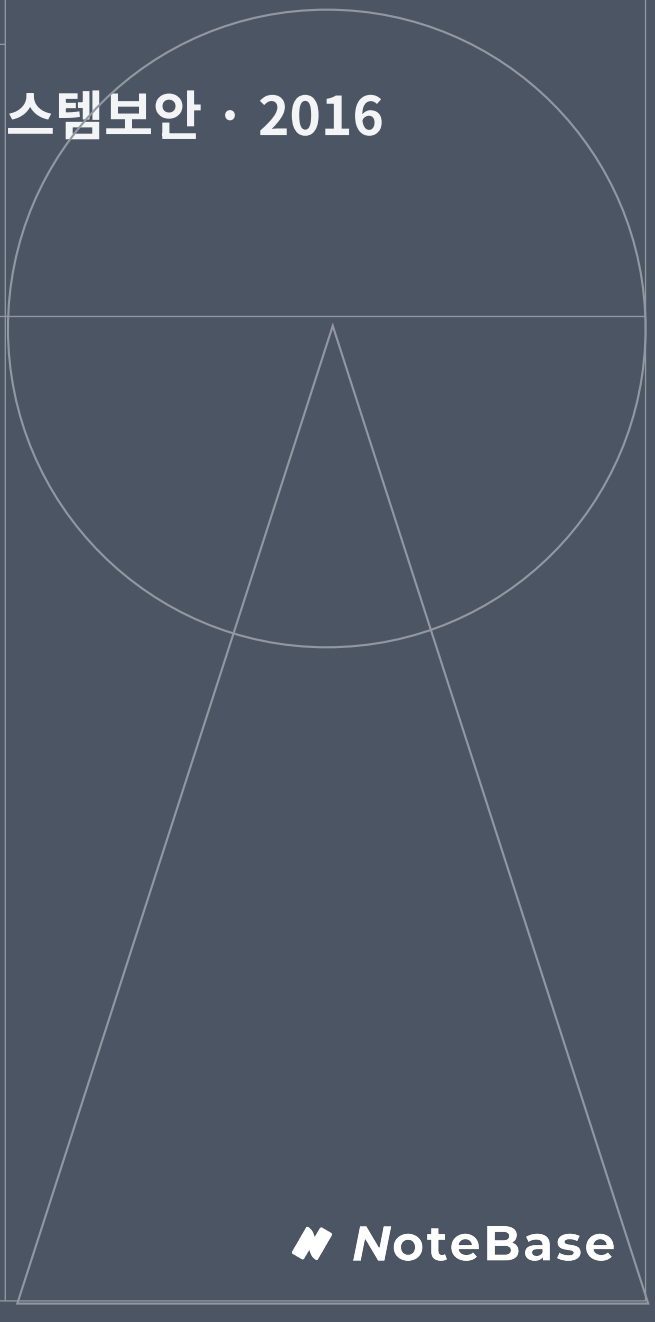
기출OX 해설집

공무원/국가직/9급 · 정보시스템보안 · 2016



NoteBase

기출OX 해설집

공무원/국가직/9급 · 정보시스템보안 · 2016

공무원/국가직/9급 · 정보시스템보안 · 2016

문 1. 모바일 운영체제인 iOS와 안드로이드의 보안 체계에 대한 설명으로 옳지 않은 것은?

① iOS는 모든 앱에 대한 코드 무결성 점검을 수행하여 설치를 제한한다.

② iOS와 안드로이드 모두 프로그램의 실행 권한이 일반 사용자에게 있다.

③ iOS는 애플의 CA를 통하여 앱을 서명 및 배포하고, 안드로이드는 개발자가 서명 및 배포한다.

④ 보안 통제권이 iOS는 애플에 있고, 안드로이드는 개발자나 사용자에게 있다.

정답 2번

간단한 해설

②의 '모두 일반 사용자에게 있다'는 설명은 실행 허용의 통제 주체를 같게 본 것이다. iOS는 **애플의 인증·서명 체계**로 실행 가능한 앱을 통제하고, 안드로이드는 개발자가 서명한 앱의 설치·실행을 **사용자가 선택**하는 방식이다.

NoteBase

선지별 해설

① O: iOS는 앱의 코드 서명을 검증하여 승인된 출처의 코드인지, 서명 이후 변조되지 않았는지 확인한다. 이러한 **코드 무결성 검증**을 통해 유효한 서명이 없는 앱의 설치·실행을 제한한다. 무결성 확인은 코드의 변조 여부를 확인하는 것이며, 앱에 취약점이 전혀 없음을 보장하는 것은 아니다.

