

### 13 Windows 인증 프로세스의 역할

Winlogon	Windows 로그인 프로세스의 한 부분
GINA (Graphical Identification and Authentication)	Winlogon 내에서 msGINA.dll을 로딩시켜 사용자가 입력한 계정과 패스워드를 LSA에게 전달
LSA (Local Security Authority)	<ul style="list-style-type: none"> <li>전달받은 계정과 패스워드를 검증하기 위해 NTLM 모듈을 로딩하고 모든 계정의 로그인에 대한 검증</li> <li>시스템 자원 및 파일 등에 대한 접근 권한을 검사(로컬, 원격 모두에 해당)</li> <li>SRM이 작성한 감사 로그를 기록하는 역할 수행</li> <li>NT 보안의 중심 요소이며, 보안 서브 시스템(Security Subsystem)이라고도 함</li> </ul>
SAM (Security Account Manager)	<ul style="list-style-type: none"> <li>사용자, 그룹 계정 정보(암호화된 해시값)에 대한 데이터베이스를 관리(사용자 계정 정보 저장)</li> <li>사용자의 로그인 입력 정보와 SAM 데이터베이스 정보를 비교하여 인증 여부를 결정</li> <li>리눅스의 /etc/shadow와 같은 역할, 중요하므로 복사본 존재</li> <li>SAM 위치 : %systemroot%/system32/config/sam</li> </ul>
SRM (Security Reference Monitor)	<ul style="list-style-type: none"> <li>SAM이 사용자의 계정과 패스워드가 일치하는지를 확인하여 SRM(Security Reference Monitor)에게 알려주면, 사용자에게 고유의 SID(Security Identifier)를 부여</li> <li>SID에 기반하여 파일이나 디렉터리에 접근제어를 하게 되고, 이에 대한 감사 메시지를 생성</li> <li>사용자에게 고유의 SID를 부여</li> <li>SID에 준하는 권한을 부여</li> </ul>

### 14 Windows SID(Security Identifier) 구조

The SID for account NEWGENERATION \administrator is

S-1-5-21-1801674531-839522115-1708537768-500

(a)

(b)

(c)

(d)

(a)	S-1	Windows 시스템을 의미
(b)	5-21	시스템이 도메인 컨트롤러이거나 단독 시스템(Stand Alone System)임을 표시
(c)	D1-D2-D3	<ul style="list-style-type: none"> <li>시스템의 고유한 숫자</li> <li>- 이 고유한 숫자는 시스템을 설치할 때 시스템의 특성을 수집하여 생성</li> <li>도메인에 가입하면 D1, D2, D3 값이 관리자와 같게 되는데, 그 값을 통해 가입 여부를 확인</li> </ul>
(d)	RID (Relative Identifier)	<ul style="list-style-type: none"> <li>각 사용자별 숫자로 표현되는 고유한 ID(상대 식별자)</li> <li>관리자(Administrator)는 500번, Guest 계정은 501번, 일반 사용자는 1000번 이상의 숫자를 가짐</li> </ul>



## 15 Unix/Linux /etc/passwd 파일 구조

root	x	0	0	root	/root	/bin/bash
계정명	패스워드	UID	GID	계정 설명	홈 디렉터리 경로	셸 경로

  

UID, GID	0	root
	500 이상	일반 사용자(일반 계정 추가 시 500번부터 시작)
	500 이하	시스템 예약

## 16 Unix/Linux /etc/shadow 파일 구조

root	!@~3Ux1	16551	0	999	7	::
계정명	암호화된 패스워드	패스워드 변경기간	패스워드 변경 최소일수	패스워드 최대 유효기간	패스워드 만료 경고일	기타

- /etc/shadow 파일의 두 번째 필드는 사용자의 패스워드가 암호화되어 저장
- 형식 : [\$Hash 종류][\$salt][\$Hash 결과]
  - [root:\$6\$bWUzfiz54MQFmZ3i\$OZ4zTWsrCy3uKrGpQQf4MH4KIZZ9enc77PS,p...]

root	사용자 계정명
\$6	Hash 종류 - \$1 : MD5 - \$2 : Blowfish - \$5 : SHA-256 - \$6 : SHA-512
\$bWUzfiz54MQFmZ3i	• Salt • 동일한 패스워드가 서로 같은 해시 값을 갖지 않도록 사용하는 랜덤 값
\$OZ4zTWsrCy3uKrGpQQf4MH4KIZZ9enc77PS,p.....	패스워드 + Salt 값을 해싱한 결과 값

## 17 허가권(Permission)

파일 형식	소유자 허가권	그룹 허가권	제3자 허가권	소유자명	그룹명
- , d, b, c, l, s	rwX	rwX	rwX	root	root
	421	421	421		



## 18 Sticky Bit, SetGID, SetUID

특수 비트	허가권	설명
Sticky Bit	1000	<ul style="list-style-type: none"> <li>• /tmp 디렉터리가 대표적</li> <li>• Other 권한의 사용자는 /tmp 디렉터리 안에서 파일에 대한 모든 권한을 사용 가능</li> <li>• /tmp 디렉터리에 대한 삭제 권한은 root와 소유자만 가진</li> <li>• 속성 표시 및 허가권 : <code>rw-rw-r-t</code> → 1765</li> </ul>
SetGID	2000	<ul style="list-style-type: none"> <li>• 파일이 실행될 때만 실행한 그룹에게 소유자 권한을 부여하는 허가권</li> <li>• 속성 표시 및 허가권 : <code>rw-r-sr-x</code> → 2655</li> </ul>
SetUID	4000	<ul style="list-style-type: none"> <li>• 파일이 실행될 때만 실행한 사용자에게 소유자 권한을 부여하는 허가권</li> <li>• 속성 표시 및 허가권 : <code>rwsrw-r--</code> → 4764</li> </ul>

## 19 FAT와 NTFS 파일 시스템의 비교

	FAT	NTFS
장점	<ul style="list-style-type: none"> <li>• 호환성 우수</li> <li>• 단순성</li> <li>• 저장량 볼륨에 최적화</li> </ul>	<ul style="list-style-type: none"> <li>• 대용량 볼륨 지원</li> <li>• 디스크의 효율적 사용</li> <li>• 강력한 보안 기능</li> <li>• 자동 압축 및 안정성</li> <li>• 향상된 파일 이름 저장 및 파일 길이 지원</li> </ul>
단점	<ul style="list-style-type: none"> <li>• 보안 취약</li> <li>• 대용량 볼륨에 비효율적</li> </ul>	<ul style="list-style-type: none"> <li>• Windows NT계열 운영체제 외에는 호환 불가</li> <li>• 저장량 볼륨에서 FAT보다 속도 저하</li> </ul>

## 20 Unix/Linux 파일 시스템 구조

부트 블록 (Boot Block)	<ul style="list-style-type: none"> <li>• 부팅 시 필요한 코드를 저장하고 있는 블록</li> <li>• 파일 시스템으로부터 UNIX 커널을 적재시키기 위한 프로그램이 저장되어 있음</li> </ul>
슈퍼 블록 (Super Block)	<ul style="list-style-type: none"> <li>• 전체 파일 시스템에 대한 정보를 저장하고 있는 블록</li> <li>• 파일 시스템마다 하나씩 존재</li> <li>• Logical Volume의 4096 Byte Offset에 위치하며, 크기는 4096 Byte</li> <li>• 슈퍼 블록 정보를 확인하는 명령어 : <code>#dumpfs</code></li> <li>• 슈퍼 블록이 손상되었을 때 점검하는 명령어 : <code>#fsck</code></li> <li>• 저장정보 : 데이터 블록의 개수, 실린더 그룹의 개수, 데이터 블록과 단편의 크기, 하드웨어 설명, 마운트 위치 등의 정보 저장</li> </ul>
아이노드 블록 (I-Node Block)	<ul style="list-style-type: none"> <li>• 파일의 이름을 제외한 해당 파일이나 디렉터리에 대한 모든 정보를 저장 하고 있는 블록</li> <li>• 파일의 실제 주소로서 파일 앞부분의 블록 번지는 직접 가지고, 나머지 블록 번지는 간접 블록 번지로 가짐</li> <li>• 모든 파일은 반드시 하나의 아이노드(I-Node) 블록을 가짐</li> <li>• 저장정보 : 파일의 소유자, 파일 유형, 접근 권한, 접근시간, 파일 크기, 링크 수, 저장된 블록 주소</li> </ul>

데이터 블록 (Data Block)	<ul style="list-style-type: none"> <li>• 실제 데이터가 파일의 형태로 저장되는 공간</li> <li>• I-Node에 포함되며, I-Node가 몇 개의 데이터 블록을 포함하고 있음</li> <li>• 파일은 크게 두 개의 부분으로 구성되며, 하나는 파일에 대한 정보(Meta Data)를, 다른 하나는 실제 데이터를 담고 있는 블록</li> </ul>
------------------------	--

## 21 Unix/Linux 파일 시스템의 종류 및 특징

Minix	<ul style="list-style-type: none"> <li>• 리눅스에서 처음 사용한 파일 시스템</li> <li>• 파일 시스템당 최대 64MB 지원</li> <li>• 파일 이름 최대 30자 지원</li> </ul>
ext2	<ul style="list-style-type: none"> <li>• Boot Sector와 Block Group으로 구성(Block Group은 파일)</li> <li>• 파일 시스템 테이블을 정의하기 위하여 시스템 내의 각 파일을 아이노드(I-Node) 자료구조로 표현하며, 모든 정보를 Super Block과 Group Descriptor Table에 저장</li> <li>• 파일에 들어 있는 데이터는 데이터 블록에 저장되며, 데이터 블록의 크기는 같음</li> <li>• ext2 파일 시스템의 크기는 mke2fs 명령을 통하여 파일 시스템이 만들어질 때 결정</li> <li>• 파일 시스템이 손상되었을 때, FSCK(File System Check)를 이용하여 데이터를 복원할 수 있으나, 캐시에 저장된 데이터를 하드디스크에 저장하는 동안 시스템의 다운 등의 문제 발생 시 파일 시스템이 손상됨</li> <li>• 비정상 종료 시 e2fsck 검사 프로그램 실행 시 검사시간이 오래 걸리며, 다른 작업 수행 불가</li> </ul>
ext3	<ul style="list-style-type: none"> <li>• 커널 2.4 버전부터 지원</li> <li>• 사용자가 직접 데이터 보호유형과 보호 수준 결정 가능</li> <li>• ext2 파일 시스템의 검사 시 또는 복구 시 시간이 오래 걸리거나 시스템을 사용하지 못하는 단점을 보완하기 위하여 저널링(Journaling) 기능 추가 <ul style="list-style-type: none"> <li>- 데이터의 신뢰성과 작업 능률을 향상시킬 수 있음</li> </ul> </li> </ul>
ext4	<ul style="list-style-type: none"> <li>• 64비트 기억공간의 제한을 없애고, ext3 파일 시스템보다 성능을 향상시켰으며, 호환성이 있는 확장 버전</li> <li>• 1EB(Exabyte) 이상의 볼륨과 16TB(Terabyte) 이상의 파일 지원</li> <li>• 지연된 할당(Allocate on Flush) 파일 시스템 기능 지원</li> <li>• 실제 파일 크기에 기반하여 블록 할당을 결정하고, 향상된 블록 할당이 가능하므로 단편화를 줄이고 성능을 향상시킴</li> </ul>

## 22 Windows 로그의 종류 및 특징

응용 프로그램 로그	<ul style="list-style-type: none"> <li>• 응용 프로그램에서 기록한 이벤트 기록</li> <li>• 기본 로그저장 경로 C : Windows system32 config AppEvent.evtx</li> <li>• [Windows Vista 이상] 기본 로그저장 경로 C : Windows system32 winevt logs application.evtx</li> </ul>
보안 로그	<ul style="list-style-type: none"> <li>• 파일이나 다른 개체 생성, 열기, 삭제 등 자원 사용과 관련된 이벤트</li> <li>• 정상적인 로그인 시도 및 비정상적인 로그인 시도와 같은 이벤트 기록</li> <li>• 기본 로그저장 경로 C : Windows system32 config SecEvent.evtx</li> <li>• [Windows Vista 이상] 기본 로그저장 경로 C : Windows system32 winevt logs security.evtx</li> </ul>

시스템 로그	<ul style="list-style-type: none"> <li>• Windows 시스템 구성요소가 기록한 이벤트</li> <li>• 시스템 구성요소가 기록하는 이벤트 유형은 Windows 시스템에서 미리 정해짐</li> <li>• 기본 로그저장 경로 C : Windows system32 config SysEvent.evt</li> <li>• [Windows Vista 이상] 기본 로그 저장 경로 C : Windows system32 winevt logs system.evtx</li> </ul>
설치 로그	<ul style="list-style-type: none"> <li>• 애플리케이션 설치 시 발생하는 이벤트</li> <li>• 프로그램이 잘 설치되었는지, 호환성 문제가 일어나지 않는지를 기록</li> <li>• [Windows Vista 이상] 기본 로그 저장 경로 C : Windows system32 winevt logs setup.evtx</li> </ul>

## 23 Unix/Linux 로그의 종류 및 특징

messages	<ul style="list-style-type: none"> <li>• 로그 파일 중 장치, 네트워크, 부팅 등의 가장 다양한 정보 기록</li> <li>• syslog 계열의 로그이며, 콘솔상의 화면에 출력되는 메시지 기록</li> <li>• su 실패에 대한 로그, 특정 데몬이 비활성화된 로그, 부팅 시에 발생한 오류 등 기록</li> <li>• 시스템의 장애에 대한 기록과 보안 취약점에 의한 공격 흔적 기록</li> <li>• 텍스트로 저장되기 때문에 vi 편집기로 확인 가능</li> </ul>
lastlog	<ul style="list-style-type: none"> <li>• 사용자의 IP 주소별로 가장 최근에 로그인한 시간, 접속 장소, 사용자 이름, IP 주소 정보 등 기록</li> <li>• 확인 명령어 : lastlog</li> </ul>
wtmp	<ul style="list-style-type: none"> <li>• 사용자의 로그인 및 로그아웃 정보를 가지고 있으며, utmp와 같은 데이터 구조 사용</li> <li>• 파일이 생성되는 순간부터 로그인/로그아웃 정보 확인 가능</li> <li>• 텔넷을 통한 로그인뿐만 아니라, FTP를 통한 로그인 등 실질적으로 로그인 프로세스를 거친 정보 및 재부팅과 같이 시스템과 관련된 정보 기록</li> <li>• 확인 명령어 : last</li> </ul>
utmp	<ul style="list-style-type: none"> <li>• 시스템에 현재 로그인한 사용자에 대한 상태정보 기록 예 사용자 이름, 터미널 장치 이름, 원격 로그인 시 원격 호스트 이름, 사용자가 로그인한 시간 등</li> <li>• wtmp와 비슷하나 로그아웃에 대한 정보는 없음</li> <li>• 확인 명령어 : who, w, whoami, whodo, users, finger 등</li> </ul>
pacct	<ul style="list-style-type: none"> <li>• 시스템에 로그인한 사용자가 어떤 명령어를 실행하고, 어떠한 작업을 했는지에 대한 사용 내역 등을 기록</li> <li>• 사용자가 수행한 명령어의 모든 정보를 바이너리 파일로 기록</li> <li>• 피해 시스템의 피해 정도와 백도어 설치 여부 등을 파악하기 위해서는 이 로그 파일 분석이 필수적</li> <li>• 확인 명령어 : lastcomm, acctcom</li> <li>• 유닉스 종류에 따라 'startup' 혹은 'accton' 명령어를 사용하여 설정</li> </ul>
btmpt	<ul style="list-style-type: none"> <li>• 로그인 실패에 대한 로그 정보를 바이너리 파일로 기록</li> <li>• 바이너리 파일로, vi 편집기로 확인할 수 없음</li> <li>• 확인 명령어 : lastb</li> </ul>
sulog	<ul style="list-style-type: none"> <li>• root 계정이나 일반 사용자 계정의 사용자 전환과 관련된 su 명령어 사용에 대한 로그</li> <li>• 날짜 및 시간, 성공(+) · 실패(-), 사용한 터미널 이름, 사용자 정보 등을 기록</li> </ul>
history	<ul style="list-style-type: none"> <li>• 사용자별로 수행한 명령 기록</li> <li>• sh, csh, tcsh, ksh, bash 등 사용자가 사용하는 셸에 따라 .sh_history, .history, .bash_history 등의 파일로 기록</li> <li>• 침해 시스템 분석 시 불법 사용자 계정이나 root 계정의 history 파일을 분석하면 공격자가 시스템에 접근한 후 수행한 명령어를 확인할 수 있다는 점에서 매우 중요한 파일</li> <li>• history 파일은 acct/pacct 파일에서 기록되지 않는 명령어의 전달 인자(Argument)나 디렉터리 위치까지 기록이 가능하므로, 공격자의 행위를 추적하는 데 유용한 정보가 될 수 있음</li> </ul>

secure	<ul style="list-style-type: none"> <li>• Telnet이나 FTP 등 인증과정을 거치는 모든 로그 저장</li> <li>• syslogd 데몬에 의하여 기록</li> <li>• Linux에만 존재하는 로그 파일</li> <li>• 로그 확인 : 텍스트 파일로 저장되어, vi 편집기로 확인 가능</li> </ul>
syslog	<ul style="list-style-type: none"> <li>• 사용자 인증과 관련된 로그 및 커널, 데몬에서 생성된 모든 로그를 포함하여 기록</li> <li>• rsh, rlogin, ftp, finger, telnet, pop3 등에 대한 접속 기록 및 접속 실패 기록</li> <li>• /etc/syslog.conf 파일을 이용하여 환경설정을 할 수 있음</li> <li>• 보안사고 발생 시 가장 먼저 백업받아야 할 파일</li> <li>• 버퍼 오버플로 공격에 대한 기록이 유일하게 남은 파일</li> <li>• Linux의 경우에는 secure 로그에 기록</li> <li>• 로그 확인 : 텍스트 파일로 저장되어, vi 편집기로 확인 가능</li> </ul>

## 24 버퍼 오버플로 취약한 함수 및 안전한 함수

취약한 함수	strcpy(), strcat(), sprintf(), vsprintf(), gets(), scanf(), sscanf()
안전한 함수	strncpy(), strncat(), snprintf(), fget(), fscanf(), vfscanf()

## 25 스택(Stack)과 힙(Heap) 버퍼 오버플로 공격 및 대응방법

스택 (Stack)	공격방법	<ul style="list-style-type: none"> <li>• 프로그램이 변수가 할당된 공간의 데이터 크기에 제한을 두지 않게 하고, 데이터의 길이와 내용을 적절히 조정하여 변수 공간을 넘치게 하는 공격기법</li> <li>• 스택 오버플로가 가장 빈번하게 일어나는 함수인 strcpy() 함수를 이용하여, Parent EBP, Parent EIP에 저장된 기존 값을 덮어쓰도록 함으로써, Parent EBP=0x77777777, Parent EIP=0x73737373으로 변경</li> </ul>
	대응방법	<ul style="list-style-type: none"> <li>• SetUID Bit 제거</li> <li>• Non-Exec Stack 옵션 적용</li> <li>• Boundary 함수 추가</li> <li>• 경계(Canary) 검사(스택 무결성 검사)</li> <li>• 스택 가드(Stack Guard) 사용</li> <li>• Null Canary 사용</li> <li>• ASLR 사용</li> <li>• 스택 실드(Stack Shield)</li> </ul>
힙 (Heap)	공격방법	malloc 계열의 heapalloc, heapfree, malloc, free, new, delete 등의 메모리 할당 함수를 이용하여 프로그램을 할당할 때, 힙 영역에 오버플로를 발생시켜 공격자가 원하는 작업을 수행하는 공격기법
	대응방법	<p>버퍼 오버플로를 대응하기 위한 방법으로는 DEP, ASLR 등이 있음</p> <ul style="list-style-type: none"> <li>- DEP(Data Execution Prevention) : 실행되지 말아야 하는 메모리 영역에서 코드의 실행을 방지하여 임의의 코드가 실행되는 것을 방지하는 방어 기법(Windows에서 사용되며, Linux의 NX-Bit와 같은 것)</li> <li>- ASLR(Address Space Layout Randomization) : PE 파일(exe, dll 등)이 실행될 때마다 (메모리에 로딩될 때마다) Image Base 값을 계속 변경해 주는 기법</li> </ul>

## 26 주요 시스템 관리 명령어

ifconfig	<ul style="list-style-type: none"> <li>• 네트워크 인터페이스의 구성 정보를 확인할 때 사용</li> <li>• IP 주소, 넷 마스크, 브로드캐스트 주소를 설정할 때 사용</li> <li>• 네트워크 인터페이스의 활성화 또는 비활성화할 때 사용</li> </ul>
netstat	네트워크 인터페이스 상태, 연결 상태, 라우팅 테이블 등을 확인할 때 사용
traceroute	목적지까지 도달하는 네트워크 연결 경로를 추적할 때 사용
nslookup	DNS 서버에 질의하여 도메인의 정보를 조회할 때 사용
who, w	현재 컴퓨터에 로그인 한 사용자의 목록을 확인할 때 사용
ps	현재 실행 중인 프로세스의 목록을 확인할 때 사용
top	실시간으로 시스템 상황을 모니터링하거나, 프로세스를 관리를 관리할 때 사용