② X: iOS에서는 애플이 발급한 인증서에 따른 코드 서명이 실행 허용의 전제가 되므로, 일반 사용자가 임의의 코드 실행을 허용할 수 없다. 안드로이드는 개발자가 서명한 앱을 사용자가 선택하여 설치·실행하는 구조이다. 따라서 이 문항에서 실행 허용의 통제 주체는 **iOS는 애플, 안드로이드는 일반 사용자**로 구분한다. 여기서 실행 권한은 이미 설치된 앱을 사용자가 눌러 실행한다는 의미가 아니라, 어떤 프로그램의 실행을 허용하는지를 뜻한다.

③ O: CA는 인증서를 발급하는 인증기관이다. iOS의 개발자는 **애플이 발급한 인증서**로 앱을 서명하고, App Store 등 애플의 배포 체계를 이용한다. 안드로이드는 개발자가 자체 생성한 인증서로 앱에 서명하여 배포할 수 있으며, 앱 서명에 중앙 인증기관의 인증서가 반드시 필요한 것은 아니다. 따라서 선지의 차이는 서명을 전혀 하지 않는지 여부가 아니라, 서명에 쓰이는 인증서의 신뢰를 누가 통제하는지에 있다.

④ O: 앱의 서명·배포를 기준으로 보면 iOS는 애플이 인증서와 실행 허용 조건을 통제하는 중앙집중적 구조이다. 안드로이드는 개발자가 앱을 서명·배포하고 사용자가 설치 여부나 앱에 허용할 권한을 결정하는 등 **개발자와 사용자의 선택 범위가 더 넓다**. 다만 이는 사용자가 운영체제의 모든 보안 제한을 해제할 수 있다는 뜻은 아니다. 안드로이드도 앱별 격리 공간인 샌드박스와 권한 검사로 앱의 동작 범위를 제한한다.

NoteBase

문 2. 시스템 접근을 허락받은 후에 그 시스템의 어떤 기능 또는 서비스를 이용할 수 있도록 필요한 권한을 부여하는 것은?

① 식별(identification)
② 인증(authentication)
③ 인가(authorization)
④ 평가(evaluation)

정답 3번

간단한 해설

시스템에 접근한 사용자가 **어떤 기능이나 서비스를 이용할 수 있는지 권한을 부여**하는 것이 인가(authorization)이다.

공통 해설

인가(authorization)는 사용자에게 특정 자원이나 기능을 이용할 권한을 부여하는 것이다. 문항은 사용자가 누구인지 확인하는 것이 아니라, 접근을 허락받은 사용자가 **무엇을 할 수 있는지**를 묻고 있으므로 인가의 정의에 해당한다.

예를 들어 시스템에 로그인한 사용자에게 문서 조회만 허용하고 수정은 허용하지 않을 수 있다. 이때 조회·수정 등 행위별로 이용 범위를 정하는 것이 인가이다. 시스템에 접속할 수 있어도 모든 기능을 사용할 권한까지 자동으로 주어지는 것은 아니다.

NoteBase

문 3. 컴퓨터 기반 사회공학적 공격기법에 해당하지 않는 것은?

① 피싱(phishing)

② 파밍(pharming)

③ 스미싱(smishing)

④ 스푸핑(spoofing)

정답 4번

간단한 해설

이 문항에서는 스푸핑을 IP 주소 등 통신정보를 위조하는 **기술적 공격으로 구분하여 제외**한다.
피싱·파밍·스미싱은 정상적인 안내나 사이트로 믿게 하여 정보 제공이나 위험한 행동을 유도하는 공격으로 묶는다. 다만 스푸핑도 사용자 기만에 이용될 수 있어 두 범주가 완전히 배타적인 것은 아니다.

NoteBase

개념 구분

컴퓨터 기반 사회공학적 공격은 이메일·웹사이트·문자메시지 등을 이용하여 사람을 속이고, 민감한 정보를 제공하거나 악성 파일을 실행하도록 유도하는 공격이다. 기술을 사용하는지 자체보다 **사용자의 신뢰와 판단을 악용하는지**가 핵심이다.

① 피싱(phishing): 신뢰할 만한 기관이나 사람을 사칭한 이메일·웹사이트 등으로 개인정보나 금융정보의 제공을 유도한다. 사용자가 사칭한 대상을 진짜로 믿고 정보를 넘기게 한다는 점에서 사회공학적 공격에 해당한다.

② 파밍(pharming): 악성코드나 도메인 이름을 IP 주소로 바꾸는 DNS 정보의 변조 등을 통해 사용자를 가짜 사이트로 연결한다. **정상 주소를 입력해도 가짜 사이트로 연결될 수 있다**는 점이 특징이다. 접속 경로를 바꾸는 기술적 조작과, 가짜 사이트를 정상 사이트로 믿고 정보를 입력하게 하는 사용자 기만이 결합된다. 이 문항에서는 이러한 기만을 포함하는 공격으로 분류한다.

③ 스미싱(smishing): 문자메시지(SMS)를 이용하는 피싱이다. 택배 안내나 무료 쿠폰 등으로 위장하여 링크 접속, 악성 앱 설치 또는 정보 입력을 유도한다. 문자 내용에 대한 사용자의 신뢰를 악용한다는 점에서 사회공학적 공격에 해당한다.

④ 스푸핑(spoofing): 신원이나 출처 정보를 위조하여 다른 대상인 것처럼 가장하는 기법이다. 대표적인 IP 스푸핑은 패킷의 출발지 IP 주소를 위조하며, **사용자를 설득하는 과정 없이도 수행될 수 있다**. 이런 통신정보 위조 공격으로 해석하면 사회공학적 공격과 구별된다.

공식 채점 답은 ④이며, 이는 스푸핑을 통신정보 위조 중심의 기술적 공격으로 구분한 선택이다. 다만 원문은 스푸핑의 종류를 한정하지 않았다. 이메일 발신자 위조처럼 사람을 속이는 데 쓰이는 스푸핑도 있으므로, 모든 스푸핑이 사회공학적 공격과 무관하다는 뜻으로 일반화할 수는 없다.

NoteBase

문 4. 다음의 명령어를 실행한 파일의 접근 권한으로 옳은 것은?

chmod 751 test.c

① -rwxr-x--x

② -rwxrw---x

③ -rwxr-x---

④ -rw-r-----

정답 1번

간단한 해설

751은 **소유자·그룹·기타 사용자** 순서로 권한을 지정한다. 읽기 4·쓰기 2·실행 1을 합산하는 방식이므로, 7은 rwx, 5는 r-x, 1은 --x가 된다.

계산 과정

chmod는 파일의 접근 권한을 변경하는 명령이다. 'chmod 751 test.c'의 751은 소유자·그룹·기타 사용자에게 부여할 권한을 순서대로 나타낸다. 각 자릿값은 **읽기(r) 4, 쓰기(w) 2, 실행(x) 1** 중 허용할 권한의 값을 더한 것이다. 문자 표기는 r·w·x 순서이며, 허용하지 않는 권한은 '-'로 표시한다.

소유자: 7 = 4 + 2 + 1이므로 읽기·쓰기·실행을 모두 허용한다. 문자로는 rwx이다.

그룹: 5 = 4 + 1이므로 읽기와 실행을 허용하고 **쓰기는 허용하지 않는다**. 문자로는 r-x이다.

기타 사용자: 1은 실행 권한의 값이므로 **실행만 허용**한다. 읽기와 쓰기는 허용하지 않아 문자로는 --x이다.

세 권한 묶음을 순서대로 연결하면 rwxr-x--x이다. 선택지 맨 앞의 '-'는 일반 파일임을 나타내는 파일 종류 표시이므로, 전체 표기는 -rwxr-x--x로 ①과 일치한다.

NoteBase

문 5. FTP 보안에 대한 설명으로 옳지 않은 것은?

① 임의의 계정으로 로그인 시도를 반복적으로 수행하여 사용자 계정의 패스워드를 유추할 수 있는 취약점이 있다.

② 사용자 인증정보 유출 방지를 위한 보안 대책으로 SCP, SFTP, FTPS 등이 있다.

③ 익명 FTP는 모든 사용자에게 동일한 사용자 아이디와 유효한 사용자 이메일 주소를 패스워드로 요구한다.

④ FTP 보안 대책으로 반드시 서비스 사용이 필요하지 않은 경우 FTP 서비스 사용을 금지하는 것이 좋다.

정답 3번

간단한 해설

익명 FTP에서는 보통 'anonymous' 계정을 사용하며, 이메일 주소를 패스워드로 입력하는 것은 관행이다. **유효한 이메일 주소가 반드시 필요한 것은 아니며**, 서버 설정에 따라 임의의 문자열도 허용된다.

선지별 해설

① O: 로그인 시도 횟수를 제한하지 않으면 공격자가 특정 계정에 여러 패스워드 후보를 반복 입력하여 패스워드를 알아낼 수 있다. 이러한 **무차별 대입 공격**을 줄이려면 인증 실패 횟수를 제한하거나 실패 시 응답을 지연시키는 대책이 필요하다.

② O: 일반 FTP는 사용자 이름과 패스워드를 평문으로 전송하므로 통신을 도청당하면 인증정보가 노출될 수 있다. SCP와 SFTP는 SSH를, FTPS는 TLS를 이용하여 **인증정보가 전달되는 통신 구간을 암호화**하므로 이러한 유출 위험을 줄인다.