## 27 침입 탐지 및 차단 도구

Snort	일종의 침입 탐지 시스템(IDS : Intrusion Detection System)으로, 실시간 트래픽 분석, 프로토콜 분석, 내용 검색/매칭, 침입 탐지 규칙(Rule)에 기반하여 오버플로, 포트 스캔, CGI 공격, OS 확인 시도 등의 다양한 공격과 스캔을 탐지할 수 있음
TCP Wrapper	<ul style="list-style-type: none"> <li>• 외부에서 들어오는 클라이언트에 대해 접근통제 기능을 제공</li> <li>• FTP, Telnet, SSH 및 xinetd 기반의 서비스에 대해 접근제어(ACL) 설정이 가능</li> </ul>
Iptables	패킷을 필터링 기능을 가지고 있는 netfilter를 관리하기 위한 도구로, 광범위한 프로토콜 상태 추적, 패킷 애플리케이션 계층 검사, 속도 제한, 필터링 정책을 명시하기 위한 강력한 메커니즘을 제공

## 28 클라우드 서비스별 특징

SaaS (Software as a Service)	완벽히 작동하는 애플리케이션과 해당 애플리케이션을 실행하는 플랫폼, 플랫폼의 기반 인프라를 제공
PaaS (Platform as a Service)	애플리케이션을 실행할 수 있는 플랫폼과 플랫폼 실행에 필요한 IT 인프라를 제공
IaaS (Infrastructure as a Service)	사용자에게 컴퓨팅, 네트워킹 및 스토리지 리소스를 제공
FaaS (Function as a Service)	이벤트 기반 실행 모델로서, 개발자가 인프라를 유지관리하지 않고도 애플리케이션 및 기능을 구축, 실행 및 관리할 수 있도록 지원

## 29 클라우드 컴퓨팅 가상화

서버 가상화 (Server Virtualization)	<ul style="list-style-type: none"> <li>• 서버의 효율성을 높이기 위해 등장한 기술로 가상화 개념의 시초</li> <li>• 가상화를 가능하게 하는 하이퍼바이저(Hypervisor), 하이퍼바이저에 의해 제어되며, 각종 애플리케이션을 실행하기 위한 컴퓨팅 환경인 가상머신(VM)으로 구성</li> <li>• 하드웨어를 가상화하기 위해서는 하드웨어뿐만 아니라 각각의 가상머신들을 관리할 가상머신모니터(Virtual Machine Monitor)와 같은 중간 관리자가 필요함. 이를 하이퍼바이저라고 하며, 가상머신(VM)이 동작할 수 있는 환경을 제공</li> </ul>
데스크탑 가상화 (Virtual Desktop Infrastructure)	<ul style="list-style-type: none"> <li>• 데이터 센터의 서버에서 운영되는 가상의 PC 환경을 의미</li> <li>• 물리적으로는 존재하지 않는 가상의 개별 컴퓨터로 사용자는 모니터, 키보드, 마우스, 스피커 등의 필수적인 입출력장치만을 활용하거나 매우 단순화된 인터페이스만 가지고 컴퓨터를 활용할 수 있음</li> <li>• 가상의 데스크톱을 마치 로컬 시스템처럼 활용할 수 있으며, 모든 작업의 프로세싱과 저장은 데이터 센터에 위치한 서버에서 처리됨</li> <li>• VDI 환경에서는 언제 어디서든 네트워크만 연결이 된다면 서버에 접속하여 자신만의 PC 환경을 구동시킬 수 있음. 사용자는 보통의 PC보다 5~10% 수준의 전력 소모만으로 유사한 컴퓨팅 환경을 구현</li> <li>• 또한 데이터가 로컬 장치가 아닌 서버에 위치하기 때문에 복원, 생성 등의 작업이 쉬워지며, 보안 측면에서도 데이터 센터급의 서비스를 보장받을 수 있음</li> </ul>
애플리케이션 가상화 (Application Virtualization)	<ul style="list-style-type: none"> <li>• 해당 응용 프로그램이 실행되는 운영체제로부터 응용 소프트웨어를 캡슐화하는 기법</li> <li>• 캡슐화된 응용 프로그램은 실제 설치되지는 않으나, 마치 설치된 것처럼 실행됨</li> </ul>

## 30 디지털 포렌식의 기본원칙

정당성의 원칙	<ul style="list-style-type: none"> <li>• 증거가 적법한 절차에 의해 수집되어야 함</li> <li>• 위법 수집증거 배제법칙 : 위법절차를 통해 수집된 증거의 증거능력 부정</li> <li>• 독수의 과실이론 : 위법하게 수집된 증거에서 얻어진 2차 증거도 증거능력이 없음</li> </ul>
재현의 원칙	같은 조건과 상황에서 항상 동일한 결과가 나와야 함
신속성의 원칙	디지털 포렌식의 전 과정이 신속하게 진행되어야 함
연계 보관성의 원칙	<ul style="list-style-type: none"> <li>• 증거물의 수집 · 이동 · 보관 · 분석 · 법정 제출의 각 단계에서 담당자 및 책임자가 명확해야 함</li> <li>• 수집된 하드디스크가 이송단계에서 물리적 손상이 있었다면 이송 담당자는 이를 확인하고 해당 내용을 인수인계, 이후 과정에서 복구 및 보고서 작성 등 적절한 조치를 할 수 있어야 함</li> </ul>
무결성의 원칙	<ul style="list-style-type: none"> <li>• 수집된 증거가 위 · 변조되지 않아야 함</li> <li>• 수집 당시의 데이터 해시값과 법정 제출 시점 데이터의 해시값이 같다면 해시함수의 특성에 따라 무결성을 입증</li> </ul>

### 31 디지털 포렌식의 도구 및 특징

Encase	<ul style="list-style-type: none"> <li>• 1998년 Guidance Software Inc가 사법기관 요구사항을 바탕으로 개발한 컴퓨터 증거 분석용 소프트웨어</li> <li>• 미국 법원이 증거능력을 인정하는 독립 솔루션(2001년 Enron사 회계부정 사건)</li> <li>• 컴퓨터 관련 수사에서 디지털 증거 획득과 분석 기능을 제공</li> <li>• Windows 환경에서도 증거 원본 미디어에 어떠한 영향도 미치지 않으면서 '미리 보기' '증거 사본 작성' '분석' '결과 보고'에 이르는 전자 증거 조사의 모든 과정을 수행할 수 있음</li> </ul>
FTK	<ul style="list-style-type: none"> <li>• 포렌식 도구 중에서 가장 널리 사용되는 도구인 Encase와 더불어 쌍벽을 이루는 도구</li> <li>• 3가지 버전으로 나뉘며, 1.x는 가볍고 사용하기에 편하다는 장점이 있음</li> <li>• 3.x 버전은 데이터베이스를 탑재하여 무겁고 느리게 동작하는 단점이 있으나, Encase 같은 도구 사용 시 대용량 이미지 작업 중 프로그램이 중단되더라도 모든 정보는 데이터베이스에 남아있으므로 중지된 부분부터 작업이 가능함</li> </ul>
TCT	<ul style="list-style-type: none"> <li>• Dan Farmer와 Wietse Venema가 개발</li> <li>• 유닉스 운영체제에서 실행되는 도구</li> <li>• 이전 이벤트를 재현하기 위한 포렌식 데이터를 분석하고 수집하기 위한 매우 강력한 기술을 제공하고 있으며, 네 개의 개별적인 그룹을 하나로 모아둔 것</li> <li>• 운영 중인 시스템에서 현재 상태정보를 캡처하고, 분석할 수 있는 뛰어난 도구</li> <li>• 법원에서 증거로 채택하기 위해 데이터를 모으는 것이 아니라, 어떻게 시스템을 손상시켰는지 알아내도록 돕기 위한 것</li> </ul>
Safeback	<ul style="list-style-type: none"> <li>• 1990년에 처음 시장에 출시되었으며, FBI와 IRS의 범죄 수사부에서 포렌식 조사와 증거 수집을 위해서 사용</li> <li>• 모든 크기의 개별 파티션 또는 전체 디스크를 복제할 수 있으며, 이미지 파일은 SCSI 테이프 또는 다른 저장매체로 전송될 수 있음</li> <li>• 무결성을 제공하기 위해 CRC 함수를 제공하며, 소프트웨어의 감사 정보를 위해 날짜와 시간 정보를 포함하고 있음</li> <li>• DOS 기반의 프로그램이며, 인텔 호환 시스템에서 DOS, Windows, UNIX 디스크를 복제할 수 있음</li> <li>• 이미지를 생성할 때는 어떤 압축이나 변형도 하지 않음</li> </ul>
Autopsy	<ul style="list-style-type: none"> <li>• 사전적 의미로 부검, 검시라는 의미</li> <li>• 2000년경에 발표된 TCT(The Coroner's Toolkit)을 기반으로 지속적 개발</li> <li>• Linux 시스템에서 사용할 수 있는 오픈 소스 기반 포렌식 프로그램으로 알려진 Sleuth Kit를 Windows 시스템에서 사용할 수 있도록 개발된 무료 포렌식 프로그램</li> <li>• Windows와 Unix/Linux를 비롯하여 OS X, Android 등 다양한 운영체제의 파일 시스템 내용을 분석할 수 있으며, 검색 및 타임라인 분석, 해시 필터링 기능 제공. 그리고 기능 확장을 위한 Add-On 모듈을 지원해 Project VIC 및 C4P와 같은 데이터베이스를 통합하여 분석하거나 비디오 분석 모듈 등을 추가할 수 있음</li> </ul>

### 32 iOS와 Android의 보안 체계 비교

	iOS	Android
운영체제	Darwin UNIX에서 파생하여 발전한 OS X의 모바일 버전	리눅스 커널(2.6.25)을 기반으로 만들어진 모바일 운영체제
보안 통제권	애플	개발자 또는 사용자
프로그램 실행 권한	관리자(root)	일반 사용자

응용 프로그램에 대한 서명	애플이 자신의 CA를 통해 각 응용 프로그램을 서명하여 배포	개발자가 서명
샌드박스	엄격하게 프로그램 간 데이터 통신 통제	iOS에 비해 상대적으로 자유로운 형태의 응용 프로그램의 실행이 가능
부팅 절차	암호화 로직으로 서명된 방식에 의한 안전한 부팅 절차 확보	-
소프트웨어 관리	단말 기기별 고유한 소프트웨어 설치 키 관리	-

### 33 BYOD 보안 기술

MDM (Mobile Device Management)	IT 부서가 원격으로 직원 소유 또는 기업 소유의 스마트폰이나 태블릿, 기타 기기를 등록한 후, 직원이나 직원의 업무에 특화된 프로파일을 통해 이를 추적하고 관리하고 보호할 수 있음
MAM (Mobile Application Management)	MDM보다 좀 더 대상을 특정한 솔루션으로, 기기 자체가 아니라 기업용 애플리케이션과 관련 데이터만을 통제할 수 있음
UEM (Unified Endpoint Management)	<ul style="list-style-type: none"> <li>기업 내의 모든 하드웨어를 하나의 전략으로 포괄</li> <li>IT 부서가 스마트폰과 태블릿, 노트북, 데스크톱, 그리고 사물 인터넷 기기까지 모든 것을 원격에서 프로비저닝하고, 제어하고 보호</li> </ul>
EMM (Enterprise Mobility Management)	<ul style="list-style-type: none"> <li>다양한 소프트웨어 관리 도구를 하나의 우산 아래 모으는 것을 목표</li> <li>포괄적이고 하드웨어를 가리지 않은 원격 기기 관리 방법으로, MDM과 MAM을 통해 기기 환경 설정과 기기에서 생성된 기업 데이터도 관리</li> </ul>

### 34 블루투스의 취약점과 위협

블루 프린팅 (BluePrinting)	<ul style="list-style-type: none"> <li>블루투스 공격 장치의 검색 활동을 의미</li> <li>장치 간 종류를 식별하기 위해 서비스 발견 프로토콜(Service Discovery Protocol)을 송·수신하는데, 공격자는 이를 이용해 공격이 가능한 블루투스 장치를 검색하고, 모델을 확인할 수 있음</li> </ul>
블루 스나프 (BlueSnarf)	<ul style="list-style-type: none"> <li>블루투스의 취약점을 이용하여 장비의 파일에 접근하는 공격</li> <li>공격자는 블루투스 장치 간 상호인증 없이 정보를 간편하게 교환할 수 있는 OPP(OBEX Push Profile)를 사용하여 정보를 열람</li> </ul>
블루 버그 (BlueBug)	<ul style="list-style-type: none"> <li>블루투스 장치 간 취약한 연결 관리를 악용한 공격</li> <li>블루투스 기기는 한 번 연결되면 다시 연결하지 않아도 서로 연결되는 인증 취약점을 이용하여 공격</li> </ul>
블루 재킹 (Bluejacking)	<ul style="list-style-type: none"> <li>블루투스를 이용해 스팸처럼 명함을 익명으로 퍼트리는 것(데이터의 이동이나 변조하는 것이 아님)</li> <li>명함에는 주로 공격자의 메시지가 들어 있음</li> </ul>



## 02

## 네트워크 보안

## 1 프로토콜의 3요소

구문(Syntax)	데이터의 형식, 부호화 방법, 전기적 신호 레벨에 대한 사항
의미(Semantics)	오류, 동기 및 흐름 제어 등의 각종 제어절차에 대한 사항
타이밍(Timing)	송·수신 간 또는 통신 시스템 간 통신속도 및 순서 등에 대한 사항

## 2 OSI 7계층 계층별 역할

7계층	응용	여러 프로토콜 개체에 대하여 사용자 인터페이스 제공
6계층	표현	<ul style="list-style-type: none"> <li>• 데이터 표현형식의 차이를 해결하기 위하여 서로 다른 형식으로 변환</li> <li>• 부호화(Encoding), 압축(Compression), 암호화(Encryption)</li> </ul>
5계층	세션	<ul style="list-style-type: none"> <li>• 응용프로그램 간 세션을 형성하고 관리하며, 상위 계층인 표현 계층에서 두 개 이상의 요소 간 통신을 가능하게 함</li> <li>• 통신을 동기화하고 대화 제어</li> <li>• 애플리케이션 접근 스케줄링 담당</li> </ul>
4계층	전송	<ul style="list-style-type: none"> <li>• 종단(End to End) 간 신뢰성 있는 데이터 전송 보장</li> <li>• 메시지의 전송, 오류 제어, 흐름 제어, 연결 제어 기능</li> <li>• 출발지와 목적지의 포트 번호가 결정되는 계층</li> <li>• 부하분산 : 과도한 트래픽 분산</li> </ul>
3계층	네트워크	<ul style="list-style-type: none"> <li>• 패킷의 목적지 IP 주소를 참조하여, 최적의 경로를 설정·전송하는 계층</li> <li>• 라우팅, 흐름 제어, 단편화, 오류 제어 수행</li> </ul>
2계층	데이터 링크	<ul style="list-style-type: none"> <li>• 점 대 점(Point To Point) 간 신뢰성 있는 전송 보장</li> <li>• 프레임 구성 : 헤더+네트워크 계층에서 받은 패킷+트레이일러</li> <li>• 서브 계층(Sub Layer) <ul style="list-style-type: none"> <li>- LLC(Logical Link Control) : 오류 제어, 흐름 제어, 오류 검사 및 복구, 비트 동기 및 식별 기능 수행</li> <li>- MAC(Media Access Control) : MAC 주소를 사용한 매체접근 방식</li> </ul> </li> </ul>
1계층	물리	시스템 간 링크를 활성화하고 관리하기 위한 사용자 장비와 네트워크 종단 장비 간 기계적, 전기적, 기능적, 절차적 특성과 인터페이스 정의

### 3 OSI 7계층 계층별 특징

응용	전송 단위	메시지(Message)
	프로토콜	FTP, Telnet, SMTP, HTTP, DNS, SNMP, POP3, IMAP 등
	장비	L7 스위치(L7 Switch), 게이트웨이(Gateway)
표현	전송 단위	메시지(Message)
	프로토콜	JPEG, MPEG, ASCII, GIF 등
	장비	게이트웨이(Gateway)
세션	전송 단위	메시지(Message)
	프로토콜	SSL, Socks
	장비	게이트웨이(Gateway)
전송	전송 단위	세그먼트(Segment)
	프로토콜	TCP, UDP, SPX 등
	장비	L4 스위치(L4 Switch), 게이트웨이(Gateway)
네트워크	전송 단위	패킷(Packet)
	프로토콜	IP, IPSec, IPX, ICMP, IGMP, ARP, RARP 등
	장비	라우터(Router), L3 스위치(L3 Switch)
데이터링크	전송 단위	프레임(Frame)
	프로토콜	PPP, PPTP, L2F, L2TP, HDLC, SDLC, Ethernet 등
	장비	브리지(Bridge), 스위치(Switch)
물리	전송 단위	비트(Bit)
	프로토콜	EIA RS-232C, V.24, V.35, X.21 등
	장비	리피터(Repeater), 더미 허브(Dummy Hub)

### 4 IPv4와 IPv6의 비교

구분	IPv4	IPv6
주소 길이	32비트	128비트
표시 방법	8비트 단위 4부분 10진수 표기 예) 203.211.172.128	16비트 단위 8부분 16진수 표기 예) 2013:0102:ABCD:ABDF:0000:0F00:FFFF:2002
주소 개수	약 43억 개	거의 무한대(약 43억×43억×43억×43억)
주소 할당	A, B, C, D, E 클래스 단위의 비순차적 할당(비효율적)	규모 및 단말기 수에 따른 순차적 할당(효율적)
품질 제어	베스트 에퍼트(Best Effort) 방식으로 품질보장이 곤란(유형이나 서비스에 대한 QoS 일부 지원)	등급별, 서비스별 패킷을 구분할 수 있어 품질보장 용이(트래픽 클래스, 플로우 레이블에 의한 QoS 지원)

보안 기능	IPSec 프로토콜 설치	확장기능에서 기본적으로 제공
플러그 & 플레이	없음	있음(Auto Configuration)
모바일	곤란(비효율적)	용이(효율적)
웹 캐스팅	곤란	용이(범위 필드 추가)

## 5 IPv4의 통신방식

유니캐스트 (Unicast)	<ul style="list-style-type: none"> <li>• 단일 인터페이스를 지정하며, MAC 주소 기반으로 상대측 IP 주소를 목적으로 하는 일대일 통신 방식</li> <li>• 유니캐스트 주소로 전송된 패킷은 그 주소에 해당하는 인터페이스에만 전달</li> </ul>
멀티캐스트 (Multicast)	<ul style="list-style-type: none"> <li>• 여러 노드에 속한 인터페이스의 집합을 지정하며, 멀티캐스트 주소로 전송된 패킷은 그 주소에 해당하는 모든 인터페이스에 전달</li> <li>• 하나 이상의 송신자가 네트워크의 특정 그룹에 패킷 전송</li> </ul>
브로드캐스트 (Broadcast)	자신의 호스트가 속해 있는 네트워크 전체를 대상으로 패킷을 전송하는 일대다 통신 방식

## 6 IPv6의 통신방식

유니캐스트 (Unicast)	<ul style="list-style-type: none"> <li>• 단일 인터페이스를 지정하며, MAC 주소 기반으로 상대측 IP 주소를 목적으로 하는 일대일 통신 방식</li> <li>• 유니캐스트 주소로 전송된 패킷은 그 주소에 해당하는 인터페이스에만 전달</li> </ul>
멀티캐스트 (Multicast)	<ul style="list-style-type: none"> <li>• 여러 노드에 속한 인터페이스의 집합을 지정하며, 멀티캐스트 주소로 전송된 패킷은 그 주소에 해당하는 모든 인터페이스에 전달</li> <li>• 하나 이상의 송신자가 네트워크의 특정 그룹에 패킷 전송</li> </ul>
애니캐스트 (Anycast)	<ul style="list-style-type: none"> <li>• IPv6에서 브로드캐스트의 대안으로 사용하는 통신 방식</li> <li>• 단일 송신자와 그룹 내에서 가장 가까운 곳에 있는 일부 수신자 사이의 통신</li> <li>• 여러 노드에 속한 인터페이스의 집합을 지정하며, 애니캐스트 주소로 전송된 패킷은 그 주소에 해당하는 인터페이스 중 하나의 인터페이스에 전달</li> <li>• 한 호스트가 호스트 그룹을 위하여 라우팅 테이블을 효과적으로 갱신할 수 있도록 설계. 어떤 게이트웨이가 가장 가까이 있는지를 결정할 수 있으며, 유니캐스트 통신인 것처럼 그 호스트에 패킷을 전송할 수 있으며, 그 호스트는 모든 라우팅 테이블이 갱신될 때까지 그룹 내의 다른 호스트에게 차례로 애니캐스트할 수 있음</li> </ul>

## 7 TCP/UDP 기반 응용프로토콜과 포트 번호

Port 번호	프로토콜	응용프로그램
20	TCP	FTP(Data)
21	TCP	FTP(Control)
22	TCP	SSH
23	TCP	Telnet
25	TCP	SMTP
53	UDP	DNS
67, 68	UDP	DHCP
69	UDP	TFTP
80	TCP	HTTP
110	TCP	POP3
161, 162	UDP	SNMP
443	TCP	SSL
1433	TCP	MS SQL Server
1521	TCP	Oracle
3306	TCP	MySQL

## 8 TCP와 UDP 프로토콜 비교

	TCP	UDP
일반적 특징	애플리케이션이 네트워크 계층 문제와는 무관하게 데이터를 안정적으로 송수신할 수 있도록 하는 프로토콜	단순하고, 빠르며, 애플리케이션이 네트워크 계층에 접근할 수 있도록 하는 인터페이스만 제공할 뿐 다른 것은 거의 하지 않음
프로토콜 연결 수립	연결형	비연결형
애플리케이션의 데이터 입력 인터페이스	<ul style="list-style-type: none"> <li>스트림 기반</li> <li>애플리케이션은 특정한 구조 없이 데이터를 송신</li> </ul>	<ul style="list-style-type: none"> <li>메시지 기반</li> <li>애플리케이션은 데이터를 별도의 패키지로 송신</li> </ul>
신뢰성과 승인	<ul style="list-style-type: none"> <li>메시지 전송을 신뢰할 수 있음</li> <li>모든 데이터에 대한 승인이 있음</li> </ul>	<ul style="list-style-type: none"> <li>신뢰성이 없음</li> <li>승인이 없는 전송 방식</li> </ul>
재전송	모든 데이터 전송을 관리하며, 손실된 데이터는 자동으로 재전송	<ul style="list-style-type: none"> <li>수행하지 않음</li> <li>애플리케이션은 손실 데이터를 탐지하고 필요할 경우 재전송해야 함</li> </ul>
데이터 흐름 관리 기능	<ul style="list-style-type: none"> <li>슬라이딩 윈도우를 이용한 흐름 제어를 함</li> <li>윈도우 크기를 적절히 조정하고, 혼잡 회피 알고리즘을 사용</li> </ul>	없음
부하	높음	낮음

전송 속도	느림	빠름
적합한 데이터 양	소형에서 초대형 데이터까지(최대 수 기가 바이트)	소형에서 중형 데이터(최대 수백 바이트)
프로토콜을 사용하는 애플리케이션의 유형	<ul style="list-style-type: none"> <li>신뢰할 수 있는 방법으로 데이터를 송신해야 하는 대부분의 프로토콜과 애플리케이션</li> <li>대부분의 파일/메시지 전송 프로토콜을 포함</li> </ul>	데이터의 완전성보다 전달 속도가 중요하고, 소량의 데이터를 송신하고, 멀티캐스트/브로드캐스트를 사용하는 애플리케이션
애플리케이션과 프로토콜	FTP, Telnet, SMTP, DNS, HTTP, POP, NNTP, IMAP, BGP, IRC, NFS(개선 버전)	멀티미디어 애플리케이션, DNS, BOOTP, DHCP, TFTP, SNMP, RIP, NFS(초기 버전)

## 9 디스턴스 벡터와 링크 상태 라우팅 프로토콜의 비교

디스턴스 벡터 (Distance Vector)	<ul style="list-style-type: none"> <li>라우팅 정보전송 시 목적지 네트워크와 메트릭(Metric) 값을 알려줌</li> <li>대표적인 프로토콜 : RIP, EIGRP, BGP</li> </ul>
링크 상태 (Link State)	<ul style="list-style-type: none"> <li>목적지 네트워크 메트릭을 특정 네트워크가 접속되어 있는 라우터와 그 라우터와 인접한 라우터 등에 광고</li> <li>대표적인 프로토콜 : OSPF, ISIS</li> </ul>

## 10 랜카드(LAN Card)의 기능

PC와의 통신	NIC가 PC로부터 데이터를 넘겨받음
Buffering	데이터를 임시 저장
Frame 형성	이더넷의 프레임 크기(Frame Size)인 64~1,518바이트의 프레임을 생성
직렬/병렬 변환	PC로부터 받은 병렬신호를 케이블에 전송하기 위해 직렬신호로 변환
Encoding/Decoding	맨체스터(Manchester) 인코딩을 사용
Cable Access	케이블이 사용되지 않고 있음을 확인하고, 전송 후 충돌 여부를 감지
전송	데이터를 전기적인 펄스 신호로 변환해서 전송

## 11 스위치의 프레임 포워딩 방식

컷 스루 (Cut Through)	<ul style="list-style-type: none"> <li>수신한 프레임의 목적지 주소를 확인하고, 목적지 주소의 포트로 프레임을 즉시 전송</li> <li>전체 프레임이 수신되기 전에 중계를 하기 때문에 프레임 구분 없이 모두 중계</li> <li>프레임의 첫 6바이트(48비트)만 확인 후 목적지로 전송</li> <li>오류 처리가 어려우며, 복구능력이 떨어짐</li> <li>중계 지연 시간을 최소화하기 위해 사용하는 방법</li> </ul>
스토어 & 포워드 (Store&Forward)	<ul style="list-style-type: none"> <li>모든 프레임을 수신한 다음 처리</li> <li>오류, 목적지 주소, 출발지 주소 확인 후 목적지로 전송</li> <li>중계 지연 시간이 길어짐</li> <li>오류가 발생하면 모든 프레임을 버리고 재전송 요구를 하기 때문에 오류 복구능력이 뛰어남</li> <li>전송 매체에서 오류가 자주 발생하거나 출발지와 목적지의 전송 매체가 다를 경우에 자주 사용되는 방식</li> </ul>
프래그먼트 프리 (Fragment Free)	<ul style="list-style-type: none"> <li>컷 스루(Cut Through) 방식과 스토어&amp;포워드(Store&amp;Forward) 방식의 장점을 혼합</li> <li>프레임의 앞부분에서 오류가 없음을 보증할 수 있는 길이인 64바이트(512 비트)만 확인한 후 전송</li> <li>컷 스루(Cut Through) 방식보다는 오류 복구 능력이 뛰어남</li> </ul>

## 12 라우터와 스위치의 비교

	라우터(Router)	스위치(L3 Switch)
OSI Layer	3계층	3계층
처리 방식	소프트웨어	하드웨어
포트별 속도	같은 속도 지원	다른 속도 지원
지원하는 Layer 2	Ethernet, Fast Ethernet, Token Ring, FDDI, ATM, WAN	Fast Ethernet, Gigabit Ethernet
포워딩 능력	느림 (CPU 성능과 가격에 따라 다름)	빠름 (Near Wire Speed)
대기시간	약 200ms	10ms(100Mbps) 이하
관리 및 프로그램 가능성	매우 높음	낮음
지원 프로토콜	All	IP(일부 IPX)
라우팅 프로토콜	All	RIPv1, v2 OSPF(일부 DVMRP)
WAN 지원	지원함	지원하지 않음
비용	높음	낮음

### 13 무선랜 암호화 기술

WEP (Wired Equivalent Privacy)	<ul style="list-style-type: none"> <li>1999년 9월 와이파이 보안표준으로 채택</li> <li>성능개량 보안 취약점 완화 방법 등이 다양하게 나와 있으나 상당히 취약한 표준으로 전락했으며, 비영리 와이파이 기술 인증기관인 와이파이 연합(Wi-Fi Alliance) 또한 2004년 공식적으로 WEP 방식을 퇴출시킴</li> </ul>
WPA (Wi-Fi Protected Access)	<ul style="list-style-type: none"> <li>2003년 와이파이 보안표준으로 채택</li> <li>취약한 WEP 표준을 대체하기 위해 무선 데이터 보호(WPA) 방식을 채택</li> <li>WPA에 사용된 키는 256비트로, WEP의 64비트 및 128비트 보다 대폭 강력해짐</li> </ul>
WPA2 (Wi-Fi Protected Access)	<ul style="list-style-type: none"> <li>2006년 WPA는 무선 데이터 보호 II (WPA2) 방식으로 대체</li> <li>WPA와 달리 AES 알고리즘이 기본 적용되며, CCMP(Counter Cipher Mode with Block Chaining Message Authentication Code Protocol) 방식이 TKIP를 대체</li> </ul>

### 14 무선랜(Wireless LAN)의 취약점

물리적인 취약성	<ul style="list-style-type: none"> <li>강한 전파로 인하여 서비스 범위를 초과하여 전송</li> <li>불법 장비 설치 및 도청 가능성</li> <li>AP의 도난</li> </ul>
인증 및 암호화 메커니즘 취약성	<ul style="list-style-type: none"> <li>취약한 보안 기능</li> <li>낮은 암호화 수준(WEP)</li> <li>기본 패스워드 설정 및 사용</li> <li>AP에 대한 서비스거부(DoS) 공격</li> </ul>

### 15 네트워크 공격의 종류

	수동적 공격	능동적 공격
특징	직접적인 피해 없음	직접적인 피해 있음
탐지 가능성	어려움	쉬움
공격	스니핑(Sniffing), 도청(Eavesdrop), 패킷 분석(Packet Analysis)	DoS/DDoS 공격, 스푸핑(Spoofing), 세션 하이재킹(Session Hijacking)

## 16 서비스거부(DoS) 공격 및 대응방법

SYN Flooding	공격기법	<ul style="list-style-type: none"> <li>TCP의 3-Way Handshake 취약점을 이용한 공격</li> <li>출발지 IP 주소를 존재하지 않는 IP 주소로 변조한 후 다량의 SYN 패킷을 전송하여, 공격 대상 시스템의 백로그 큐(Backlog Queue)를 가득채움으로써, 시스템을 마비시키는 공격</li> </ul>
	탐지방법	<ul style="list-style-type: none"> <li>공격자가 전송한 패킷의 IP 헤더와 TCP 헤더 분석</li> <li>TCP 헤더 분석결과 Flag가 SYN이면 공격자가 전송하는 패킷 카운트를 증가, SYN/ACK 이 면 카운트를 감소 → SYN 개수 파악</li> <li>공격 인정시간 내 SYN 개수가 공격 인정횟수 이상이면 SYN Flooding으로 탐지</li> </ul>
	대응방법	<ul style="list-style-type: none"> <li>내부 IP 주소를 출발지 IP 주소로 변조하여 들어오는 트래픽 차단</li> <li>비정상적인 IP 주소를 출발지 IP 주소로 변조하여 들어오는 트래픽 차단</li> <li>백로그 큐 크기를 늘림</li> <li>syncookies 기능 사용</li> <li>시스템의 네트워크 설정 최적화</li> <li>가상서버 커널 패치</li> <li>방화벽과 라우터에서 차단기능 사용(인터셉트 모드, 와치 모드)</li> </ul>
UDP Flooding	공격기법	<ul style="list-style-type: none"> <li>UDP 프로토콜을 이용하여 서버에 가상 데이터를 연속적으로 전송함으로써, 서버의 부하 및 네트워크 과부하를 발생시키는 공격</li> <li>UDP의 비연결성 및 비신뢰성 특성을 이용한 공격</li> </ul>
	탐지방법	<ul style="list-style-type: none"> <li>공격자가 전송한 패킷에서 UDP 헤더 분석</li> <li>대상의 포트 번호를 확인 : 17, 135, 137번 포트</li> <li>UDP 포트 스캔이 아니면 UDP Flooding으로 간주</li> <li>패킷의 횟수를 카운트하여 일정 시간 내에 공격 인정횟수 이상이면 UDP Flooding으로 간주</li> </ul>
	대응방법	<ul style="list-style-type: none"> <li>불필요한 UDP 서비스 비활성화</li> <li>방화벽과 라우터에서 불필요한 UDP 서비스 차단</li> </ul>
Teardrop	공격기법	<ul style="list-style-type: none"> <li>IP 패킷 조각(Fragment)을 아주 작거나 겹치게 만들어 전송함으로써, 재조립하는데 과부하를 일으키는 공격</li> <li>Windows뿐만 아니라 리눅스 시스템에서도 공격 가능</li> <li>공격을 받을 경우 네트워크 연결 끊김 또는 블루스크린 오류화면을 출력하고 시스템 정지</li> </ul>
	탐지방법	<ul style="list-style-type: none"> <li>패킷을 분석하여 단편화를 확인하고 단편화되어 있으면 카운트</li> <li>패킷의 데이터 부분에 데이터가 존재하는지를 확인하여 데이터가 없으면 공격으로 간주</li> <li>패킷 횟수를 카운트하여 일정 시간 내에 공격 인정횟수 이상이면 공격으로 탐지</li> </ul>
	대응방법	<ul style="list-style-type: none"> <li>시스템의 운영체제 관련 보안패치를 최신 버전으로 업데이트</li> <li>변종이 많으므로 방화벽만으로는 근본적인 해결 불가능</li> <li>IP 패킷의 재조립 시 0보다 작은 패킷에 대한 처리루틴 포함</li> </ul>
Smurf	공격기법	<ul style="list-style-type: none"> <li>ICMP의 Ping을 이용한 공격</li> <li>위조된 IP 패킷(출발지 IP 주소를 공격대상의 IP 주소로 변조)과 브로드캐스트 IP 주소를 이용하여 네트워크나 시스템에 과부하를 일으키는 공격</li> <li>네트워크의 라우터로 하여금 ICMP Echo를 서브넷 내의 모든 호스트에 전송하여, 조작된 공격대상의 IP 주소로 응답이 폭주하도록 만들</li> <li>전체 대역폭을 점유하여 네트워크를 마비시킴</li> </ul>
	탐지방법	<ul style="list-style-type: none"> <li>패킷을 분석하여 ICMP_ECHO_REPLY 확인</li> <li>IP 스푸핑 여부 확인</li> <li>패킷 횟수를 카운트하여 일정 시간 내에 공격 인정횟수 이상이면 공격으로 간주</li> </ul>



	대응방법	<ul style="list-style-type: none"> <li>직접 브로드캐스트(Directed Broadcast) 비활성화 : 중간 경유지로 쓰이는 것을 막기 위하여 라우터에서 다른 네트워크로부터 들어오는 IP 브로드캐스트 패킷을 차단하도록 설정</li> <li>ICMP 메시지 차단, ICMP Echo Reply 패킷 차단</li> </ul>
Ping of Death	공격기법	<ul style="list-style-type: none"> <li>ICMP Ping 패킷을 비정상적으로 크게 만들어 전송하면 공격 네트워크에 도달하는 동안 아주 작은 조각(Fragment)이 됨</li> <li>공격대상 시스템은 분할된 패킷을 모두 처리해야 하므로, 정상적인 Ping의 경우보다 많은 부하가 걸리게 되어 시스템의 성능 저하 및 마비를 일으키는 공격</li> </ul>
	탐지방법	<ul style="list-style-type: none"> <li>패킷을 분석하고 ICMP 패킷을 확인하여 Ping 여부와 크기 확인</li> <li>패킷 횟수를 카운트하여 일정 시간 내에 공격 인정횟수 이상이면 공격으로 간주</li> </ul>
	대응방법	반복적으로 들어오는 일정 수 이상의 ICMP 패킷을 차단하도록 설정
LAND	공격기법	<ul style="list-style-type: none"> <li>출발지와 목적지의 IP 주소를 동일하게 공격대상의 IP 주소로 변조하여 전송</li> <li>패킷을 받은 시스템은 응답하기 위하여 출발지 IP 주소를 참조하는데, 출발지 IP 주소가 자신의 IP 주소이기 때문에 자신에게 응답하게 되고, 이 과정을 반복하게 하여 과부하를 유발</li> <li>동시 사용자 수를 점유하고, 루프 상태가 되도록 함으로써, 시스템 과부하 유발</li> </ul>
	탐지방법	<ul style="list-style-type: none"> <li>패킷에서 TCP 헤더를 분석</li> <li>IP 패킷 중 출발지와 목적지 IP 주소가 동일한 것이 존재하는지 확인</li> <li>패킷 횟수를 카운트하여 일정 시간 내에 공격 인정횟수 이상이면 공격으로 탐지</li> </ul>
	대응방법	라우터와 방화벽에서 내부 IP 주소와 동일한 출발지 IP 주소 차단
Newtear/ Targa/ Nestea	공격기법	<ul style="list-style-type: none"> <li>작업 중 저장되지 않은 데이터를 모두 삭제하는 공격</li> <li>단편화된 패킷을 재조립할 때 오프셋 값을 비정상적으로 크게 함으로써, 오버플로우를 발생시키고 시스템을 마비시키는 공격</li> <li>침입 탐지 시스템이나 방화벽 우회 가능</li> </ul>
	대응방법	변종 공격이 많으므로 운영체제의 최신 버전 패치

## 17 분산 서비스거부(DDoS) 공격의 종류 및 특징

Trinoo	공격자가 하나 혹은 그 이상의 마스터에 접속하여 여러 에이전트(슬레이브)에게 특정 시스템을 일시에 공격하도록 명령을 내리면 공격대상 시스템에 다량의 UDP 패킷(UDP Flood 공격)을 전송하여 공격대상 시스템을 마비시킴
TFN (Tribe Flood Network)	<ul style="list-style-type: none"> <li>• Trinoo의 발전형</li> <li>• UDP Flood, TCP SYN Flood, ICMP Echo, Smurf 공격 등 다양한 기능 포함</li> <li>• 공격자가 클라이언트(혹은 마스터) 프로그램을 이용하여 공격 명령을 TFN 서버(혹은 데몬)로 전송함으로써 이루어짐</li> <li>• 데몬(Daemon)은 설치 시 자신의 프로세스 이름을 변경함으로써, 프로세스 모니터링을 회피</li> <li>• 지정된 TCP 포트에 백도어(Backdoor)를 실행시킬 수 있음</li> <li>• UDP 패킷 헤더가 실제 UDP 패킷보다 3바이트 만큼 더 큼</li> <li>• TCP 패킷의 길이는 항상 0(정상적인 패킷이라면 0이 될 수 없음)</li> </ul>
TFN2K	<ul style="list-style-type: none"> <li>• TFN의 발전형으로, Targa를 제작한 Mixer에 의하여 개발되었고, Targa가 기본적으로 제공하는 서비스거부(DoS) 공격 외에 5개의 공격이 추가됨</li> <li>• TFN2K에서 제공하는 공격은 Change IP Antispoof-Level, Change Packet Size, Binf Root-Shell, UDP Flood, TCP/SYN Flood, ICMP/Ping Flood, ICMP/Smurf Flood, MIX Flood, Targa3 Flood 등이 있음</li> <li>• 모든 명령은 CAST-256 알고리즘으로 암호화</li> <li>• 분산 모드로 작동하여 서버 모듈, 클라이언트 모듈로 구분 : 클라이언트 모듈은 서버 모듈을 제어하는 모듈로서, 서버에게 어떤 공격을 누구에게 할 것인지를 지시</li> <li>• 지정된 TCP 포트에 백도어 실행 가능</li> <li>• TCP 패킷 헤더의 길이는 항상 0</li> </ul>
Stacheldraht	<ul style="list-style-type: none"> <li>• TFN이나 TFN2K처럼 ICMP Flood, SYN Flood, UDP Flood와 Smurf 등의 공격을 이용함으로써, DDoS 공격을 할 수 있는 기능을 가짐</li> <li>• TFN 도구에 공격자, 마스터, 에이전트 간 통신에 암호화 기능만 추가된 도구</li> </ul>

## 18 피싱과 파밍의 비교

구분	피싱(Phishing)	파밍(Pharming)
목적	가짜 사이트로 유도하여 개인정보 탈취	가짜 사이트로 유도하여 개인정보 탈취
공격방법	<ul style="list-style-type: none"> <li>• 실제 도메인 이름과 유사한 가짜 도메인 이름 사용</li> <li>• 진짜처럼 위장된 가짜 사이트로 접속하여 개인정보 입력 유도</li> </ul>	조직, 목적, 분류 등 명칭을 영문 약어로 표시한 최상위 도메인 이름 사용
특징	주로 이메일, SMS 등에 첨부된 링크를 통해 접속 유도	정상적인 도메인 입력으로도 공격이 가능하기 때문에 별다른 유인방법이 불필요
위험성	사용자가 세심한 주의를 기울이면 공격탐지 가능	<ul style="list-style-type: none"> <li>• 실제 도메인 이름이 그대로 사용되기 때문에 공격 탐지 어려움</li> <li>• 사용자가 많은 DNS 캐시 서버에 공격을 성공할 경우 피해가 광범위</li> </ul>

## 19 포트 스캐닝 종류 및 특징

TCP Scanning	Open	<ul style="list-style-type: none"> <li>• TCP Connect 스캔</li> <li>• TCP 프로토콜의 3-Way Handshaking을 이용하여, 각 포트에 완전한 TCP 연결 확립 후 대상 포트에 SYN 패킷 전송</li> <li>• 속도가 느리고, 상대 시스템에 로그 기록 남음</li> <li>• 열린 포트 : SYN/ACK, 닫힌 포트 : RST/ACK 패킷 응답</li> </ul>
	Haif Open	<ul style="list-style-type: none"> <li>• SYN 스캔</li> <li>• 완전한 TCP 연결을 하지 않고, Half-Open 연결을 통하여 포트의 Open/Close 상태를 확인하기 때문에 TCP Connect() 스캔과 달리 시스템에 로그 기록 남지 않음</li> <li>• 스캔 속도가 TCP Connect() 스캔보다 빠르므로 많이 사용</li> </ul>
	Stealth	<ul style="list-style-type: none"> <li>• SYN 패킷 자체를 전송하지 않고 스캐닝</li> <li>• FIN, NULL, X-MAS, ACK 스캔</li> <li>• UNIX 계열 시스템에서만 사용할 수 있으므로, TCP FIN, NULL, X-MAS 스캔을 수행 후 응답이 없다면 윈도우 계열의 시스템이라고 판단할 수 있음</li> <li>• 열린 포트 : 무응답, 닫힌 포트 : RST 패킷 응답</li> </ul>
UDP Scanning	UDP Port	<ul style="list-style-type: none"> <li>• 공격대상 시스템 포트를 목적지 포트로 하여, UDP 패킷 전송</li> <li>• 열린 포트 : 무응답, 닫힌 포트 : ICMP Unreachable 메시지 응답</li> </ul>