③ X: 익명 FTP는 개인별 계정을 발급받지 않고 보통 공용 계정인 'anonymous'로 공개 파일에 접근하는 방식이다. 이메일 주소를 패스워드로 입력하도록 안내하는 경우가 있지만, **유효한 이메일 주소를 반드시 요구하는 것은 아니다**. 전통적으로 임의의 문자열을 허용하며, 이메일 주소 입력 여부나 형식 검사 등은 서버 정책에 따라 달라진다.

④ O: 사용하지 않는 FTP 서비스를 계속 실행하면 외부에서 접근할 수 있는 공격 경로가 남는다. 서비스가 필요하지 않다면 **FTP 서비스를 중지하고 비활성화**하여 불필요한 접근과 취약점 악용 가능성을 줄이는 것이 좋다.

NoteBase

문 6. SSL 프로토콜 스택에 포함되지 않는 것은?

① handshake 프로토콜

② alert 프로토콜

③ record 프로토콜

④ user authentication 프로토콜

정답 4번

간단한 해설

SSL의 인증 기능은 **handshake 프로토콜**에서 수행한다. 'user authentication 프로토콜'은 SSL 스택을 구성하는 별도의 프로토콜 명칭이 아니다.

공통 해설

SSL 프로토콜 스택은 **record 프로토콜과 그 위에서 동작하는 제어 프로토콜**로 구성된다. record는 상위 메시지를 레코드 단위로 나누어 전달하고, 협상된 보안 설정에 따라 암호화와 무결성 보호를 적용한다. 그 위에는 보안 통신에 필요한 조건을 협상하는 handshake, 오류나 연결 종료를 알리는 alert, 협상한 암호 설정의 적용을 알리는 change cipher spec이 있다. 따라서 handshake·alert·record는 모두 SSL 스택의 구성 요소이다.

SSL에서 통신 상대의 신원을 확인하는 인증은 **handshake에 포함된 기능**이다. 인증서를 사용하는 방식에서는 handshake 과정에서 인증서를 교환하며, 클라이언트 인증은 선택적으로 이루어진다. 인증 기능이 있다는 사실이 'user authentication'이라는 별도 SSL 구성 프로토콜이 있다는 뜻은 아니므로, 이 명칭은 SSL 스택에 포함되지 않는다.

NoteBase

문 7. 전자우편의 보안성 향상을 위해 개발된 것이 아닌 것은?

① SMTP

② PGP

③ S/MIME

④ PEM

정답 1번

간단한 해설

SMTP는 **전자우편을 전송·중계하는 프로토콜**이다. 전자우편 메시지에 암호화와 전자서명 등의 보호 기능을 제공하기 위해 개발된 보안 방식과는 목적이 다르다.

공통 해설

전자우편의 보안성 향상을 위한 방식은 메시지를 암호화하여 내용을 숨기는 기밀성, 전자서명으로 발신자를 확인하는 인증과 내용의 변경 여부를 확인하는 무결성 등의 보호 기능을 제공한다. PGP, S/MIME, PEM은 모두 이러한 **메시지 보호 기능**을 제공하는 전자우편 보안 방식에 해당한다.

SMTP의 기본 목적은 **전자우편의 전송·중계**이다. 기본 SMTP 자체가 메시지 암호화나 전자서명을 제공하는 것은 아니므로, 전자우편 보안성 향상을 위해 개발된 방식에 해당하지 않는다.

NoteBase

문 8. 일반적으로 이메일 형식으로 전달되며, 이메일 혹은 게시판 등에 거짓정보나 소문 등을 실어 사용자를 겁주거나 속이는 것은?

① 혹스(Hoax)

② 트로이 목마(Trojan horse)

③ 백도어(Backdoor)

④ 스파이웨어(Spyware)

정답 1번

간단한 해설

거짓정보나 소문을 퍼뜨려 사용자를 겁주거나 속이는 것은 **혹스(Hoax)**의 특징이다.

공통 해설

혹스(Hoax)는 사실이 아닌 경고나 소문을 이메일·게시판 등에 퍼뜨려 사용자를 속이거나 불안하게 하는 허위 정보이다. 예를 들어 존재하지 않는 위험한 바이러스를 경고하며 다른 사람에게도 그 메시지를 전달하라고 요구하는 경우가 이에 해당한다. 문항에서는 이메일이라는 전달 매체보다 **거짓정보로 사용자를 겁주거나 속인다**는 내용이 혹스를 식별하는 기준이다.

NoteBase

문 9. 클라이언트 측에서 웹사이트에 접속할 때 발생하는 HTTP 에러 코드와 이의 원인을 설명한 것으로 옳지 않은 것은?