## 20 스위치 환경에서의 스니핑 공격유형

Switch Jamming	<ul style="list-style-type: none"> <li>• 스위치가 MAC 주소 테이블을 기반으로 포트에 패킷을 스위칭할 때 정상적인 스위칭을 하지 못하도록 하는 공격. MACOF 공격이라고도 함</li> <li>• 공격자는 위조된 MAC 주소를 지속적으로 전송하여, MAC 주소 테이블을 오버플로우되게 함으로써, 스위치가 허브처럼 동작하여, 모든 네트워크 세그먼트로 이더넷 프레임이 브로드캐스팅하도록 함</li> </ul>
ARP Spoofing	<ul style="list-style-type: none"> <li>• 두 대상의 MAC 주소를 공격자 자신의 MAC 주소로 변조하여 중간에 서 패킷을 가로채는 공격</li> <li>• 네트워크 내에서 두 대상의 IP 주소에 대한 MAC 주소를 공격자의 MAC 주소로 변조하여 클라이언트에서 서버로 가는 패킷이나 서버에서 클라이언트로 가는 패킷이 공격자에게 전송되도록 하는 공격</li> </ul>
ARP Redirect	<ul style="list-style-type: none"> <li>• 위조된 ARP Reply를 주기적으로 브로드 캐스트함으로써, 네트워크 내의 호스트들이 공격자를 라우터로 인식하도록 만들고, 외부로 전달되는 모든 패킷이 공격자를 한번 거친 후 라우터로 전송되도록 하는 공격</li> <li>• 이때 자신이 공격자임을 숨기기 위해 스니핑 후 패킷을 IP 포워딩하여 라우터로 전달해줘야 함</li> <li>• 이 과정에서 공격자는 네트워크에 있는 호스트가 어떤 패킷을 라우터를 통해 외부로 통신하려 하는지 감시와 분석을 할 수 있음</li> </ul>
ICMP Redirect	<ul style="list-style-type: none"> <li>• ARP Redirect와 동일하게 공격자가 라우터로 인식되도록 하는 공격</li> <li>• 여러 라우터가 존재하는 네트워크에서 최적의 경로를 찾기 위해 여러 알고리즘으로 동작하게 되는데, 공격자는 공격대상에게 '자신이 라우터이고 최적의 경로'라는 변조된 ICMP Redirect를 보내 데이터를 전달 받음</li> <li>• 이때 자신이 공격자임을 숨기기 위해 스니핑 후 패킷을 IP 포워딩하여 라우터로 전달해줘야 함</li> <li>• 이 과정에서 공격자는 네트워크에 있는 호스트가 어떤 패킷을 라우터를 통해 통신하려 하는지 감시와 분석을 할 수 있음</li> </ul>
SPAN 포트 태핑 (Port Mirroring)	<ul style="list-style-type: none"> <li>• 스위치의 포트 미러링 기능을 이용한 공격</li> <li>• 각 포트에 전송되는 데이터를 미러링하는 포트에도 동일하게 전달하는 것으로 침입 탐지시스템, 네트워크 모니터링, 로그 시스템 등에 많이 사용. 이 포트를 이용해 모든 정보를 볼 수 있음</li> </ul>

## 21 방화벽의 구조

배스천호스트 (Bastion Host)	특징	<ul style="list-style-type: none"> <li>• 방화벽 소프트웨어가 설치된 호스트로 보안 취약점을 방어하기 위한 시스템</li> <li>• 내부 네트워크의 전면에서 전체의 보안을 책임지는 호스트이기 때문에 공격자의 공격 목표가 되기 쉬움</li> <li>• 내부 네트워크에서 내부와 외부 네트워크의 연결점으로 사용되는 호스트</li> <li>• OSI 7계층의 네트워크계층과 전송계층 수행</li> </ul>
	장점	<ul style="list-style-type: none"> <li>• 스크린 라우터 방식보다 안전</li> <li>• 각종 로깅 정보 생성, 관리 용이</li> </ul>
	단점	<ul style="list-style-type: none"> <li>• 배스천 호스트가 손상되면 내부 네트워크 보호 불가능</li> <li>• 로그온 정보유출 시 내부 네트워크 보호 불가능</li> <li>• 데이터링크 계층 공격을 통한 방화벽 우회 공격에 취약</li> </ul>
스크리닝 라우터 (Screening Router)	특징	<ul style="list-style-type: none"> <li>• 라우터의 기본기능 이외에 패킷의 헤더를 참조하여 패킷의 통과 여부를 결정할 수 있는 필터링(스크린) 기능을 가지고 있으며, 이러한 라우터를 스크린 라우터라 함</li> <li>• OSI 7계층의 네트워크계층과 전송계층만 수행</li> </ul>
	장점	<ul style="list-style-type: none"> <li>• 필터링 속도가 빠르고 저비용</li> <li>• 하나의 스크린 라우터로 내부 네트워크 전체 보호 가능</li> </ul>
	단점	<ul style="list-style-type: none"> <li>• OSI 7계층의 네트워크 계층과 전송 계층만 방어 가능</li> <li>• 패킷 필터링 규칙을 검증하기 어려움</li> <li>• 스크린 라우터 허용/차단에 대한 패킷 기록 관리의 어려움</li> <li>• 라우터에 구현된 펌웨어만으로는 접근제어가 어려우므로, 배스천 호스트와 함께 사용</li> </ul>
듀얼 홈드 게이트웨이 (Dual Homed Gateway)	특징	<ul style="list-style-type: none"> <li>• 두 개의 랜카드(NIC)를 가진 배스천 호스트 구조</li> <li>• 두 개의 랜카드(NIC)를 가진 호스트가 내부와 외부 네트워크의 물리적인 연결을 함으로써, 두 인터페이스 사이에서 필터링 수행</li> </ul>
	장점	<ul style="list-style-type: none"> <li>• 응용서비스에 의존적이기 때문에 스크린 라우터보다 안전</li> <li>• 정보 지향적인 공격 방어 가능</li> <li>• 각종 로깅 정보를 생성 및 관리 용이</li> <li>• 설치 및 유지보수 쉬움</li> </ul>
	단점	<ul style="list-style-type: none"> <li>• 게이트웨이 손상 시 내부 네트워크 보호 불가능</li> <li>• 로그온 정보유출 시 내부 네트워크 보호 불가능</li> <li>• 제공되는 서비스가 증가할수록 프록시 소프트웨어가 많이 필요</li> </ul>
스크린 호스트 게이트웨이 (Screen Host Gateway)	특징	<ul style="list-style-type: none"> <li>• 배스천 호스트(듀얼 홈드 게이트웨이)와 스크린 라우터를 결합한 구조</li> <li>• 1차 스크린 라우터에서 패킷 필터링한 후, 2차 배스천 호스트에서 패킷 필터링</li> </ul>
	장점	<ul style="list-style-type: none"> <li>• 2단계로 필터링하기 때문에 보안성 우수</li> <li>• 네트워크계층과 응용계층 접근제어</li> <li>• 가장 많이 사용되는 시스템이며, 융통성이 좋음</li> <li>• 듀얼 홈 게이트웨이의 장점도 가짐</li> </ul>
	단점	<ul style="list-style-type: none"> <li>• 단일실패지점 장애가 발생하면 전체 네트워크 마비</li> <li>• 두 번에 걸쳐 필터링하기 때문에 속도 지연</li> <li>• 스크린 라우터의 라우팅 테이블 변경 시 방어 불가능</li> </ul>

스크린 서브넷 게이트웨이 Screen Subnet Gateway	특징	<ul style="list-style-type: none"> <li>• 스크린 호스트 게이트웨이와 듀얼 홉 게이트웨이를 결합한 구조</li> <li>• 스크린 호스트 방식의 보안상 문제점을 보완하기 위하여 내부와 외부 네트워크에 하나 이상의 경계 네트워크를 구성하여 분리한 구조</li> <li>• 스크린 라우터는 외부 네트워크와 스크린된 서브넷 그리고 내부 네트워크와 스크린된 서브넷 사이에 각각 설치하고, 입출력된 패킷을 규칙에 따라 필터링하며, 스크린 서브넷에 설치된 배스천 호스트는 프록시 서버(응용 게이트웨이)를 이용하여 허용되지 않은 모든 패킷 차단</li> </ul>
	장점	<ul style="list-style-type: none"> <li>• 다른 방화벽 구조의 장점을 지니며, 다단계 구조로 강력한 보안서비스 제공</li> <li>• 융통성 뛰어남</li> </ul>
	단점	<ul style="list-style-type: none"> <li>• 설치와 관리 어려움</li> <li>• 구축 비용 고가</li> <li>• 서비스 속도 저하</li> </ul>

## 22 침입 탐지 시스템(IDS)(탐지방법에 의한 분류)

오용 탐지 (Unused Detection)	장점	<ul style="list-style-type: none"> <li>• 침입에 사용된 특정 도구와 기술에 대한 분석 가능</li> <li>• 신속하고 정확한 침해사고 대응</li> <li>• 관리 및 보고 용이(탐지 내용 확실, 참고 자료 많음)</li> <li>• 설치 즉시 사용 가능</li> <li>• 오탐률 낮음</li> </ul>
	단점	<ul style="list-style-type: none"> <li>• 새로운 공격유형에 취약(지속적 업데이트 필요)</li> <li>• 다양한 우회 가능성 존재</li> <li>• 오탐을 줄이기 위한 세밀한 패턴 정의 필요</li> <li>• 공격에 대한 정보수집이 어려우며, 취약점에 대한 최신정보 유지 어려움</li> </ul>
이상 탐지 (Anomaly Detection)	장점	<ul style="list-style-type: none"> <li>• 새로운 공격유형에 대한 탐지 가능</li> <li>• 취약점을 사용하지 않는 권한 남용형 공격 탐지 가능</li> <li>• 이론상 완전한 침입 탐지 가능</li> </ul>
	단점	<ul style="list-style-type: none"> <li>• 주기적인 행동 프로파일 재학습 필요</li> <li>• 실제 학습환경을 만드는 그 자체가 어려움</li> <li>• 구축과 관리 어려움</li> <li>• 구현하는데 고비용</li> <li>• 오탐률 높음</li> </ul>

## 23 침입 탐지 시스템(IDS)(데이터 수집원에 따른 분류)

호스트 기반 (Host based IDS)	장점	<ul style="list-style-type: none"> <li>• 시스템에 직접 설치하여 사용하므로 네트워크 환경과는 무관</li> <li>• 시스템이 기록하는 로그 파일 정확한 침입 탐지</li> <li>• 오탐률이 낮음</li> </ul>
	단점	<ul style="list-style-type: none"> <li>• 로그 파일 변조 가능성</li> <li>• DoS 공격으로 IDS 무력화 가능성</li> <li>• 시스템 성능에 의존적이며, 서버 부하 발생 가능성</li> </ul>

네트워크 기반 (Network based IDS)	장점	<ul style="list-style-type: none"> <li>• 네트워크에 하나만 설치해도 되므로, 설치 용이</li> <li>• 서버 부하 발생 문제 없음</li> <li>• 네트워크에서 발생하는 여러 유형의 침입 탐지 가능</li> <li>• IDS 공격에 대한 방어가 가능하며, 은닉도 가능</li> </ul>
	단점	<ul style="list-style-type: none"> <li>• 패킷이 암호화되어 전송될 때 침입 탐지 불가능</li> <li>• 네트워크 트래픽이 많이 증가하므로, 성능 문제 발생</li> <li>• 오탐률이 높음</li> </ul>

## 24 침입 방지시스템(IPS)

개요	<ul style="list-style-type: none"> <li>• 넓은 범주의 통합위협관리시스템(UTM)에 속함</li> <li>• 공격에 대한 탐지만을 하는 침입 탐지시스템의 한계성을 극복한 보안 시스템</li> <li>• 시스템 및 네트워크에 대한 다양한 불법 침입행위를 실시간으로 탐지하고 분석 하여 비정상적인 패킷 인 경우 자동으로 차단하는 시스템</li> </ul>
특징	<ul style="list-style-type: none"> <li>• 손실이 발생하기 전에 대응 가능</li> <li>• 독립된 에이전트를 가지고 있음</li> <li>• 시스템 자원에 접근 어려움</li> <li>• 보안 이벤트는 메일, 메시지, 지식 프로세스의 형태로 발생</li> </ul>

## 25 허니팟(Honeypot)

목적	<ul style="list-style-type: none"> <li>• 경각심(Awareness), 정보(Information), 연구(Research)</li> <li>• 공격자를 유인하여 정보를 얻거나 찾아내기 위함 <ul style="list-style-type: none"> <li>- 정보의 수집과 시스템 제어의 기능을 충실히 수행할 수 있어야 함</li> </ul> </li> <li>• 공격의 회피 : 중요한 시스템을 보호하기 위한 위장 서버의 역할</li> </ul>
역할	<ul style="list-style-type: none"> <li>• 허니팟으로 들어오는 모든 트래픽을 감시한다는 개념</li> <li>• 일반적으로 실제 서버, 실제 운영체제, 실제처럼 보이는 자료를 기본으로 갖추고 있으며, 주된 차이점은 실제 서버와의 관계에서 시스템의 위치</li> </ul>
요구조건	<ul style="list-style-type: none"> <li>• 쉽게 공격자에게 노출되어야 함</li> <li>• 쉽게 공격이 가능한 것처럼 취약해 보여야 함</li> <li>• 시스템의 모든 구성요소를 갖추고 있어야 함</li> <li>• 시스템을 통과하는 모든 패킷을 감시해야 함</li> <li>• 시스템에 접속하는 모든 사용자에게 관리자에게 알려줘야 함</li> </ul>

## 26 IPSec VPN과 SSL VPN의 비교

	IPSec VPN	SSL VPN
OSI 계층	네트워크 계층(3계층)	세션 계층(5계층)
사용 방법	별도의 소프트웨어 설치 필요	웹브라우저 자체 지원
접근제어	애플리케이션 기반의 정교한 접근제어 미흡	애플리케이션 기반의 정교한 접근제어 가능
암호화	<ul style="list-style-type: none"> <li>• DES, AES, RC4, MD5, SHA-1</li> <li>• 패킷 단위 암호화</li> </ul>	<ul style="list-style-type: none"> <li>• DES, AES, RC4, MD5, SHA-1</li> <li>• 메시지 단위 암호화</li> </ul>
운영방식	Site to Site	Site to Remote
장점	<ul style="list-style-type: none"> <li>• 종단간 보안 가능</li> <li>• 종단 부하 없음</li> </ul>	<ul style="list-style-type: none"> <li>• 접근과 관리의 편리성</li> <li>• 클라이언트/서버 상호 인증</li> </ul>

## 27 IPSec의 운영 모드

전송 모드 (Transport Mode)	<ul style="list-style-type: none"> <li>• IPSec Header 필드가 IP와 TCP header 사이에 위치</li> <li>• IP 헤더를 제외한 IP 패킷 페이로드만을 보호</li> <li>• 종단 대 종단 전송 모드</li> <li>• 종단에 IPSec 에이전트(프로토콜 해석기)가 설치되어야 동작하므로, Peer to Peer를 원하는 경우에 유용</li> </ul>
터널 모드 (Tunnel Mode)	<ul style="list-style-type: none"> <li>• IP 패킷 전체(IP 헤더 + IP 페이로드)를 보호</li> <li>• 새로운 IPSec 헤더가 IP 패킷 전체(IP 헤더 + IP 페이로드) 앞에 추가</li> <li>• 라우터에서 IPSec 헤더를 판단한 후 제거하여 기존 패킷을 그대로 하위 단말에 전달하는 구조 <ul style="list-style-type: none"> <li>- IPSec 해석기가 단말에 설치되지 않고 라우터에 설치되어 있음</li> </ul> </li> <li>• 모든 Original IP 패킷 전체가 암호화되어 보호됨</li> </ul>

## 28 통합 기업보안 관리 시스템(ESM)의 구성요소

에이전트	방화벽이나 침입 방지 시스템 등의 장비에서 정보를 수집하여 실시간으로 수집 서버에 전송하는 시스템
수집 서버	에이전트를 통하여 수신한 정보를 수집 및 정리하여 데이터베이스 서버 및 분석 서버에 전송하는 시스템
데이터베이스 서버	수집 서버에서 수집된 정보를 데이터베이스에 저장하는 시스템
분석 서버	수집 서버에서 수신한 데이터 및 데이터베이스에 저장된 정보를 바탕으로, 네트워크의 상태 및 위기상황을 분석하고 정리하여 그 결과를 사용자에게 알려주고 데이터베이스에 저장하는 시스템

## 29 네트워크 접근통제 시스템(NAC)의 기능

내부 네트워크에 접근하는 사용자 및 단말기 제어	사용자 인증, IP 주소 관리, 소프트웨어 패치 등
네트워크의 안정성 확보	비인가 장비 및 서비스 탐지, 네트워크 우회경로 탐지
정책 기반 네트워크 관리	내부 사용자의 보안정책 준수 통제
비정상 트래픽 탐지 및 차단	내부 네트워크 악성 트래픽 탐지

## 30 통합위협관리시스템(UTM)

개요	<ul style="list-style-type: none"> <li>• 하나의 장비에 여러 보안 기능을 탑재한 장비를 통칭하며, 통합위협관리시스템을 의미</li> <li>• 기존의 다양한 보안 솔루션(방화벽, IDS, IPS, VPN, 바이러스 필터링, 콘텐츠 필터링 등)의 보안 기능을 하나로 통합한 기술과 장비             <ul style="list-style-type: none"> <li>- 최근에 구축되는 보안장비는 대부분 통합위협관리시스템(UTM)에 속함</li> </ul> </li> </ul>
기능	<ul style="list-style-type: none"> <li>• 침입 방지시스템(IPS)</li> <li>• 침입 탐지시스템(IDS)</li> <li>• 침입 차단시스템(Firewall)</li> <li>• 가상 사설망(VPN)</li> <li>• 안티바이러스(Anti-Virus) · 안티 스팸(Anti-SPAM)</li> <li>• 웹 콘텐츠 필터링(Web Contents Filtering)</li> <li>• 무선랜 보안(Wireless LAN Security)</li> </ul>



## 1 FTP 운영모드

능동모드 (Active Mode)	<ul style="list-style-type: none"> <li>• 서버가 20번 포트로 클라이언트의 1024 이후의 포트로 접속을 요청하는 방식</li> <li>• PORT 명령 사용 : FTP Bounce 공격, 포트 스캐닝 공격에 이용될 수 있음</li> <li>• 클라이언트에 방화벽이나 NAT 설치 시 정상적인 서비스 불가</li> </ul>
수동모드 (Passive Mode)	<ul style="list-style-type: none"> <li>• 서버에서 클라이언트로 접속하는 모순을 해결하기 위해 고안된 방식</li> <li>• 제어포트 연결단계는 액티브 모드와 동일하나 데이터 접속에서 서버의 20번 포트를 사용하지 않고 1,024 포트 이후의 임의의 포트로 클라이언트가 먼저 접속을 시도하는 방식</li> <li>• 서버에 방화벽이나 NAT 설치 시 정상적인 서비스 불가</li> </ul>

## 2 FTP 공격유형

FTP Bounce	FTP 프로토콜의 취약점을 이용한 공격으로, FTP 서버가 데이터를 전송할 때 목적지를 검사하지 않는다는 점을 이용한 공격
Anonymous FTP	<ul style="list-style-type: none"> <li>• /bin 디렉터리의 권한을 잘못 설정할 경우 /bin 디렉터리의 특정 프로그램을 변경하거나 업로드하여 시스템에 침입</li> <li>• .rhosts나 .forward 파일 등을 생성하여 다른 호스트를 통한 신뢰 관계 설정 이용</li> <li>• 미러 사이트(Mirror Site)인 경우, 주 사이트의 프로그램에 특정 코드를 삽입함으로써, 원격지 사이트 공격</li> </ul>
TFTP	<ul style="list-style-type: none"> <li>• 인증 절차를 요구하지 않아, 누구나 호스트에 접근할 수 있으므로 파일 유출 가능</li> <li>• TFTP의 취약점을 이용한 파일의 전송이나 복제 공격</li> <li>• TFTP는 웹에 의해 악용되고 있으며, 블래스터 같은 웜들은 다른 컴퓨터를 감염시키는 데 TFTP를 악용</li> </ul>