① 401 Unauthorized: 특정 웹사이트에 접속하기 위해 정확한 사용자 아이디와 암호를 입력하여야 하는데, 잘못된 정보를 입력하였을 경우

② 403 Forbidden: 다른 요청이나 서버의 구성과 충돌이 발생할 경우

③ 404 Not Found: URL이나 링크가 변경되어 요청한 주소의 페이지가 없을 경우

④ 414 Request-URI Too Long: 요청에 사용된 URL이 서버가 감당할 수 없을 만큼 너무 길 경우

정답 2번

간단한 해설

403 Forbidden은 서버가 요청을 이해했지만 **접근을 거부**한다는 뜻이다. 요청이 대상 자원의 현재 상태와 충돌하여 처리되지 못하는 경우는 **409 Conflict**에 해당한다.

선지별 해설

① O: 401 Unauthorized는 인증이 필요한 자원에 대한 요청에 **유효한 인증 정보가 없는 경우**를 나타낸다. 필요한 사용자 아이디나 암호를 잘못 입력하여 인증에 실패한 상황이 이에 해당한다.

② X: 403 Forbidden은 서버가 요청을 이해했지만 권한 부족이나 접근 제한 등의 이유로 **요청 수행을 거부**하는 경우이다. 선지의 '충돌'은 403의 의미가 아니다. 요청이 대상 자원의 현재 상태와 충돌하여 완료될 수 없는 경우에는 **409 Conflict**를 사용한다.

③ O: 404 Not Found는 요청한 주소에서 **해당 자원을 찾을 수 없는 경우**를 나타낸다. 페이지 주소가 바뀌었는데 기존 URL이나 링크로 요청하여 그 주소에 페이지가 없다면 이에 해당한다.

④ O: 414 Request-URI Too Long은 **요청 URI가 서버가 처리할 수 있는 길이를 초과**한 경우를 나타낸다. 요청 URL에 매개변수 등이 지나치게 길게 붙어 서버의 허용 길이를 넘는 상황이 이에 해당한다.

NoteBase

문 10. 다음에서 설명하는 윈도우 NTFS 파일시스템의 구조는?

모든 파일 및 디렉터리에 대한 정보가 저장된다. 즉, 정보는 파일 내용들을 정의하는 속성들의 집합으로 구성된다.

① PBS
② MFT
③ 시스템 파일
④ 백업 수퍼 블록

정답 2번

간단한 해설

NTFS는 모든 파일과 디렉터리의 정보를 **MFT(Master File Table, 마스터 파일 테이블)**에서 관리한다. 각 파일의 정보는 이름·시간 정보·데이터 등을 나타내는 속성들의 집합으로 구성된다.

공통 해설

MFT(Master File Table)는 NTFS 볼륨의 모든 파일과 디렉터리에 대한 레코드를 관리하는 구조이다. 각 레코드는 **파일 정보를 나타내는 속성들의 집합**으로 구성되며, 파일 이름·크기·시간 정보뿐 아니라 파일의 데이터도 속성으로 다룬다. 따라서 지문의 '모든 파일 및 디렉터리에 대한 정보'와 '속성들의 집합'은 MFT의 관리 방식에 해당한다.

예를 들어 문서 파일 하나를 저장하면 MFT의 해당 파일 레코드를 통해 파일 이름과 데이터에 관한 정보를 찾을 수 있다. 데이터가 작으면 레코드 안에 직접 저장할 수 있고, 레코드 밖에 저장되는 경우에는 그 데이터를 찾을 수 있는 위치 정보를 기록한다. 즉, MFT가 모든 파일의 정보를 관리한다는 말이 **모든 파일의 내용 전체가 MFT 안에 저장된다는 뜻은 아니다**.

NoteBase

문 11. <보기 1>의 상황과 개인정보의 안전한 전달을 위해 제공되어야 할 <보기 2>의 정보보호서비스를 바르게 연결한 것은?

<보기 1>
ㄱ. 갑은 송신하는 개인정보를 도청당하는 일이 없이 을에게 전달하기 원한다.
ㄴ. 갑은 송신하는 개인정보가 조작당하는 일이 없이 을에게 전달되기 원한다.
ㄷ. 갑은 통신 상대의 웹 서버가 진짜 을의 서버라는 것을 확인하고 싶다.
ㄹ. 갑은 을의 서버에 적절한 시간에 접속하여 정상적으로 요청된 내용을 수행하고 싶다.
<보기 2>
A. 기밀성
B. 무결성
C. 인증
D. 가용성

① ㄱ: A, ㄴ: B, ㄷ: C, ㄹ: D
② ㄱ: A, ㄴ: D, ㄷ: C, ㄹ: B
③ ㄱ: B, ㄴ: A, ㄷ: C, ㄹ: D
④ ㄱ: B, ㄴ: A, ㄷ: D, ㄹ: C

정답 1번

간단한 해설

개인정보를 **허가 없이 읽지 못하게 하는 것**은 기밀성이고, **내용의 무단 변경을 막는 것**은 무결성이다. 도청 방지와 조작 방지는 서로 다른 정보보호 요구이다.