## 3 FTP 보안 취약점

FTP 프로토콜	FTP 세션 자체는 암호화되어 있지 않아 스니핑 공격에 취약
익명 FTP	<ul style="list-style-type: none"> <li>• 특정 패스워드를 요구하지 않으며, 모든 사용자가 같은 계정을 사용할 수 있음</li> <li>• 모든 사용자에게 접속이 허용되기 때문에 쓰기 권한이 통제되지 않을 경우, 불법 파일의 업로드 등을 통한 공격에 노출될 취약점 존재</li> </ul>
무차별 대입 공격	임의의 계정으로 로그인 시도를 반복적으로 수행함으로써, 실제 시스템에 존재하는 계정의 패스워드를 유추>Password Guessing)할 수 있는 취약점 존재
FTP Bounce 공격	포트가 20번과 21번으로 분리된 것을 이용함으로써, 명령 채널을 통해 파일 전송을 요청한 클라이언트와 데이터 채널을 통해 실제 파일을 수신하는 클라이언트가 분리될 수 있는 취약점 존재

#### 4 메일(Mail) 서비스 공격유형

<b>액티브 콘텐츠 공격</b> (Active Contents Attack)	<ul style="list-style-type: none"> <li>• 불특정 다수에게 메일을 유포하는 방식이 아닌 특정 집단을 대상으로 사전에 수집된 이메일 정보를 이용하여 악의적인 문서 파일을 첨부하여 열람하도록 유도한 후 취약성이 있는 시스템을 장악하는 공격</li> <li>• 메일 열람 시 HTML 기능이 있는 메일 클라이언트나 웹 브라우저를 사용하는 사용자를 대상으로 하는 공격기법</li> </ul>
<b>트로이목마 공격</b> (Trojan Horse Attack)	<ul style="list-style-type: none"> <li>• 일반 사용자가 트로이목마 프로그램을 실행시켜, 시스템에 접근할 수 있는 백도어를 만들거나 시스템에 피해를 줌</li> <li>• 사용자가 트로이목마를 실행시키도록 유도하기 위해 사회공학적인 방법을 사용하며, 사용자가 첨부파일을 실행하도록 유도</li> </ul>
<b>버퍼 오버플로 공격</b> (Buffer Overflow Attack)	<p>공격자가 조작된 메일을 전송함으로써, 공격대상의 컴퓨터에서 임의의 명령을 실행하거나 트로이목마 같은 악성 프로그램을 설치하도록 함</p>

#### 5 메일(Mail) 보안 프로토콜의 특징

PGP	PEM	PGP/MIME	S/MIME
<ul style="list-style-type: none"> <li>• 필 짐머만(Phill Zimmerman)이 개발</li> <li>• 분산화된 키 인증</li> <li>• 구현 용이</li> <li>• 일반 용도의 보안성</li> <li>• 대중적으로 사용</li> </ul>	<ul style="list-style-type: none"> <li>• IETF 표준안</li> <li>• 중앙 집중화된 키 인증</li> <li>• 구현 어려움</li> <li>• 높은 보안성</li> <li>• 대중적이지 못함</li> </ul>	<ul style="list-style-type: none"> <li>• 메일 메시지 표준(MIME) 기반</li> <li>• PGP 암호기법+메일 시스템</li> <li>• X.509 인증서 지원 안 됨</li> </ul>	<ul style="list-style-type: none"> <li>• RSA Data Security Inc 개발</li> <li>• 메일 메시지 표준(MIME) 기반</li> <li>• 다양한 상용 툴킷(Tool Kit)</li> <li>• X.509 인증서 지원</li> </ul>

#### 6 PGP와 S/MIME의 기능

	PGP	S/MIME
기능	메시지 암호화(기밀성), 전자서명(인증), 압축, 분할 및 재결합	메시지 암호화, 전자서명, 압축

## 7 캐시, 쿠키, 세션의 차이점

캐시 (Cache)	<ul style="list-style-type: none"> <li>• 웹페이지 요소를 저장하기 위한 임시 저장소, 특히, 나중에 필요할 것 같은 요소들을 저장</li> <li>• 웹페이지가 빠르게 렌더링할 수 있도록 도와줌. 그림 파일이나 문서 파일 등의 요소들이 있음</li> <li>• 사용자가 직접 수동으로 삭제해주어야 함</li> </ul>
쿠키 (Cookie)	<ul style="list-style-type: none"> <li>• 기본적으로 웹서버에서 PC로 보내는 작은 파일들을 저장</li> <li>• 보통 쿠키는 누군가 특정한 웹사이트를 접속할 때 발생</li> <li>• 사용자의 인증을 도와줌</li> <li>• HTTP의 비연결(Connectionless)과 무상태(Stateless)를 보완</li> </ul>
세션 (Session)	<ul style="list-style-type: none"> <li>• 웹 브라우저를 통해 서버에 접속한 이후부터 브라우저를 종료할 때까지 유지되는 상태</li> <li>• 서버에 직접 저장되므로, 세션 내의 데이터를 탈취하는 것은 어려움(보안성 높음)</li> </ul>

## 8 HTTP 상태 코드

### (1) 트랜잭션이 성공한 경우(2xx)

코드	상태	설명
200	OK	요청이 성공적으로 완료
201	Created	요청이 POST이며, 성공적으로 완료
202	Accepted	요청이 서버에 전달되었으나, 처리결과를 알 수 없음
203	Non Authoritative Information	GET 요청이 실행되었으며, 부분적인 정보를 리턴하였음
204	Not Content	요청이 실행되었으나, 클라이언트에게 전송할 데이터가 없음

### (2) 트랜잭션의 리다이렉션(3xx)

코드	상태	설명
300	Multiple Choices	요청이 여러 위치에 존재하는 자원을 필요로 하므로 응답은 이에 대한 정보 전송
301	Moved Permanently	요청 데이터는 영구적으로 새로운 URL로 옮겨짐
302	Found	요청 데이터를 발견하였으나, 실제 다른 URL에 존재
303	Not Modified	If Modified Since 필드를 포함한 GET 메소드를 수신했으나, 문서는 수정되지 않음

### (3) 오류 메시지(4xx)-Client

코드	상태	설명
400	Bad Request	요청 문법 오류
401	Unauthorized	요청이 서버에게 Authorization 필드를 사용하였으나 값을 지정하지 않았음
403	Forbidden	요청이 허용되지 않은 자원 요구
404	Not Found	서버는 요구된 URL을 찾을 수 없음

### (4) 오류 메시지(5xx)-Server

코드	상태	설명
500	Internal Server Error	서버 내부에 오류가 발생하여 더 이상 진행할 수 없음
501	Not Implemented	요청은 합법적이나, 서버는 요청된 메소드(Method)를 지원하지 않음
503	Service Unavailable	서버 과부하로 서비스를 할 수 없음
504	Gateway Timeout	502의 오류와 유사하나, 보조 서버의 응답이 너무 오래 지체되어 처리 실패

## 9 OWASP Top 10(2017)

A1 (인젝션)	<ul style="list-style-type: none"> <li>SQL, OS, XXE, LDAP 인젝션 취약점은 신뢰할 수 없는 데이터가 명령어나 쿼리문의 일 부분으로써, 인터프리터로 보내질 때 발생</li> <li>공격자의 악의적인 데이터는 예기치 않은 명령을 실행하거나 올바른 권한 없이 데이터에 접근하도록 인터프리터를 속일 수 있음</li> </ul>
A2 (취약한 인증)	인증 및 세션 관리와 관련된 애플리케이션 기능이 종종 잘못 구현되어 공격자들이 암호, 키, 세션 토큰을 위험에 노출될 수 있거나 일시적 또는 영구적으로 다른 사용자의 권한 획득을 위해 구현상 결함을 악용하도록 허용
A3 (민감한 데이터 노출)	<ul style="list-style-type: none"> <li>다수의 웹 애플리케이션과 API는 금융 정보, 건강 정보, 개인 식별 정보와 같은 중요한 정보를 제대로 보호하지 않음</li> <li>공격자는 신용카드 사기, 신분 도용 또는 다른 범죄를 수행하기 위해 보호가 취약한 데이터를 훔치거나 수정할 수 있음</li> <li>중요한 데이터는 저장 또는 전송할 때 암호화 같은 추가 보호 조치가 없으면 탈취당할 수 있으며, 브라우저에서 주고받을 때 각별한 주의가 필요함</li> </ul>
A4 (XML 외부 개체(XXE))	<ul style="list-style-type: none"> <li>오래되고 설정이 잘못된 많은 XML 프로세서들은 XML 문서 내에서 외부 개체 참조를 평가함</li> <li>외부 개체는 파일 URI 처리기, 내부 파일 공유, 내부 포트 스캔, 원격 코드 실행과 서비스 거부 공격을 사용하여 내부 파일을 공개하는데 사용할 수 있음</li> </ul>
A5 (취약한 접근통제)	<ul style="list-style-type: none"> <li>인증된 사용자가 수행할 수 있는 작업에 대한 제한이 제대로 적용되어 있지 않음</li> <li>공격자는 이러한 결함을 악용하여 다른 사용자의 계정에 접근하거나, 중요한 파일을 보거나, 다른 사용자의 데이터를 수정하거나, 접근 권한을 변경하는 등 권한 없는 기능과 데이터에 접근할 수 있음</li> </ul>

A6 (잘못된 보안 구성)	<ul style="list-style-type: none"> <li>• 잘못된 보안 구성은 가장 흔하게 보이는 이슈</li> <li>• 취약한 기본 설정, 미완성(또는 임시 설정), 개방된 클라우드 스토리지, 잘못 구성된 HTTP 헤더 및 민감한 정보가 포함된 잘못된 에러 메시지로 인한 결과</li> <li>• 모든 운영체제, 프레임워크, 라이브러리와 애플리케이션을 안전하게 설정해야 할 뿐만 아니라 시기적절하게 패치/업그레이드를 진행해야 함</li> </ul>
A7 (크로스 사이트 스크립팅(XSS))	<ul style="list-style-type: none"> <li>• 애플리케이션이 올바른 유효성 검사 또는 필터링 처리 없이 새 웹 페이지에 신뢰할 수 없는 데이터를 포함하거나, 자바스크립트와 HTML을 생성하는 브라우저 API를 활용한 사용자 제공 데이터로 기존 웹 페이지를 업데이트할 때 발생</li> <li>• 피해자의 브라우저에서 공격자에 의해 스크립트를 실행시켜 사용자 세션을 탈취할 수 있게 만들고, 웹사이트를 변조시키고, 악성 사이트로 리다이렉션할 수 있도록 허용</li> </ul>
A8 (안전하지 않은 역 직렬화)	<ul style="list-style-type: none"> <li>• 안전하지 않은 역 직렬화는 종종 원격 코드 실행으로 이어짐</li> <li>• 역 직렬화 취약점이 원격 코드 실행결과를 가져오지 않더라도 이는 권한 상승 공격, 주입 공격과 재생 공격을 포함한 다양한 공격 수행에 사용될 수 있음</li> </ul>
A9 (알려진 취약점이 있는 구성요소 사용)	<ul style="list-style-type: none"> <li>• 라이브러리, 프레임워크 및 다른 소프트웨어 모듈 같은 컴포넌트는 애플리케이션과 같은 권한으로 실행됨</li> <li>• 만약에 취약한 컴포넌트가 악용된 경우, 이는 심각한 데이터 손실을 일으키거나 서버가 장악됨</li> <li>• 알려진 취약점이 있는 컴포넌트를 사용한 애플리케이션과 API는 애플리케이션 방어를 약화하거나 다양한 공격에 영향을 미침</li> </ul>
A10 (불충분한 로깅 및 모니터링)	<ul style="list-style-type: none"> <li>• 불충분한 로깅과 모니터링은 사고 대응의 비효율적인 통합 또는 누락과 함께 공격자들이 시스템을 더 공격하고, 지속성을 유지하며, 더 많은 시스템을 중심으로 공격할 수 있도록 만들고, 데이터를 변조, 추출 또는 파괴할 수 있음</li> <li>• 대부분의 침해 사례에서 침해를 탐지하는 시간이 200일이 넘게 걸리는 것을 보여주고, 이는 일반적으로 내부 프로세스와 모니터링보다 외부기관이 탐지함</li> </ul>

## 10 SQL Injection

공격기법	<ul style="list-style-type: none"> <li>• 공격자가 입력값을 조작하여 원하는 SQL 구문을 실행하는 기법</li> <li>• 잠재적인 SQL 구문의 구조 확인 후 적절히 실행되는 문자의 결합을 찾을 때까지 입력을 조작하는 기법</li> <li>• 부적절한 입력값을 전달하여 오류를 발생시키고, SQL 구문을 확인하는 방법</li> <li>• MS-SQL에서의 시스템 명령어 실행 : xp_cmdshell 저장 프로시저를 이용한 시스템 명령어 실행</li> </ul>
대응방법	<ul style="list-style-type: none"> <li>• 사용자의 입력에 특수문자가 포함되어 있는지 검증</li> <li>• SQL 서버의 오류 메시지 숨김</li> <li>• 일반 사용자 권한으로 시스템 저장 프로시저에 접근 차단</li> </ul>

## 11 XSS와 CSRF의 비교

구분	XSS	CSRF
주 공격지점	클라이언트	서버
기능 구현	공격자가 Script를 이용하여 직접 구현	서버에서 제공하는 기능을 도용
Script 사용 여부	반드시 Script 사용이 가능해야 함	Script를 사용할 수 없어도 공격 가능
공격 시 준비사항	XSS 취약점만 발견 후 즉시 사용 가능	공격하고자 하는 요청(Request)/응답(Response)의 로직을 분석해야 함
공격 감지 가능 여부	저장(Stored)/반사(Reflective)	구분할 수 없음

## 12 DNS 공격기법과 대응방법

공격기법	DNS Spoofing	공격대상자에게 전달되는 DNS 응답을 위조하거나 DNS 서버에 위조된 IP 주소가 저장되게 하여 희생자가 의도하지 않은 주소로 접속하게 하는 공격기법
	DNS Cache Poisoning	<ul style="list-style-type: none"> <li>취약한 DNS 서버에 조작된 요청을 전송하여 DNS 서버가 저장하고 있는 주소 캐시 정보를 임의로 변조하여 조작된 사이트로 접속하게 됨</li> <li>DNS 질의 요청 시 출발지와 목적지 포트, 요청처리 ID(Transaction ID)를 부여할 때, 임의의 값이 생성되는 것을 예측할 경우 캐시 조작이 가능하다는 취약점 존재</li> </ul>
대응방법	DNSSEC	<ul style="list-style-type: none"> <li>DNS 데이터 대상의 “데이터 위조 · 변조 공격”을 방지하기 위한 인터넷 표준기술</li> <li>DNS 데이터의 위조 · 변조 가능성을 원천적으로 차단하기 위해 공개키 암호방식의 전자서명 기술을 DNS 체계에 도입 적용</li> <li>기술적으로 DNS Cache Poisoning 공격에 대응하는 솔루션으로 나온 것이 DNSSEC (DNS Security Extension) <ul style="list-style-type: none"> <li>DNS 보안 취약점을 해결하기 위해서는 DNSSEC을 구현해야 함</li> </ul> </li> </ul>

## 13 DB 보안 통제

흐름통제	<ul style="list-style-type: none"> <li>접근 가능한 객체 간의 정보 흐름을 조정</li> <li>높은 보안등급에서 낮은 등급으로 객체 전송 시 정보의 흐름에서 기밀성이 위반되지 않도록 조정</li> </ul>
추론통제	<ul style="list-style-type: none"> <li>사용자 X를 찾을 후 <math>Y=f(x)</math>를 통하여 Y를 유도한 후 간접 접근을 통한 추론</li> <li>보이는 데이터 집합 X에 대한 질의와 Y에 대한 조건을 통하여 데이터 집합 Y에 대한 정보 습득</li> <li>상관 데이터 : 보이는 데이터 A가 보이지 않는 데이터 B와 의미적으로 연결될 때 A를 통하여 B를 추론</li> <li>추론통제 해결방법 : 비밀 데이터의 암호화, 사용자의 데이터 지식 추적, 데이터 위장 등</li> </ul>
접근통제	<ul style="list-style-type: none"> <li>인증된 사용자에게 허가된 범위 내에서 시스템 내부의 정보에 대한 접근을 허용하는 기술적 방법</li> <li>사용자가 DB 접근 시 접근 권한을 검사하여 허용 여부 결정</li> </ul>
허가규칙	정당한 절차를 통하여 DBMS에 로그인한 사용자라 하더라도 허가받지 않은 데이터에 접근을 통제하기 위한 규칙
가상테이블	전체 DB 중 자신이 허가받은 사용자 관점만 볼 수 있도록 한정
암호화	불법적인 데이터 접근을 허용하더라도 내용을 알 수 없는 형태로 변형시킴

#### 14 SET과 SSL 프로토콜의 비교

구분	SET	SSL/TLS
기능	전자 지불 프로토콜	통신 보안 프로토콜
상호 운용성	SET 규격에 맞으면 보장	지불관련 상세 규약 없음
온라인 결제	제공함	제공하지 않음
전자지갑	반드시 전자지갑 사용	전자지갑 개념 없음
부인방지	서명 기능으로 제공	서명 기능이 없어 제공하지 않음
안전성	높음 (금융기관만 카드번호 확인)	다소 낮음 (상점에 카드번호 노출)
기타	시스템 구현이 복잡하고, SSL에 비해 느림	사용 간편, 고속, 시스템 구현이 SET에 비해 간단

#### 15 S-HTTP와 SSL/TLS의 비교

	S-HTTP	SSL/TLS
배경	1994년 EIT, NCSA, RSA에 의하여 HTTP의 안전성 확보를 위해 개발	1993년 웹서버와 웹브라우저 간 안전한 통신을 위해 넷스케이프사에서 개발
동작 계층	응용 계층	전송 계층
용도	웹에서만 사용	FTP, Telnet 등에서 사용
인증방식 및 인증서	<ul style="list-style-type: none"> <li>• 각각의 인증서 필요</li> <li>• 클라이언트에서 인증서를 보낼 수 있음</li> </ul>	<ul style="list-style-type: none"> <li>• 클라이언트의 인증이 선택적</li> <li>• 서버에서만 인증</li> </ul>
인증 단위	메시지 단위	서비스 단위
지시자	shttp://	https://
문제점	대중화되지 못함	<ul style="list-style-type: none"> <li>• 미국의 정책에 의하여 국외 판매제품 512비트 RSA Public Key와 40비트 RC2 Single Key 제한</li> <li>• 외국산 알고리즘 사용 시 보안 문제</li> </ul>

#### 16 SSL/TLS 프로토콜

Handshake	암호 알고리즘 결정, 키 분배, 서버 및 클라이언트 인증을 수행하기 위해 사용되는 프로토콜
Change Cipher Spec	암호화 알고리즘과 보안정책을 송수신 측간에 조율하기 위해 사용하는 프로토콜
Alert	SSL/TLS 수행 중 발생하는 오류 메시지를 제공하는 프로토콜
Record	데이터의 압축을 수행하여 안전한 TCP 패킷으로 변환하고, 데이터 암호화 및 무결성을 위한 메시지 인증을 수행하는 프로토콜

## 17 OTP 인증방식

시간 동기화 방식	특징	<ul style="list-style-type: none"> <li>가장 많이 사용하는 방식</li> <li>해시함수의 입력으로 비밀 값과 현재의 시간(실시간)을 입력하는 방식</li> <li>서버와 클라이언트는 시간이라는 공통된 값을 공유함으로써 동기화시킬 수 있다는 것에서 착안했으며, 토큰의 시간을 함부로 변경할 수 없게 만드는 것이 중요</li> </ul>
	장점	<ul style="list-style-type: none"> <li>토큰에 PIN을 입력하는 절차가 없으므로 사용 간편</li> <li>OTP 유효시간이 짧으므로 유출되어도 대체로 안전</li> <li>사용하기 편리하고 보안성이 우수하여 가장 많이 사용</li> </ul>
	단점	<ul style="list-style-type: none"> <li>토큰의 분실을 대비하여 일회용 패스워드는 사용자의 PIN, 토큰에서 생성된 응답 값으로 구성. 이 경우 네트워크에 PIN이 전송된다는 것과 PIN이 항상 서버에 등록되어 있어야 하므로 변경이 불편</li> <li>토큰은 1분에 한 번씩 패스워드가 바뀌게 되고, 서버와 클라이언트 간 시간의 오차가 생길 수 있으며, 그 오차의 보정이 이 방식의 핵심</li> <li>OTP 토큰 시간이 정확해야 하므로 구현하기 어려움</li> </ul>
비동기화 방식 (이벤트 방식)	특징	<ul style="list-style-type: none"> <li>응답 값을 생성할 때 해시함수의 입력으로 비밀 값과 특정한 사건이 일어난 횟수 등을 함께 사용하는 방법</li> <li>질의/응답 방식과 비교하면 질의 값으로 특정 사건을 수치화한 것을 사용한다고 볼 수 있으며, 서버에서 질의 값을 수신할 필요가 없고, 질의/응답 방식의 단점을 보완한 것</li> <li>OTP 기기의 버튼을 누를 때마다 OTP 값이 생성</li> </ul>
	장점	<ul style="list-style-type: none"> <li>가장 간단한 구조로 구현하기 쉬움</li> <li>사용하기 편리</li> </ul>
	단점	<ul style="list-style-type: none"> <li>생성될 응답 값 예측 가능</li> <li>인증 서버와 OTP 기기의 이벤트 값을 자주 일치시켜야 하는 불편</li> <li>OTP 값의 유효시간이 정해지지 않아 유출 시 보안 취약</li> </ul>
질의/응답 (Challenge/Response) 방식	특징	<ul style="list-style-type: none"> <li>난수를 암호 알고리즘의 입력값으로 사용하여, 일회용 패스워드를 생성</li> <li>금융기관에서 사용하는 보안카드가 질의/응답 방식 중의 하나</li> <li>OTP 기기에 질의 값을 버튼으로 입력해야 하므로, 사용이 불편하여 거의 사용되지 않음</li> </ul>