NoteBase

대응 관계 해설

ㄱ → A(기밀성): 기밀성은 허가받지 않은 사람에게 정보가 노출되지 않도록 하는 성질이다. 갑이 보내는 개인정보를 제삼자가 도청하여 알 수 없게 하려는 것이므로, 보호할 대상은 **정보 내용의 비밀**이다.

ㄴ → B(무결성): 무결성은 정보가 허가 없이 변경되거나 훼손되지 않도록 하는 성질이다. 갑이 보낸 개인정보가 전송 중 조작되지 않고 을에게 전달되어야 하므로, 핵심은 **보낸 내용이 그대로 유지되는 것**이다. 정보가 노출되지 않더라도 내용이 바뀌면 무결성은 침해된다.

ㄷ → C(인증): 인증은 대상이 주장하는 신원이 실제 신원과 일치하는지 확인하는 것이다. 접속한 웹 서버가 을을 사칭하는 서버가 아니라 **실제로 을의 서버인지 확인**하려는 상황이므로 서버 인증이 필요하다.

ㄹ → D(가용성): 가용성은 정당한 사용자가 필요한 때에 정보나 서비스를 정상적으로 이용할 수 있도록 하는 성질이다. 갑이 적절한 시간에 을의 서버에 접속하고 요청한 기능을 수행하려면 **필요한 시점의 접근과 정상 이용**이 보장되어야 한다.

따라서 ㄱ–A, ㄴ–B, ㄷ–C, ㄹ–D로 연결되며, ①의 조합과 일치한다.

NoteBase

문 12. 운영체제에서 제공하는 파일 및 디렉터리 관리에 대한 설명으로 옳지 않은 것은?

① 윈도우즈 NT 계열의 운영체제는 파일과 디렉터리에 대한 접근 제어를 통제하기 위하여 NTFS를 사용한다.

② 윈도우즈 NTFS는 모든 권한, 수정, 읽기 및 실행, 폴더내용 보기, 읽기, 쓰기와 같은 6가지 권한을 설정하여 운영한다.

③ 유닉스 계열은 파일이나 디렉터리 등의 자원에 대한 접근 제어를 위해 소유권과 접근 권한을 할당한다.

④ 유닉스 계열에서는 ‘etc/passwd’ 파일이나 ‘shadow’ 파일의 읽기 및 쓰기 권한을 일반 사용자에게 부여해도 안전하다.

정답 4번

간단한 해설

④의 '일반 사용자에게 읽기 및 쓰기 권한을 부여해도 안전하다'는 설명이 잘못되었다. 계정 정보가 담긴 passwd 파일은 일반 사용자의 읽기가 허용되지만 **쓰기는 제한**해야 하며, 비밀번호 해시가 담긴 shadow 파일은 **일반 사용자의 읽기와 쓰기를 모두 제한**해야 한다.

선지별 해설

① O: NTFS는 윈도우즈 NT 계열에서 사용하는 파일 시스템으로, 파일과 디렉터리에 **사용자·그룹별 접근 권한**을 설정할 수 있다. 접근 제어 목록(ACL)에 어떤 사용자가 해당 자원을 읽거나 변경할 수 있는지 기록하여 접근을 통제한다.

② O: NTFS의 기본 권한은 모든 권한, 수정, 읽기 및 실행, 폴더 내용 보기, 읽기, 쓰기의 6가지로 구분한다. '폴더 내용 보기'는 폴더에 적용되는 권한이다. '수정'은 읽기·쓰기·실행과 삭제를 허용하며, '모든 권한'에는 여기에 **권한 변경과 소유권 가져오기**까지 포함된다. 기본 권한은 여러 세부 권한을 묶어 제공하는 것이다.

③ O: 유닉스 계열에서는 파일과 디렉터리에 소유 사용자와 소유 그룹을 지정하고, **소유자·그룹·기타 사용자**를 구분하여 읽기(r), 쓰기(w), 실행(x) 권한을 설정한다. 따라서 같은 파일이라도 소유자는 읽고 쓸 수 있고, 기타 사용자는 읽기만 할 수 있도록 접근을 통제할 수 있다.

④ X: passwd 파일에는 사용자 이름, 사용자 ID, 홈 디렉터리 등 계정 정보가 저장된다. 일반 사용자가 이 정보를 읽는 것은 허용되지만, 임의로 변경하면 계정 정보가 변조될 수 있으므로 **일반 사용자에게 쓰기 권한을 주어서는 안 된다**. shadow 파일에는 비밀번호 해시와 비밀번호 사용 기간 등의 정보가 저장된다. 해시가 노출되면 비밀번호 추측 공격에 악용될 수 있으므로, shadow 파일은 일반 사용자의 읽기와 쓰기를 모두 제한하고 필요한 특권 계정이나 인증 프로그램만 접근하게 해야 한다.

NoteBase

문 13. 패스워드 공격에 대한 설명으로 옳지 않은 것은?

① 사용자의 패스워드는 암호화하여 저장하는 것이 안전하다.

② 패스워드를 알아내기 위하여 사용자의 신원이나 주변 정보로 패스워드를 알아내는 사회공학적 방법이 있다.

③ Brute force 공격은 사용자가 패스워드를 입력할 때 가로채는 공격이다.

④ Crypt() 함수를 이용하여 패스워드를 추측할 수 있다.

정답 3번

간단한 해설

Brute force 공격은 입력 중인 패스워드를 가로채는 것이 아니라 **가능한 패스워드 조합을 하나씩 대입**하는 공격이다. 입력 내용을 몰래 기록해 얻는 방식은 키로깅 등에 해당한다.

선지별 해설

① O: 패스워드를 평문으로 저장하면 저장 정보가 유출될 때 원래 패스워드가 그대로 드러난다. 선지의 '암호화하여 저장'은 이러한 평문 노출을 막는다는 취지이다. 실제 패스워드 저장에는 복호화할 수 있는 암호문보다 **솔트를 적용한 패스워드 전용 해시값**을 사용하며, 로그인 때 입력값을 같은 방식으로 처리하여 저장값과 비교한다.

② O: 사회공학적 방법은 사람의 심리나 행동, 신뢰 관계 등을 이용해 정보를 얻는 방법이다. 사용자의 이름·생일·가족 정보 등을 단서로 패스워드를 추측하거나, 신뢰할 만한 사람을 사칭해 패스워드를 알아내는 방식이 이에 해당한다. 암호 알고리즘 자체보다 **사람과 관련된 취약점**을 이용한다.

③ X: Brute force 공격은 가능한 문자 조합을 차례로 대입하여 일치하는 패스워드를 찾는 **무차별 대입 공격**이다. 사용자가 입력하는 순간의 내용을 가로채는 방식과는 다르다. 예를 들어 키로깅은 키보드 입력을 몰래 기록하여 패스워드를 얻는 방식이다.

④ O: Crypt()는 주어진 패스워드 후보와 솔트 등을 이용해 일방향 해시값을 만드는 함수이다. 공격자가 저장된 해시값을 확보했다면, 후보를 같은 조건으로 처리한 뒤 **저장된 해시값과 비교**하여 추측을 확인할 수 있다. Crypt()가 저장값을 복호화해 원래 패스워드를 직접 돌려주는 것은 아니다.

NoteBase

문 14. 보안이 취약한 웹 게시판이 저장 XSS 공격을 받았다고 가정했을 때, 이를 해결하기 위한 방법으로 가장 적절한 것은?

① 가상 사설망을 통해서만 사용자 게시판에 접근하도록 한다.

② 접근 권한을 설정하여 허가된 사용자만 글을 올릴 수 있게 한다.

③ 사용자 게시글 속에 있는 악성 스크립트 코드를 찾아서 제거한다.

④ SSL을 사용하여 사용자가 게시글을 올릴 수 있게 한다.

정답 3번

간단한 해설

저장 XSS는 게시판에 저장된 악성 스크립트가 글을 읽는 사용자의 브라우저에서 실행되는 공격이다. 따라서 이미 저장된 게시글에서 **악성 스크립트를 제거**하여 실행 원인을 없애야 한다.

선지별 해설

저장 XSS는 공격자가 게시글 등에 악성 스크립트를 저장하고, 다른 사용자가 해당 글을 열면 그 코드가 사용자의 브라우저에서 실행되게 하는 공격이다. 게시판에 접근할 수 있는 사람이나 통신 경로를 제한하는 것만으로는 저장된 코드의 실행을 막을 수 없다.

① X: 가상 사설망(VPN)은 게시판에 접속할 수 있는 네트워크 경로를 제한한다. 그러나 VPN을 이용하는 사용자가 악성 게시글을 열면 스크립트가 실행될 수 있으므로, **접속 경로 제한만으로는 해결되지 않는다**. 이미 저장된 악성 코드를 제거하고 게시글이 표시될 때 실행되지 않도록 처리해야 한다.