## 18 SSO, EAM, IAM의 비교

	SSO	EAM	IAM
목적	중앙집중식 ID 관리	ID 관리와 권한, 자원정책의 결합	기존 EAM에 자동적 권한 부여 및 관리 기능
관련 기술	PKI, LDAP	SSO, AC, LDAP, PKI, 암호화	통합자원관리와 Provisioning 기능
장점	<ul style="list-style-type: none"> <li>단일 ID로 사용의 편리성</li> <li>인증정책과 권한설정 용이</li> </ul>	<ul style="list-style-type: none"> <li>자원접근 시 권한까지 제어</li> <li>개별 응용 레벨의 권한 제어</li> </ul>	자동화된 자원 관리로 확장성 용이
단점	<ul style="list-style-type: none"> <li>ID 및 비밀번호 노출 시 전체 시스템 위험</li> <li>자원별 권한 관리 약함</li> </ul>	<ul style="list-style-type: none"> <li>사용자 및 자원별 권한 관리 어려움</li> <li>구축비용 고가</li> <li>구현의 복잡성</li> </ul>	EAM에 비해 구현의 복잡성



## 19 스테가노그래피, 워터마킹, 핑거프린팅

	스테가노그래피	워터마킹	핑거프린팅
은닉 정보	메시지	판매자 정보	구매자 추적 정보
목적	은닉 메시지 검출	저작권 표시	구매자 추적
트래킹	불가	가능	가능
불법 예방	하	중	상
저작권 증명	하	중	상
공격 내성	약함	강함	강함

## 20 DRM의 구성 요소

메타 데이터 (Meta Data)	콘텐츠 생명주기 내에서 관리되어야 할 각종 데이터의 구조 및 정보
패키저 (Packager)	메타 데이터와 함께 시큐어 컨테이너 포맷으로 패키징하는 모듈
시큐어 컨테이너 (Secure Container)	DRM의 범위 내에서 유통되는 콘텐츠 배포의 단위
식별자 (Identifier)	콘텐츠를 식별하기 위한 식별자
DRM 제어기 (DRM Controller)	콘텐츠를 사용하는 사용자의 컴퓨터 또는 디바이스 플랫폼에서 콘텐츠가 지속적으로 보호될 수 있도록 프로세스를 제어
클리어링 하우스 (Clearing House)	디지털 허가를 사용자에게 발급해 주고, 콘텐츠 사용자에게 로열티 수수료를 지불하며, 배급자에게 해당되는 배급 수수료를 지불하는 재정적 거래를 취급

## 04

## 정보보안 일반

## 1 사용자 인증 시 보안 요구사항

식별 (Identification)	<ul style="list-style-type: none"> <li>시스템에 주체의 식별자를 요청하는 과정</li> <li>각 시스템의 사용자는 시스템이 식별할 수 있는 유일한 식별자(ID)를 가짐</li> <li>개인 식별자는 유일한 것을 사용해야 하며, 공유해서는 안 됨</li> </ul>
인증 (Authentication)	<ul style="list-style-type: none"> <li>임의의 정보에 접근할 수 있는 주체의 능력이나 자격을 검증하는 과정</li> <li>시스템의 부당한 사용이나 정보의 부당한 전송을 방지할 수 있음</li> </ul>
인가 (Authorization)	<ul style="list-style-type: none"> <li>누구에게 무엇을 할 수 있게 하거나, 소유할 수 있는 권한을 부여하는 것</li> <li>사용자, 프로그램, 프로세스에 권한을 부여하는 것</li> </ul>
책임 추적성 (Accountability)	<ul style="list-style-type: none"> <li>다중 작업이 지원되는 네트워크 환경에서 누가, 언제, 어떤 행동을 하였는지 기록</li> <li>필요 시 그 행위자를 추적하여 책임소재를 명확히 할 수 있는 기반</li> </ul>

## 2 사용자 인증의 종류

Type I	지식기반 (What you Know)	개인식별번호(PIN), 패스워드, 패스 프레이즈(Pass phrase), 계좌번호 등
Type II	소유기반 (What you Have)	IC 카드(Smart Card), 마그네틱 카드(Memory Card), 일회용 패스워드(OTP) 단말기, 토큰, 열쇠, 운전면허증, 여권 등
Type III	신체기반 (What you Are)	지문, 얼굴, 음성, 홍채, 망막, 정맥, 서명 등
Type IV	행동기반 (What you Do)	키스트로크(Keystroke), 서명(Signature), 음성(Voice)

## 3 생체 인증 기술의 평가 항목

보편성	모든 사람이 가지고 있는 생체 특성인가?
유일성	동일한 특징을 가진 다른 사람은 없는가?
영구성	시간에 따른 변화가 없는 생체 특성인가?
획득성	정량적으로 측정이 가능한 생체 특성인가?
성능	환경 변화와 무관하게 높은 정확성을 얻을 수 있는가?
수용성	사용자의 거부감은 없는가?
반기만성	고의적인 부정 사용으로부터 안전한가?

#### 4 사용자 인증의 비교

생체인식 기술	장점	단점
지문(Fingerprinting)	<ul style="list-style-type: none"> <li>비용 저렴</li> <li>우수한 안정성</li> </ul>	지문이 보이지 않거나, 손상될 가능성
안면(Face)	쉽고, 빠르고, 비용 저렴	조명 및 자세에 따라 영향을 받고, 정확도 낮음
장문/손모양 (Palm/Hand Geometry)	최소의 저장용량 요구	처리속도가 느리고, 정확도 떨어짐
홍채(Iris)	위조 불가능	대용량 특징 벡터(256바이트)
망막(Retina)	안정성 우수	사용 거부감
음성(Voiceprint)	비용 저렴, 원격접근에 적당	처리속도 느리고, 사람 상태에 쉽게 영향
서명(Signature)	비용 저렴	사람 상태에 쉽게 영향을 받고, 정확도 낮음
DNA	안정성, 정확성 우수	거부감, 즉시성 떨어짐

#### 5 커버로스(Kerberos)

특징	<ul style="list-style-type: none"> <li>Needham과 Schroeder의 신뢰할 수 있는 제3자 프로토콜에 근거한 모델로, 중앙의 인증서버가 네트워크의 모든 실체와 서로 다른 비밀키를 공유하고, 그 비밀키를 알고 있는 것으로 실체의 신원을 증명하며, 특히 패스워드를 네트워크에 노출시키지 않음</li> <li>클라이언트 서버 모델을 목적으로 개발되었으며, 사용자와 서버가 서로 식별할 수 있는 상호 인증(양방향 인증)을 제공</li> <li>커버로스 프로토콜의 메시지는 도청과 재전송 공격으로부터 보호</li> <li>티켓에 포함되는 정보 : 서버 ID, 클라이언트 ID, 클라이언트 네트워크 주소, 티켓 유효 기간, 클라이언트와 서버가 서비스 기간 동안 공유하는 세션키</li> </ul>
장점	<ul style="list-style-type: none"> <li>사용자 간 메시지를 암호화 키 및 암호 프로세스를 이용하여 보호하기 때문에 데이터의 기밀성과 무결성을 보장</li> <li>타임스탬프를 이용하므로, 재생 공격 방지</li> <li>이 기종 간 자유로운 서비스 인증(SSO)</li> <li>대칭키를 사용하여 도청으로부터 보호</li> </ul>
단점	<ul style="list-style-type: none"> <li>모든 사용자의 암호화 키를 키 분배센터가 가지고 있으므로, 키 분배센터가 단일 실패 지점(Single Point of Failure)이 되어 오류가 발생하면 전체 서비스 마비</li> <li>사용자의 비밀키가 사용자의 시스템에 임시로 저장되기 때문에 사용자 시스템의 정보 유출 가능성</li> <li>사용자 세션키도 사용자 시스템에 임시저장되기 때문에 취약</li> <li>사용자가 패스워드를 변경하면 비밀키도 변경해야 하는 번거로움</li> <li>패스워드 사전 공격과 패스워드 추측 공격에 취약</li> <li>모든 클라이언트/서버와의 시간 동기화 필요</li> <li>UDP 기반으로 방화벽에서 자주 차단</li> <li>커버로스 서버 자체의 보안 문제</li> </ul>

## 6 커버로스의 구성요소

KDC (Key Distribution Center)	<ul style="list-style-type: none"> <li>• TGS와 AS로 구성</li> <li>• 모든 사용자와 서비스의 비밀키를 보관</li> <li>• 신뢰할 수 있는 제3의 기관으로서 티켓을 생성, 인증 서비스 제공</li> </ul>
AS (Authentication Service)	<ul style="list-style-type: none"> <li>• 실질적인 인증 수행</li> <li>• 사용자에게 인증을 수행하는 KDC의 부분 서비스</li> </ul>
TGS (Ticket Granting Service)	티켓을 부여하고, 티켓을 분배하는 KDC의 부분 서비스
Ticket	<ul style="list-style-type: none"> <li>• 사용자의 신원을 확인하는 토큰</li> <li>• 사용자가 통신할 때마다 패스워드를 입력하지 않도록 도와줌</li> </ul>

## 7 디바이스 인증기술의 특징

보안성	<ul style="list-style-type: none"> <li>• 민간 기업 등에서 기기 서비스를 할 경우 위협요소 및 대응방안이 디바이스 인증체계에서 제시될 수 있음</li> <li>• 기기인증서 발급부터 서비스까지 절차적으로 검증된 보안 수준을 구축하여 보안 취약점 제거</li> </ul>
경제성	<ul style="list-style-type: none"> <li>• 일관된 보안정책 및 안정성을 마련하여 구축 및 운영 비용 절감</li> <li>• 스마트그리드를 통한 전력 요금의 실시간 확인에 의한 요금 절감 및 에너지 절약 효과, 인터넷 전화를 이용한 통신비 절감 효과</li> </ul>
상호 연동성	<ul style="list-style-type: none"> <li>• 유비쿼터스 환경에서 디바이스의 서비스 간, 기종 간 통신 및 인증 필요</li> <li>• 단일 보안 프레임워크 기반 상호 연동성 보장이 가능한 인증체계 도입 필요</li> </ul>

## 8 공동 인증서와 디바이스 인증서의 비교

구분	공동 인증서	디바이스 인증서
발급대상	사용자	디바이스 및 애플리케이션
발급대상 확인	사용자 신원확인 등록	<ul style="list-style-type: none"> <li>• 디바이스의 고유 식별값(예 ICC ID, MIM)으로 확인</li> <li>• 모든 디바이스에 적용(대상 인증 없음)</li> </ul>
키 생성 주체	사용자	디바이스, 제조사, 인증 시스템
인증서 발급	온라인을 통한 일대일 발급	온라인 외에 다량의 인증 서버 발급
인증서 관리기능	폐지, 재발급, 효력정지, 복구	디바이스 폐기에 따른 인증서 폐지, 자동갱신
공개키 알고리즘	RSA, ECC	RSA, ECC

## 9 접근통제의 분류

물리적 통제 (Physical Control)	차단막, 자물쇠, CCTV, 센서, 경보, 생체인식 장치 등
관리적 통제 (Manage Control)	정책 · 수행, 직무 분리, 사후체크 등
기술적 통제 (Technical Control)	암호화, 접근통제 소프트웨어, 패스워드, 스마트카드, IDS 등

## 10 MAC, DAC, RBAC의 비교

	MAC	DAC	RBAC
권한 부여 방법	주체와 객체의 등급을 비교하여 접근 권한을 부여	주체의 신분에 따라 접근 권한 부여	주체와 객체 간 역할을 부여하여 임의적, 강제적 접근통제의 단점을 보완한 방식
접근 권한 주체	시스템	객체 소유자	중앙 권한
접근 여부 기준	보안 레이블	신분(ID)	역할(Role)
정책	경직	유연	유연
TCSEC Level	B-Level	C-Level	C-Level
장점	중앙집중관리	구현 용이	다양한 접근 권한
단점	구현, 비용, 성능 문제	신분 위장	x

## 11 접근통제 모델의 비교

벨-라파둘라 (Bell-Lapadula) 모델	<ul style="list-style-type: none"> <li>1973년 미국 MITRE 연구소에서 Bell과 Lapadula가 개발하였으며, 최초의 수학적 모델</li> <li>정보가 하위에서 상위로 흐른다(Bottom-Up)는 개념을 적용한 모델</li> <li>TCSEC 기반 기밀성 모델, 강제적 접근통제 모델, 군사적 모델</li> </ul>
비바 (Biba) 모델	<ul style="list-style-type: none"> <li>Bell-Lapadula 모델의 단점인 무결성을 보완한 최초의 수학적 모델</li> <li>정보가 상위에서 하위로 흐른다(Top-Down)는 개념을 적용한 모델로 기밀성보다는 정보의 불법 변경을 방지하기 위한 금융권 등에서 사용되는 모델</li> <li>무결성 모델, 군사적 모델</li> </ul>
클락-윌슨 (Clark-Wilson) 모델	<ul style="list-style-type: none"> <li>Biba 모델과 같이 정보의 무결성을 강조한 모델로서, Biba 모델보다 더 진화한 형태</li> <li>금융이나 회계 분야에서 기밀성보다 무결성이 중요함을 고려하여 설계</li> <li>무결성 모델, 상업적 모델</li> </ul>
만리장성 (Chines Wall, Brewer Nash) 모델	<ul style="list-style-type: none"> <li>사용자의 이해 충돌을 피하기 위한 모델</li> <li>어떤 회사의 특정 분야에서 근무했던 사람이 다른 회사의 같은 영역의 자료에 접근을 금지하는 모델</li> <li>직무분리를 접근통제에 반영한 모델</li> </ul>

## 12 키 분배 프로토콜의 종류 및 특징

KDC 기반 키 분배	<ul style="list-style-type: none"> <li>• KDC(Key Distribution Center), TA(Trusted Authority) 역할을 포함하는 포괄적 개념</li> <li>• 사용자는 A와 B가 비밀통신을 원할 때 KDC에게 작업시간을 포함하는 세션키를 요구하게 되고, KDC는 키를 생성하여 A와 B가 복호화할 수 있도록 암호화된 상태로 키를 전달하는 방법</li> <li>• 모든 사용자로부터 신뢰받는 TA(Trusted Authority) 존재</li> </ul>
Diffie-Hellman 키 분배	<ul style="list-style-type: none"> <li>• 사용자 간 암호화되지 않은 통신망. 즉, 공개된 통신망을 통해 공통의 비밀키를 공유할 수 있도록 하는 공개키 암호방식</li> <li>• 사용자는 각자 생성한 공개키를 상호 교환하여 공통의 비밀키를 생성하는 키 합의 방식</li> <li>• 키 전달 문제를 해결하기 위한 방식으로, 안전한 비밀키 공유를 목적으로 함</li> </ul>

## 13 키 배송 문제 해결 방법

키 사전 공유	키 관리기관이 사전에 임의의 두 사용자에게 임의의 키를 선택하여 전달하는 방법
키 분배 센터	키 분배 센터(KDC)라는 신뢰받는 제3자로부터 사용자와 키 분배 센터 간 키를 분배하는 방법
Diffie-Hellman 키 교환	공개키 암호방식을 이용하여 두 사용자 간 암호키를 안전하게 공유하는 방법
공개키 암호	송신자는 공개키를 사용하여 메시지를 암호화하고, 수신자는 개인키를 사용하여 암호문을 복호화하는 방법

## 14 전자서명의 종류 및 특징

부인방지 서명	<ul style="list-style-type: none"> <li>• 자체 인증방식을 배제하여 서명을 검증할 때, 반드시 서명자의 도움이 있어야 검증이 가능한 전자서명 방식</li> <li>• 서명자의 도움 없이 서명 진위 확인이 불가능하므로, 서명의 진위 검증도 제한적으로 수행하고자 할 때 사용</li> </ul>
은닉서명	<ul style="list-style-type: none"> <li>• D.Chaum에 의하여 제안된 방식</li> <li>• 서명자가 서명문 확인이 불가능한 상태에서 서명하도록 하는 방식</li> <li>• 서명하고자 하는 메시지의 내용을 공개하지 않고, 메시지에 대한 서명을 받고자 할 때 사용</li> <li>• 서명을 받는 사람의 신원과 서명문을 연결시킬 수 없어 익명성을 유지할 수 있음</li> <li>• 서명자와 송신자의 익명성을 보장하여 기밀성을 유지하게 하는 특수한 서명 방식</li> </ul>
이중서명	고객의 결제정보가 판매자를 통하여 지불정보 중계기관(PG)으로 전송됨에 따라, 고객의 결제정보가 판매자에게 노출될 가능성과 판매자에 의한 결제정보의 위·변조의 가능성 제거
위임서명	<ul style="list-style-type: none"> <li>• 위임서명자를 지정하여 서명하는 방식으로 제3자는 위임서명 생성 불가능</li> <li>• 위임서명 검증자는 위임서명을 위임한 서명자의 동의를 확인할 수 있어야 함</li> </ul>
대리서명	<ul style="list-style-type: none"> <li>• 본인 부재 시 대리로 서명하게 하는 방식으로 본인을 대신하여 제3자가 서명</li> <li>• 검증자는 대리서명으로부터 서명자의 위임사실을 확인할 수 있어야 함</li> </ul>
다중서명	<ul style="list-style-type: none"> <li>• 여러 사람이 서명하는 방식</li> <li>• 동시에 서명이 이루어지는 동시 다중서명과 서명이 순차적으로 이루어지는 순차 다중서명이 있음</li> </ul>

## 15 전자서명의 조건

위조불가	<ul style="list-style-type: none"> <li>합법적인 서명자만이 전자문서에 전자서명 생성 가능</li> <li>서명자 이외의 타인이 서명을 위조하기 어려워야 함</li> </ul>
서명자 인증	<ul style="list-style-type: none"> <li>전자서명의 서명자를 누구든지 검증 가능</li> <li>누구의 서명인지 확인 가능해야 함</li> </ul>
부인불가	<ul style="list-style-type: none"> <li>서명자는 서명 후 자신의 서명 사실을 부인 불가능</li> <li>서명자는 서명 사실을 부인할 수 없음</li> </ul>
변경불가	<ul style="list-style-type: none"> <li>서명한 문서의 내용 변경 불가능</li> <li>한 번 서명한 문서는 내용 변경 불가능</li> </ul>
재사용 불가	<ul style="list-style-type: none"> <li>전자문서의 서명은 다른 전자문서의 서명으로 재사용 불가능</li> <li>다른 문서의 서명을 위조하거나 기존 서명을 재사용할 수 없음</li> </ul>

## 16 공동 인증서의 구조

버전(Version)	인증서 형식의 연속된 버전 구분
일련번호(Serial Number)	발행 CA 내부 유일한 정수값
알고리즘 식별자 (Algorithm Identifier)	인증서를 생성하는 데 이용되는 서명 알고리즘을 확인하기 위한 OID 값
발행자(Issuer)	인증서를 발행하고 표시하는 CA
유효기간(Period of Validity)	인증서가 유효한 첫 번째와 마지막 날짜 2개로 구성
주체(Subject)	인증서가 가르키는 사람
공개키 정보 (Public-Key Information)	주체의 공개키와 이 키가 사용될 알고리즘 식별자
서명(Signature)	CA의 개인 서명키로 서명한 서명문
용도	E-Mail, SSL, 전자 지불, 소프트웨어 코드 서명, IPSec 등

## 17 인증서의 보관 및 폐기

유효기간이 지난 경우	인증기관은 해당 인증서를 디렉터리에서 제거하고, 추후 부인방지 서비스를 위하여 일정 기간 보관
인증서를 폐기하는 경우	<ul style="list-style-type: none"> <li>개인키가 유출되었다고 판단되는 경우</li> <li>사용자가 조직을 변경한 경우</li> <li>사용자가 CA에 인증서를 해지 신청한 경우</li> </ul>

## 18 공개키 기반 구조(PKI)의 구성요소

인증기관 (Certification Authority)	<ul style="list-style-type: none"> <li>PKI 구조에 가장 기반이 되는 요소</li> <li>인증서와 인증서 관리를 위한 모든 작업 담당</li> <li>인증서 발급, 인증서 상태관리, 인증서 문제 시 해당 인증서 철회를 위한 CRL 발급, 유효한 인증서와 CRL 목록 발행, 지금까지 발행한 인증서와 CRL의 모든 리스트 저장</li> <li>인증기관 : 금융결제원, 한국정보인증, 한국증권전산, 한국인터넷진흥원, 한국전자인증, 한국무역정보통신</li> </ul>
저장소, 디렉터리 (Repository)	<ul style="list-style-type: none"> <li>인증서와 CRL을 사용자에게 분배하는 역할</li> <li>사용자의 요청이 있을 경우 해당 인증서와 CRL을 사용자에게 전송</li> </ul>
사용자 (User)	<ul style="list-style-type: none"> <li>인증서를 사용하는 사람</li> <li>인증서를 CA로부터 발급받고, 다른 사람의 인증서를 사용하여 인증할 수도 있음</li> </ul>
인증서 정책 (Certificate Policy)	<ul style="list-style-type: none"> <li>인증서에 관련된 정책</li> <li>인증서를 어떻게 사용할 것인가에 따른 항목을 포함</li> </ul>
인증 요청서 (Certification Request)	<ul style="list-style-type: none"> <li>인증서 발급을 위한 요청서</li> <li>CA에게 인증서를 발급받기 위한 요청서</li> <li>사용자, 주체의 정보, 주체의 공개키 등의 정보 포함</li> </ul>
인증서 취소 목록 (Certificate Revocation List)	<ul style="list-style-type: none"> <li>CA에서 발급되는 인증서 중 취소된 인증서의 목록</li> <li>취소해야 하는 인증서 목록, 일련번호를 통하여 구분</li> <li>CA 정보, CRL 유효기간, CA가 CRL의 내용에 서명한 서명값 포함</li> </ul>

## 19 인증서 취소 목록(CRL)의 구조(기본영역)

서명 알고리즘	CRL에서 서명한 알고리즘 ID 및 관련 데이터
발급자	발급자 CA의 X.509 이름
최근 수정 일자	최근 수정 일자(UTC Time)
차후 수정 일자	다음 수정 일자(UTC Time)
취소 인증 목록	취소한 인증서 목록
CRL 확장자	CRL 확장자 유무 및 내용
발급자 서명문	발급자의 서명문

## 20 OCSP와 CRL의 비교

OCSP	CRL
<ul style="list-style-type: none"> <li>RFC 2560을 따름</li> <li>실시간 인증서 유효성 검증 프로토콜</li> <li>특정 CA와 사용 계약을 해야 하고, 사용량에 따라 추가 비용 지불</li> <li>CA와 계약 후 서버 인증서와 개인키가 발급되고, CA의 OCSP 서버에 인증서 유효성 요청 시 서버용 인증서 사용</li> <li>서버용 인증서는 1년마다 교체</li> </ul>	<ul style="list-style-type: none"> <li>RFC 3280을 따름</li> <li>CA가 인증서 폐기 시마다 인증서 취소 목록(CRL) 생성 하는 것이 아닌 일정 주기마다 인증서 취소 목록 생성 (6~24시간)</li> <li>CRL이 갱신되어야만 폐기로 판단</li> <li>비용 지불 없이 사용 가능</li> </ul>



## 21 암호 공격의 종류 및 특징

암호문 단독 공격 (Ciphertext Only Attack)	<ul style="list-style-type: none"> <li>암호문만을 이용하여 평문이나 키를 찾아내는 방법 예) 고전 암호</li> <li>평문의 통계적 성질과 문자의 특성 등을 추정하여 해독하는 방법</li> </ul>
기지 평문 공격 (Known Plaintext Attack)	<ul style="list-style-type: none"> <li>약간의 평문에 대응하는 암호문을 알고 있는 상태에서 사용하는 방법 예) 편지의 시작이나 끝말 등을 추측할 수 있음</li> <li>스니핑한 암호문에 대하여 암호화 방식 추론</li> <li>암호문과 평문의 관계로부터 키나 평문 추정</li> <li>주로 대칭키를 통하여 이루어진 통신 채널을 공격하기 위하여 사용</li> </ul>
선택 평문 공격 (Chosen Plaintext Attack)	<ul style="list-style-type: none"> <li>암호 해독자가 암호기에 접근할 수 있는 경우에 사용 가능하며, 평문을 추측하여 암호화한 후 비교하여 평문 추정</li> <li>평문을 선택하고, 그 평문에 해당하는 암호문을 얻어 키나 평문을 추정하여 암호를 해독하는 방법</li> <li>암호문에 사용된 알고리즘을 알고 있다면 메시지에서 특정한 문자열을 집중적으로 공격함으로써 키에 대한 정보를 알아낼 수 있음</li> <li>주로 공개키 시스템을 공격할 때 많이 사용</li> </ul>
선택 암호문 공격 (Chosen Ciphertext Attack)	<ul style="list-style-type: none"> <li>암호 복호기에 접근할 수 있는 경우에 사용 가능</li> <li>복호화 방식을 알고 있을 때 키값을 추정하여 복호화하는 공격방법으로, 일부 암호문에 대한 평문을 얻어 암호 해독</li> <li>주로 공개키 시스템을 공격할 때 많이 사용하며, 선택 평문 공격과는 반대</li> </ul>

## 22 암호 분석방법

전수 조사 (Exhaustive Attack)	평문을 암호화할 수 있는 모든 경우에 대하여 조사하는 방법으로, 가장 확실한 방법이지만 거의 실현 불가능
통계적 분석 (Statistical Analysis)	각 나라 언어와 마찬가지로 영어문장에 사용하는 알파벳은 고유한 출현 빈도를 가지고 있는데, 이러한 통계적인 자료를 이용하여 암호문을 분석하는 방법
수학적 분석 (Mathematical Analysis)	수학적인 이론을 이용하여 암호문을 분석하는 방법

## 23 혼돈(Confusion), 확산(Diffusion), 쇄도 효과(Avalanche Effect)

혼돈 (Confusion)	암호문의 통계적 성질과 평문의 통계적 성질 관계를 난해하게 만드는 성질
확산 (Diffusion)	각각의 평문 비트와 키 비트가 암호문의 모든 비트에 영향을 주는 성질
쇄도 효과 (Avalanche Effect)	평문 또는 키값을 조금만 변경시켜도 암호문에는 큰 변화가 생기는 효과

## 24 Feistel 구조와 SPN 구조를 사용하는 알고리즘

Feistel 구조를 사용하는 알고리즘	SPN 구조를 사용하는 알고리즘
DES	AES(Rijndael)
SEED(변형된 Feistel)	ARIA(Involucional SPN)
LOKI	IDEA
CAST	SHARK
Blowfish	Square
MISTY	CRYPTON
RC5, RC6	SAFER
CAST256	SAFER +
E2	Serpent
Twofish	
Mars	

## 25 주요 용어 정리(암호)

치환 암호	비트, 문자, 블록을 다른 비트, 문자, 블록으로 치환
전치 암호	비트, 문자, 블록을 재배열
블록 암호	블록(비트의 집합) 단위로 암호화
스트림 암호	한 번에 1비트 또는 1바이트의 단위로 암호화
링크 암호화	<ul style="list-style-type: none"> <li>• 네트워크 사업자가 암호화</li> <li>• 헤더를 포함한 데이터를 암호화하여 네트워크에 전송하면 라우터나 장비는 해당 패킷을 목적으로 전송하기 위해 복호화해야 함</li> </ul>
종단간 암호화	<ul style="list-style-type: none"> <li>• 사용자가 암호화</li> <li>• 헤더를 제외한 데이터를 암호화하여 네트워크에 전송하므로, 라우터나 장비는 해당 패킷을 목적으로 전송하기 위해 복호화할 필요가 없음</li> </ul>

## 26 스트림 암호화와 블록 암호화의 비교

	스트림 암호화	블록 암호화
암호화 과정	평문의 각 문자를 순서대로 암호화 스트림으로 만들	평문 자체를 블록 단위로 배열하고, 순차적으로 암호화
장점	<ul style="list-style-type: none"> <li>• 암호화 속도가 상대적으로 빠름</li> <li>• 오류 전파가 제한적</li> </ul>	<ul style="list-style-type: none"> <li>• 평문에 혼돈성을 주어 해독을 어렵게 함</li> <li>• 완성된 암호문에 내용 추가 및 변경 어려움</li> </ul>
단점	<ul style="list-style-type: none"> <li>• 평문의 특성이 암호문에도 그대로 반영</li> <li>• 약의적 공격자에 의하여 쉽게 내용첨가 및 변경 가능</li> </ul>	<ul style="list-style-type: none"> <li>• 암호화 속도가 상대적으로 느림</li> <li>• 암호화 시 오류의 파급효과가 큼</li> </ul>
알고리즘	RC4, A5/1, SEAL	DES, AES, IDEA, SEED, ARIA, RC5
용도	음성, 오디오/비디오 스트리밍	일반 데이터
예	<ul style="list-style-type: none"> <li>• 단순 알파벳 암호 알고리즘</li> <li>• 복합 알파벳 암호 알고리즘</li> </ul>	<ul style="list-style-type: none"> <li>• 세로 방향으로 옮긴 알고리즘</li> <li>• 모리스 부호 응용 알고리즘</li> </ul>

## 27 대칭키와 공개키 암호 알고리즘의 비교

구분	대칭키 암호 방식	공개키 암호 방식
암호키 관계	암호화 키와 복호화 키가 서로 같음	암호화 키와 복호화 키가 서로 다름
암호화 키	비밀	공개
복호화 키	비밀	비밀
암호 알고리즘	비밀로 하거나 공개하기도 함	공개
비밀키 수	$n(n-1)/2$ ( $n=100$ 이면 4,950개의 키가 생성)	$2n$ ( $n=100$ 이면 200개의 키가 생성)
안전한 인증	안전한 인증 곤란	안전한 인증 용이
암호화 속도	고속	저속
경제성	높음	낮음
전자서명	복잡	간단
예	DES, AES, IDEA, SEED, ARIA	RSA, Rabin, DSA, ECC, Elgamal

## 28 ARIA와 SEED 알고리즘의 비교

	ARIA	SEED
표준화	기술 표준원에서 개발	<ul style="list-style-type: none"> <li>정보통신단체 표준(TTA) 제정</li> <li>IETF 표준으로 제정</li> <li>ISO/IEC 국제표준 블록 암호 알고리즘으로 제정</li> </ul>
키 길이	128비트(고정)	128, 192, 256비트(가변)
알고리즘 구조	Involution SPN	변형된 Feistel
성능	암호화 시간 비율(ARIA : SEED)=2 : 1(ARIA가 우수)	

## 29 대칭키 암호 알고리즘의 비교

알고리즘	키 크기 (비트)	블록 크기 (비트)	라운드 수	응용
DES	56	64	16	SET, Keberos
3DES	112, 168	64	48	금융 키 관리, PGP, S/MIME
AES	128, 192, 256	128	10, 12, 14	DES와 3DES를 대체
IDEA	128	64	8	PGP
Blowfish	448까지 다양	64	16	다양한 소프트웨어 패키지
RC5	2048까지 다양	64	256까지 다양	다양한 소프트웨어 패키지
SEED	128, 192, 256	128	16	다양한 소프트웨어 패키지
ARIA	128	128	12, 14, 16	경량 환경 및 하드웨어 구현
CRYPTON	0~256	128	12	
RC5	0~256	64	16	알고리즘 간단, 속도 빠름
FEAL	64	64	4	소프트웨어 구현에 적합
MISTY	128	64	8	차분, 선형 공격에 안전
SKIPJACK	80	64	32	Fortezza 카드에 칩 형태로 사용

### 30 대칭키 암호 공격유형

차분 공격 (Differential Cryptanalysis)	<ul style="list-style-type: none"> <li>• 1980년대 후반 Eli Biham과 Shamir가 발견하면서 처음 알려진 공격으로, 여러 블록 암호와 암호학적 해시함수의 취약성을 발표하였으며, DES의 취약성을 연구하면서 DES가 차분 공격에 강하도록 설계되었다는 것을 발견</li> <li>• 암호 해독(Cryptanalysis)의 한 방법으로, 입력값의 변화에 따른 출력값의 변화를 이용하는 방법</li> <li>• 두 개의 평문 블록의 비트의 차이에 대하여 대응되는 암호문 블록의 비트 차이를 이용함으로써 암호키를 찾아내는 방법</li> <li>• 일반적으로 선택 평문 공격(Chosen Plaintext Attack)을 가정</li> <li>• 차분 쌍을 이용하여 블록 암호의 암호키를 찾는 공격이 가능할 수도 있음 : 블록 암호가 SPN(Substitution Permutation Network) 구조를 가질 때 마지막 라운드를 제외한 나머지 과정에서 차분 현상이 일어났다면 마지막 라운드에 입력되는 값의 차이를 일정 확률로 알 수 있고, 출력값의 차이를 예측할 수 있으므로 마지막 라운드에 더해지는 키의 일부도 알 수 있기 때문</li> </ul>
선형 공격 (Linear Cryptanalysis)	<ul style="list-style-type: none"> <li>• 마쓰이 미쓰루의 논문에서 처음 공개되었으며, 해당 논문에서는 선형 공격으로 FEAL 암호공격 방법 제시</li> <li>• 1993년 Matsui에 의하여 개발되어 기지 평문 공격법으로 알고리즘 내부의 비선형 구조를 적당히 선형화시켜 키를 찾는 방법</li> <li>• 암호공격의 한 방법으로, 암호화 과정에서 근사적 선형 관계식을 찾는 것을 목적으로 함</li> <li>• 차분 공격과 함께 블록 암호공격 방법으로 널리 이용</li> <li>• 먼저 암호화 과정에서 근사적인 선형 관계성을 찾음</li> <li>• 선형 관계식을 찾는 방법은 암호화 과정에 따라 다르며, SPN의 경우 치환과정은 선형적이며, S-Box에서의 대치과정에서만 비선형적인 변환이 일어남</li> <li>• 따라서 S-Box의 연산을 선형과정으로 근사화할 방법을 찾는 것이 목표이고, 근사 선형 관계성을 찾았다면 이를 통하여 선택 평문 공격을 수행할 수 있음</li> <li>• 암호화 모듈에 임의의 입력값을 입력할 수 있을 경우, 관계식에서 입력값 부분과 출력값 부분이 얼마나 일치하는지 횟수를 측정하여 이 횟수를 기반으로 암호화 키 비트에 대한 관계식을 확률적으로 추정할 수 있음</li> </ul>

### 31 스트림 암호화와 블록 암호화의 비교

	스트림 암호화	블록 암호화
암호화 과정	평문의 각 문자를 순서대로 암호화 스트림으로 만들	평문 자체를 블록 단위로 배열하고, 순차적으로 암호화
장점	<ul style="list-style-type: none"> <li>• 암호화 속도가 상대적으로 빠름</li> <li>• 오류 전파가 제한적</li> </ul>	<ul style="list-style-type: none"> <li>• 평문에 혼돈성을 주어 해독을 어렵게 함</li> <li>• 완성된 암호문에 내용 추가 및 변경 어려움</li> </ul>
단점	<ul style="list-style-type: none"> <li>• 평문의 특성이 암호문에도 그대로 반영</li> <li>• 악의적 공격자에 의하여 쉽게 내용추가 및 변경 가능</li> </ul>	<ul style="list-style-type: none"> <li>• 암호화 속도가 상대적으로 느림</li> <li>• 암호화 시 오류의 파급효과가 큼</li> </ul>
예	<ul style="list-style-type: none"> <li>• 단순 알파벳 암호 알고리즘</li> <li>• 복합 알파벳 암호 알고리즘</li> </ul>	<ul style="list-style-type: none"> <li>• 세로 방향으로 옮긴 알고리즘</li> <li>• 모리스 부호 응용 알고리즘</li> </ul>

### 32 공개키 암호 알고리즘의 비교

소인수분해	기본 원리	<ul style="list-style-type: none"> <li>백자리 크기 이상의 두 소수 <math>p, q</math>가 존재하고 <math>p, q</math>의 곱 <math>n</math>을 계산할 경우 <math>p</math>와 <math>q</math>를 알고 있는 사람은 <math>n</math>을 계산하기 쉬움</li> <li><math>n</math>만 알고 있는 사람은 <math>n</math>으로부터 <math>p</math>와 <math>q</math>를 인수분해하여 찾아내기 매우 어려움</li> </ul>
	대표 알고리즘	RSA, Rabin 등
이산대수	기본 원리	<ul style="list-style-type: none"> <li>사용자는 큰 소수 <math>p</math>를 선정하여 <math>Z_p</math> 상의 원시원소 <math>g</math>와 함께 <math>p</math>를 공개</li> <li>송신자 A : <math>Z_p</math> 상의 임의의 원소 <math>x_A</math>를 비밀정보로 선택하여, <math>y_A \equiv g^{x_A} \pmod{p}</math>의 공개 정보 <math>y_A</math>를 계산</li> <li>수신자 B : <math>Z_p</math> 상의 임의의 원소 <math>x_B</math>를 비밀정보로 선택하여, <math>y_B \equiv g^{x_B} \pmod{p}</math>의 공개 정보 <math>y_B</math>를 계산</li> <li>송신자 A와 수신자 B의 <math>y_A, y_B, p, q</math>를 공개 목록에 등록 : <math>y_A</math>와 <math>y_B</math>가 송신자 A와 수신자 B의 공개키 <math>K_e</math>이고, <math>x_A</math>와 <math>x_B</math>가 송신자 A와 수신자 B의 비밀키 <math>K_d</math>가 됨</li> </ul>
	대표 알고리즘	ElGamal, 타원 곡선(ECC), KCDSA 등

### 33 해시함수의 안정성

일방향성 (One-Wayness) (역상저항성)	<ul style="list-style-type: none"> <li>주어진 임의의 출력값에 대응하는 입력 메시지를 찾는 것이 계산적으로 불가능</li> <li>주어진 임의의 출력값 <math>y</math>에 대하여, <math>y=h(x)</math>를 만족하는 입력값 <math>x</math>를 찾는 것이 계산적으로 불가능</li> </ul>
충돌 저항성 (Collision Resistance)	<ul style="list-style-type: none"> <li>충돌을 발견하는 것이 어려운 성질</li> <li>충돌 내성을 가질 필요가 있음</li> </ul>
	<p>[1st 충돌 저항 : 두 번째 역상 저항성]</p> <ul style="list-style-type: none"> <li>주어진 입력값에 대응하는 출력값과 같은 출력값을 갖는 다른 입력값을 찾는 것이 계산적으로 불가능</li> <li>주어진 입력값 <math>x</math>에 대하여 <math>h(x)=h(x')</math>, <math>x \neq x'</math>을 만족하는 다른 입력값 <math>x'</math>을 찾는 것이 계산적으로 불가능</li> </ul>
	<p>[2nd 충돌 저항 : 충돌 저항성]</p> <ul style="list-style-type: none"> <li>임의의 두 입력 쌍에 대하여 같은 출력값을 갖는 서로 다른 입력값을 찾는 것이 계산적으로 불가능</li> <li><math>h(x)=h(x')</math>을 만족하는 임의의 두 입력값 <math>x, x'</math>을 찾는 것이 계산적으로 불가능</li> </ul>

### 34 전자서명에 사용되는 해시함수의 성질

약 일방향성 (Weak Onewayness)	해시값 $H$ 로부터 $h(M)=H$ 가 되는 서명문 $M$ 을 찾는 것은 계산상 불가능(해시함수의 역함수 계산 방지)
강 일방향성 (Strong Onewayness)	어떤 서명문 $M$ 과 그의 해시값 $H=h(M)$ 가 주어졌을 때, $h(M')=H$ 되는 서명문 $M' \neq M$ 을 찾는 것은 계산상 불가능(해시함수의 역함수 계산 방지)
충돌 회피성 (Collision Freeness)	$h(M)=h(M')$ 되는 서명문 쌍 $(M, M')$ ( $M \neq M'$ )을 찾는 것은 계산상 불가능(부인방지)

### 35 해시함수의 안전성 정리

구분	동일 용어	전자서명	함수 공식	취약 공격
역상 저항성	프리이미지 저항성	약 일방향성	$y = h(x) = x?$	-
두 번째 역상 저항성	제2프리이미지 저항성, 약한 충돌 내성	강 일방향성	$h(x) = h(x') = x?$	무차별 대입 공격
충돌 저항성	강한 충돌성	충돌 회피성	$h(x) = h(x') \Rightarrow x, x'?$	생일공격

### 36 해시함수의 종류별 비교

알고리즘	출력 해시값 크기	블록 크기	충돌성
HAVAL	256/224/192/160/128	1,024	있음
MD2	128	128	있음
MD4	128	512	있음
MD5	128	512	있음
PANAMA	256	256	있음
RIPEMD	128	512	있음
RIPEMD-128/256	128/256	512	없음
RIPEMD-160/320	160/320	512	없음
SHA-0	160	512	있음
SHA-1	160	512	있음
SHA-256/224	256/224	512	없음
SHA-512/384	512/384	1024	없음

### 37 해시함수 공격의 종류 및 특성

생일 공격 (Birthday Attack)	<ul style="list-style-type: none"> <li>• <math>n</math>비트 출력을 하는 해시함수는 생일의 역설에 의하여 <math>2n/2</math>개의 메시지만 있으면 <math>1/2</math>이상 충돌쌍이 발생하기 때문에 128비트, 180비트의 출력값을 갖는 해시함수의 안전도는 각각 64비트, 80비트의 안전도를 가짐</li> <li>• 해시함수뿐만 아니라 메시지 인증 코드(MAC)의 안전성 분석에서도 중요한 역할</li> </ul>
일치 블록 연쇄 공격	새로운 메시지 $M'$ 를 사전에 다양하게 만들어 놓은 상태에서 공격하고자 하는 메시지 $M$ 의 해시함수 값 $h(M)$ 와 같은 해시함수 값을 갖는 것을 선택하여 사용하는 공격
중간자 연쇄 공격	전체 해시값이 아니라 해시 중간의 결과에 대한 충돌쌍을 찾아서 특정 포인트를 공격
고정점 연쇄 공격	<ul style="list-style-type: none"> <li>• 압축함수에서 고정점이란 <math>f(H_{i-1}, x_i) = H_{i-1}</math>을 만족하는 쌍 <math>(H_{i-1}, x_i)</math>을 말함</li> <li>• 메시지 블록과 연쇄 변수쌍을 얻게 되면 연쇄변수가 발생하는 특정한 점에서 임의의 수의 동등한 블록들 <math>x_i</math>를 메시지의 중간에 삽입해도 전체 해시값이 변하지 않음</li> <li>• 이러한 공격은 고정점으로 밝혀진 특정한 점에서 연쇄변수의 값을 다양하게 조절할 수 있다는 전제가 있어야 효율적이며, 따라서 연쇄변수가 알려진 고정점에 대하여 어떤 값을 갖도록 재배열될 수 있는지에 따라 공격 가능 여부 결정</li> </ul>
차분 연쇄 공격	<p>다중 라운드 블록 암호의 공격</p> <p>- 다중라운드 블록 암호를 사용하는 해시함수에서 입력값과 그에 대응하는 출력값 차이의 통계적 특성을 조사하는 기법</p>

### 38 MDC와 MAC의 비교

구분	특징	종류	무결성	인증	부인방지	키 유무
MDC	Hash	MD5, SHA-1	○	×	×	없음
MAC	Hash+대칭키	HMAC, CBC-MAC	○	○	×	있음



## 1 정보보호의 특성

기밀성 (Confidentiality)	개념	<ul style="list-style-type: none"> <li>내부 정보 비인가된 개인, 단체, 프로세스 등으로부터 중요한 정보를 보호하는 것</li> <li>정보 소유자의 인가를 받은 사용자만이 정보 접근이 가능하도록 하는 것</li> <li>접근통제의 모든 행위는 근본적으로 기밀성 보호를 위한 것</li> </ul>
	침해유형	공개(Display), 노출(Exposure)
	접근통제	물리적 수준, 운영체제 수준, 네트워크 수준 접근통제
무결성 (Integrity)	개념	내부에 있는 정보의 저장과 전달 시 비인가된 방식으로 정보와 소프트웨어가 변경되지 않도록 정확성과 안정성을 보호하는 것
	침해유형	변조(Alteration), 파괴(Destruction)
	무결성 통제	물리적 통제와 해시함수 사용
가용성 (Availability)	개념	내부의 정보자원에 대하여 인가된 사용자가 정보나 서비스를 요구할 때 언제든지 사용 가능하도록 하는 것
	침해유형	지체(Delay), 재난(Disaster)
	가용성 통제	데이터 백업, 중복성 유지, 물리적 위협으로부터 보호 기술 사용

## 2 정보보호 기술의 분류

관리적 보안	<ul style="list-style-type: none"> <li>기업의 민감한 정보를 보호하기 위한 인적·행정적 정보보호 방안</li> <li>정보보호 관리체계(ISMS), 정보보안 지침과 절차, 비상대책 수립과 보안사고 대응 능력, 자산의 보안 등급 분류 및 가치평가, 보안 관리 및 보안교육 시행 등</li> </ul>
기술적 보안	<ul style="list-style-type: none"> <li>정보보호를 위한 기술적 보안시스템</li> <li>보안 솔루션, 보안 모니터링 및 감사, DRM, 인증, 데이터 암호화, DB 보안, ESM 등</li> </ul>
물리적 보안	<ul style="list-style-type: none"> <li>중요한 자원 및 민감한 정보를 보호하기 위한 물리적인 시설 및 수단</li> <li>입·출입 통제시스템, 자연재해 통제, 데이터 백업 및 저장매체 반입·반출 통제시스템, 재해복구시스템, 전원 및 케이블 보호, 제한구역 설정, 항온항습 장치 등</li> </ul>

### 3 정책, 표준, 지침, 절차의 정의 및 특성

구분	정의 및 특성
정책 (Policy)	<ul style="list-style-type: none"> <li>정보보호에 대한 상위 수준의 목표 및 방향 제시</li> <li>조직의 경영목표를 반영하고, 정보보호 관련 상위정책과 일관성 유지</li> <li>정보보호를 위하여 관련된 모든 사람이 반드시 지켜야 할 요구사항을 전반적이며 개략적으로 규정</li> </ul>
표준 (Standard)	<ul style="list-style-type: none"> <li>정보보호 정책과 마찬가지로 반드시 지켜야 하는 요구사항에 대한 규정이지만, 정책의 만족을 위하여 반드시 준수해야 할 구체적인 사항이나 양식 규정</li> <li>조직의 환경 또는 요구사항에 따라 관련된 모든 사용자가 준수하도록 요구되는 규정</li> </ul>
지침 (Guideline)	<ul style="list-style-type: none"> <li>반드시 지켜야 하는 것이 아니라 선택적이거나, 권고적인 내용이며, 융통성 있게 적용할 수 있는 사항 설명</li> <li>정보보호 정책에 따라 특정 시스템 또는 특정 분야별로 정보보호 활동에 필요한 세부 정보 설명</li> </ul>
절차 (Procedure)	<ul style="list-style-type: none"> <li>정책을 만족하기 위하여 수행해야 하는 사항을 순서에 따라 단계적으로 설명</li> <li>정보보호 활동의 구체적인 적용을 위하여 필요한 적용 절차 등의 구체적이고 세부적인 방법 기술</li> </ul>

### 4 위협의 종류

가로채기 (Interception)	<ul style="list-style-type: none"> <li>비인가된 사용자, 공격자가 전송 중인 정보를 열람하거나 도청하는 행위</li> <li>정보의 기밀성 보장 위협</li> </ul>
위조 (Fabrication)	<ul style="list-style-type: none"> <li>마치 다른 송신자로부터 정보가 수신된 것처럼 속이는 것. 즉, 시스템에 불법접근 후 저장정보를 변경하여 정보를 속이는 행위</li> <li>정보의 무결성 보장 위협</li> </ul>
변조 (Modification)	<ul style="list-style-type: none"> <li>시스템에 불법적으로 접근하여 데이터를 다른 내용으로 바꾸는 행위</li> <li>정보의 무결성 보장 위협</li> </ul>
차단 (Interruption)	<ul style="list-style-type: none"> <li>정보의 송수신을 원활하게 하지 못하도록 방해하는 행위</li> <li>정보의 흐름을 차단하고, 정보의 가용성 보장 위협</li> </ul>

### 5 위협의 구성요소

자산 (Asset)	조직이 보호해야 할 대상으로, 정보, 하드웨어, 소프트웨어, 시설 등을 말하며, 관련 인력, 기업 이미지 등의 무형자산을 포함하기도 함
위협 (Threat)	자산에 손실을 초래할 수 있는 원치 않는 사건의 잠재적 원인(Source)이나 행위자(Agent)
취약성 (Vulnerability)	자산의 잠재적 속성이며, 위협의 이용 대상으로 정의하나, 때로는 정보보호 대책의 미비로 정의되기도 함

## 6 위험분석 방법론

베이스라인 접근법 (Baseline Approach)	<ul style="list-style-type: none"> <li>모든 시스템에 대하여 표준화된 보호 대책의 항목들을 체크리스트 형태로 제공</li> <li>체크리스트에 있는 보호 대책이 현재 구현되어 있는지를 조사하여 구현되지 않은 보호 대책 식별</li> </ul>
비정형 접근법 (Informal Approach)	<ul style="list-style-type: none"> <li>구조적인 방법론에 의하지 않고, 경험자의 지식을 사용하여 위험분석 수행</li> <li>특정 위험분석 방법론과 기법을 선정하여 수행하지 않고, 수행자의 경험에 따라 중요 위험 중심으로 분석</li> </ul>
상세위험분석 (Detailed Risk Analysis)	<ul style="list-style-type: none"> <li>자산분석, 위협분석, 취약성 분석의 각 단계를 수행하여 위험평가</li> <li>방법론에 따라 취약성 분석과 별도로 설치된 정보보호 대책에 대한 분석을 수행하기도 함</li> </ul>
복합 접근법 (Combined Approach)	고위험(High Risk) 영역을 식별하여 상세위험 분석을 수행하고, 그 외의 다른 영역은 베이스라인 접근법을 사용하는 방식

## 7 정량적 · 정성적 분석의 비교

구분	정량적 분석	정성적 분석
개념	위험 발생 확률×손실 크기=기대 위험 가치분석	<ul style="list-style-type: none"> <li>손실 크기를 화폐가치로 표현하기 어려움</li> <li>위험 크기는 기술변수로 표현</li> </ul>
유형	<ul style="list-style-type: none"> <li>수학 공식 접근법</li> <li>과거 자료 분석법</li> <li>확률 분포</li> <li>확률지배</li> <li>몬테카를로 시뮬레이션</li> </ul>	<ul style="list-style-type: none"> <li>델파이법</li> <li>시나리오법</li> <li>순위 결정법</li> <li>퍼지 행렬법</li> <li>질문서법</li> </ul>
척도	연간기대손실(ALE)	점수(5점 척도, 10점 척도)
장점	<ul style="list-style-type: none"> <li>객관적인 평가 기준 적용</li> <li>정보의 가치가 논리적으로 평가되고, 화폐로 표현되기 때문에 이해하기 쉬움</li> <li>위험관리, 성능평가 용이</li> <li>위험평가 결과가 금전적 가치, 백분율, 확률 등으로 표현되기 때문에 이해하기 쉬움</li> </ul>	<ul style="list-style-type: none"> <li>계산에 대한 노력이 적게 들</li> <li>정보 자산에 대한 가치평가 불필요</li> <li>비용 · 이익평가 불필요</li> </ul>
단점	<ul style="list-style-type: none"> <li>계산이 복잡하여 분석하는 데 시간 · 노력 · 비용이 많이 들</li> <li>수작업의 어려움으로 자동화 도구를 사용 시 신뢰도가 벤더에 의존</li> </ul>	<ul style="list-style-type: none"> <li>위험평가과정과 측정기준이 주관적이기 때문에 사람에 따라 달라질 수 있음</li> <li>측정결과를 화폐가치로 표현하기 어려움</li> <li>위험 완화 대책의 비용 · 이익분석에 대한 근거가 제공되지 않고, 문제에 대한 주관적인 지적만 있음</li> <li>위험관리 성능을 추적할 수 없음</li> </ul>
사용지역	미국	유럽

## 8 위험처리 전략

위험수용 (Risk Acceptance)	<ul style="list-style-type: none"> <li>• 현재의 위험을 받아들이고, 잠재적 손실 비용을 감수하는 것</li> <li>• 어떠한 대책을 도입하더라도 위험을 완전히 제거할 수는 없으므로, 일정 수준 이하의 위험은 감수하고 사업을 진행하는 것</li> </ul>
위험감소 (Risk Reduction)	<ul style="list-style-type: none"> <li>• 위험을 감소시킬 수 있는 대책을 채택하여 구현하는 것으로, 대책의 채택 시에는 이에 따른 비용이 소요되기 때문에 비용과 실제 감소하는 위험의 크기를 비교하는 비용효과 분석이 필요</li> <li>• 정보보호 대책의 효과=기존 ALE-대책구현 후 ALE-연간 대책 비용</li> </ul>
위험회피 (Risk Avoidance)	<ul style="list-style-type: none"> <li>• 위험이 존재하는 프로세스나 사업을 수행하지 않고 포기하는 것</li> <li>• 위험 발생 원인을 제거하는 것, 즉 위험의 영향으로부터 프로젝트 목표를 보호하기 위하여 프로젝트 자체를 수정하는 것</li> </ul>
위험전가 (Risk Transfer)	<ul style="list-style-type: none"> <li>• 보험이나 외주 등으로 잠재적 비용을 제3자에게 이전하거나 할당하는 것</li> <li>• 위험의 발생결과 및 대응의 주체를 제3자에게 이동시키는 것</li> <li>• 유의할 점은 위험에 대한 관리 책임을 제3자에게 전가하는 것이 위험 자체를 제거하는 것이 아니라는 것</li> </ul>

## 9 재해 복구 시스템

미러 사이트 (Mirror Site)	<ul style="list-style-type: none"> <li>• 주 센터와 동일한 구성의 백업 센터를 구축하고, 주 센터와 백업 센터 간 실시간 데이터 동기화를 유지하여, 주 센터 재해 발생 즉시 백업 센터에서 업무 대행을 실시간으로 처리할 수 있음</li> <li>• 주 센터와 동일한 수준의 정보기술 자원을 원격지에 구축하고, 주 센터와 재해복구센터가 모두 운용 상태로 서비스하는 방식(Active/Active 방식)</li> <li>• RTO(복구 소요 시간)는 이론적으로 0</li> </ul>
핫 사이트 (Hot Site)	<ul style="list-style-type: none"> <li>• 재난 발생으로 영향을 받는 업무기능을 즉시 복구할 수 있도록, 주 센터와 동일한 모든 설비와 자원을 보유한 안전한 시설</li> <li>• 자원을 대기 상태로 사이트에 보유하면서 동기적 또는 비동기적 방식으로, 실시간 미러링을 통하여 데이터를 최신 상태로 유지(Active/Standby 방식)</li> <li>• 주 센터와 동일한 H/W, S/W, 부대설비를 준비하고, 실시간 DB Log 전송 및 DB 이미지 백업(Image Backup)을 준비하여, 주 센터 재해 발생 시 데이터 복구 작업을 실시</li> <li>• RTO(복구 소요 시간)는 수 시간 이내</li> </ul>
웜 사이트 (Warm Site)	<ul style="list-style-type: none"> <li>• 주 센터와 동일한 수준의 정보기술 자원을 보유하는 대신, 중요성이 높은 기술 자원만 부분적으로 보유하는 방식</li> <li>• 주 센터 장비 일부 및 데이터 백업만을 준비하여 재해 발생 시 주요 업무 데이터만 복구하는 시설</li> <li>• 실시간 미러링을 수행하지 않음</li> <li>• RTO(복구 소요 시간)는 데이터 백업 주기가 수 시간~1일 정도</li> </ul>
콜드 사이트 (Cold Site)	<ul style="list-style-type: none"> <li>• 재난 발생 시 새로운 정보시스템을 설치할 수 있는 전산실을 미리 준비하여 둔 것으로, 별다른 장비는 갖추고 있지 않음</li> <li>• 데이터만 원격지에 보관하고 서비스를 위한 정보자원은 확보하지 않거나, 최소한으로만 확보하는 유형</li> <li>• RTO(복구 소요 시간)는 주 센터의 데이터를 주기적으로 수 일~수 주로 원격지에 백업</li> </ul>
상호 백업 협정	서로 비슷한 시스템을 갖추고 있는 기업 간 재난 발생 시 상호 백업해주기로 협정

## 10 국내외 평가기준 등급체계 비교

미국				캐나다	유럽		국제		한국
TCSEC		FC		CTCP EC	ITSEC		Common Criteria		침입차단 시스템 평가기준
		PP	보증						
D	최소한의 보호	—	—	—	E0	부적절한 보증	EAL0	부적절한 보증	K0
		—	—	—			EAL1	기능시험	K1(E)
C1	임의적 보호	—	—	—	E1 F-C1	비정형적 기본설계	EAL2	구조시험	K2(E)
C2	통제된 접근보호	CS-1	T1	T1	E2 F-C2	비정형적 상세설계	EAL3	방법론적 시험과 점검	K3(E)
B1	레이블된 보호	LP-1 CS-2 CS-3	T2 T3 T4	T2 T3	E3 F-B1	소스코드와 하드웨어 도면 제공	EAL4	방법론적인 설계, 시험 및 검토	K4(E)
B2	구조적 보호	LP-2	T5	T4	E4 F-B2	준정형적 기능명세서, 기본설계, 상세설계	EAL5	준정형적 설계 및 시험	K5(E)
B3	보안영역	LP-3	T6	T5	E5 F-B3	보안요소 상호관계	EAL6	준정형적 검증된 설계 및 시험	K6(E)
A1	검증된 설계	LP-4	T7	T6T7	E6 F-B3	정형적 기능 명세서, 상세설계	EAL7	정형적 검증	K7(E)

## 11 정보보호 및 개인정보보호 관리체계 인증(ISMS-P)

영역	분야	적용 여부	
		ISMS	ISMS-P
관리체계 수리 및 운영 (16개)	관리체계 기반 마련	○	○
	위험관리	○	○
	관리체계 운영	○	○
	관리체계 점검 및 개선	○	○
보호대책 요구사항 (64개)	정책, 조직, 자산 관리	○	○
	인적 자원	○	○
	외부자 보안	○	○
	물리 보안	○	○
	인증 및 권한관리	○	○
	접근통제	○	○
	암호화 적용	○	○
	정보시스템 도입 및 개발 보안	○	○
	시스템 및 서비스 운영관리	○	○
	시스템 및 서비스 보안관리	○	○
	사고 예방 및 대응	○	○
	재해복구	○	○
개인정보 처리 단계별 요구사항 (22개)	개인정보 수집 시 보호조치	-	○
	개인정보 보유 및 이용 시 보호조치	-	○
	개인정보 제공 시 보호조치	-	○
	개인정보 파기 시 보호조치	-	○
	정보주체 권리보호	-	○

## 12 사이버 위기 경보 단계

심각	<ul style="list-style-type: none"> <li>국가 차원의 주요 정보통신망 및 정보시스템 장애 또는 마비</li> <li>침해사고가 전국적으로 발생했거나 피해 범위가 대규모인 사고 발생</li> </ul>
경계	<ul style="list-style-type: none"> <li>복수 정보통신서비스제공자(ISP)망·기간통신망에 장애 또는 마비</li> <li>침해사고가 다수기관에서 발생했거나 대규모 피해로 확대될 가능성 증가</li> </ul>
주의	<ul style="list-style-type: none"> <li>일부 정보통신망 및 정보시스템 장애</li> <li>침해사고가 다수기관으로 확산될 가능성 증가</li> <li>국내·외 정치·군사적 위기 발생 등 사이버 안보 위해 가능성 고조</li> </ul>
관심	<ul style="list-style-type: none"> <li>위험도가 높은 웜·바이러스, 취약점 및 해킹 기법 출현으로 인해 피해 발생 가능성 증가</li> <li>해외 사이버공격 피해가 확산되어 국내 유입 우려</li> <li>침해사고가 일부 기관에서 발생</li> <li>국내·외 정치·군사적 위기 상황 조성 등 사이버 안보 위해 가능성 증가</li> </ul>

## 13 개인정보보호법

목적	개인정보의 처리 및 보호에 관한 사항을 정함으로써 개인의 자유와 권리를 보호하고, 나아가 개인의 존엄과 가치를 구현함을 목적으로 함
개인정보보호 책임자의 업무	<ul style="list-style-type: none"> <li>개인정보 보호 계획의 수립 및 시행</li> <li>개인정보 처리 실태 및 관행의 정기적인 조사 및 개선</li> <li>개인정보 처리와 관련한 불만의 처리 및 피해 구제</li> <li>개인정보 유출 및 오용·남용 방지를 위한 내부통제시스템의 구축</li> <li>개인정보 보호 교육 계획의 수립 및 시행</li> <li>개인정보 파일의 보호 및 관리·감독</li> <li>그 밖에 개인정보의 적절한 처리를 위하여 대통령령으로 정한 업무</li> </ul>
개인정보 수집 시 동의 사항	<ul style="list-style-type: none"> <li>개인정보의 수집·이용 목적</li> <li>수집하려는 개인정보의 항목</li> <li>개인정보의 보유 및 이용 기간</li> <li>동의를 거부할 권리가 있다는 사실 및 동의 거부에 따른 불이익이 있는 경우에는 그 불이익의 내용</li> </ul>
개인정보 제공 시 동의 사항	<ul style="list-style-type: none"> <li>개인정보를 제공받는 자</li> <li>개인정보를 제공받는 자의 개인정보 이용 목적</li> <li>제공하는 개인정보의 항목</li> <li>개인정보를 제공받는 자의 개인정보 보유 및 이용 기간</li> <li>동의를 거부할 권리가 있다는 사실 및 동의 거부에 따른 불이익이 있는 경우에는 그 불이익의 내용</li> </ul>

## 14 정보통신망 이용촉진 및 정보보호 등에 관한 법률

목적	정보통신망의 이용을 촉진하고 정보통신서비스를 이용하는 자를 보호함과 아울러 정보통신망을 건전하고 안전하게 이용할 수 있는 환경을 조성하여 국민생활의 향상과 공공복리의 증진에 이바지함을 목적으로 함
시책	<ul style="list-style-type: none"> <li>• 정보통신망에 관련된 기술의 개발·보급</li> <li>• 정보통신망의 표준화</li> <li>• 정보내용물 및 제11조에 따른 정보통신망 응용서비스의 개발 등 정보통신망의 이용 활성화</li> <li>• 정보통신망을 이용한 정보의 공동활용 촉진</li> <li>• 인터넷 이용의 활성화</li> <li>• 정보통신망에서의 청소년 보호</li> <li>• 정보통신망을 통하여 유통되는 정보 중 인공지능 기술을 이용하여 만든 거짓의 음향·화상 또는 영상 등의 정보를 식별하는 기술의 개발·보급</li> <li>• 정보통신망의 안전성 및 신뢰성 제고</li> <li>• 그 밖에 정보통신망 이용 촉진 및 정보보호 등을 위하여 필요한 사항</li> </ul>