② X: 글쓰기 권한은 누가 게시글을 등록할 수 있는지 통제한다. 그러나 **허가된 사용자의 입력도 악성 코드를 포함할 수 있다**. 이미 저장된 악성 게시글도 그대로 남으므로, 권한 설정과 별도로 게시글의 위험한 내용을 제거하고 안전하게 출력해야 한다.

③ O: 게시글에 저장된 악성 스크립트를 찾아 제거하면 다른 사용자가 글을 열 때 실행될 **공격 코드 자체를 없앨 수 있다**. 재발을 막으려면 입력 내용에서 위험한 HTML 요소·속성 등을 정제하고, 출력 위치에 맞게 인코딩하여 사용자 입력이 실행 코드로 해석되지 않도록 해야 한다.

④ X: SSL을 이용한 암호화 통신은 전송 중인 데이터의 도청이나 변조를 방지하는 수단이다. 그러나 **전송 내용의 악성 여부를 판별하거나 제거하지는 않는다**. 악성 스크립트도 암호화된 채 전달되어 저장될 수 있으므로, 게시글 자체에 대한 정제와 안전한 출력 처리가 필요하다.

NoteBase

문 15. 괄호 안에 들어갈 웹의 취약점은?

( )는(은) 불특정 다수를 대상으로 로그인된 사용자가 자신의 의도와는 무관하게 공격자가 의도한 행위(수정, 삭제, 등록, 송금) 등을 하게 만드는 공격이다.

① 명령 삽입 취약점
② XSS 취약점
③ 디렉터리 리스팅 취약점
④ CSRF 취약점

정답 4번

간단한 해설

로그인된 사용자의 **인증 상태를 악용**하여 본인이 원하지 않는 수정·삭제·송금 등의 요청을 보내게 하는 공격은 **CSRF(사이트 간 요청 위조)**이다. 핵심은 사용자가 로그인되어 있다는 사실과 해당 행위를 실제로 원한다는 사실이 다르다는 점이다.

공통 해설

CSRF(Cross-Site Request Forgery, 사이트 간 요청 위조)는 공격자가 로그인된 사용자의 브라우저를 통해 위조한 요청을 보내어, 해당 사용자의 권한으로 원하지 않는 작업이 처리되게 하는 공격이다. 판단 기준은 **사용자의 인증 상태를 이용한 의도하지 않은 요청**이다. 지문의 '로그인된 사용자', '자신의 의도와는 무관하게', '수정, 삭제, 등록, 송금'은 이 정의에 대응하므로 빈칸에는 CSRF 취약점이 들어간다.

예를 들어 사용자가 어떤 사이트에 로그인한 상태에서 공격자가 만든 페이지에 접속했다고 하자. 그 페이지가 원래 사이트로 정보 변경 요청을 보내고, 브라우저가 그 요청에 로그인 인증용 쿠키를 함께 전송하면, 방어가 미흡한 서버는 이를 사용자가 요청한 변경으로 받아들일 수 있다. 이때 **로그인 여부만으로 요청의 진정한 의도를 보장할 수 없다**는 점이 공격의 핵심이다.

NoteBase

문 16. 다음은 스택 버퍼 오버플로우 공격을 효과적으로 방어하기 위한 스택 보호 메커니즘을 서술한 것이다. ㉠~㉢에 들어갈 말을 바르게 연결한 것은?

스택가드(stackguard)는 가장 잘 알려진 보호 메커니즘 중 하나이다. 이것은 GCC 컴파일러의 확장 버전으로 추가의 함수 진입과 종료 코드를 삽입한다. 추가되는 함수 진입 코드는 지역변수를 위한 공간을 할당하기 전에 이전 (㉠) 주소 앞에 (㉡) 값을 기록한다. 추가되는 함수 종료 코드는 이전 (㉠)를 복원하고 제어를 (㉢)로 이동하는 보통의 함수 종료 연산을 수행하기 전에 (㉡) 값이 변경되었는지를 검사한다. 전통적인 스택 버퍼 오버플로우 시도는 이전 (㉠)와 (㉢)를 변경하기 위해 (㉡) 값을 바꾸어야 하는데, 만약 변경되었다면 프로그램을 종료하게 된다.

① ㉠: 스택포인터, ㉡: 반환주소, ㉢: 카나리아(canary)

② ㉠: 스택포인터, ㉡: 카나리아(canary), ㉢: 반환주소

③ ㉠: 프레임 포인터, ㉡: 카나리아(canary), ㉢: 반환주소

④ ㉠: 프레임 포인터, ㉡: 반환주소, ㉢: 스택포인터

정답 3번

간단한 해설

스택가드는 **카나리아 값의 변조**를 검사하여 전통적인 스택 버퍼 오버플로우를 감지한다. 함수 종료 시 복원하는 것은 이전 프레임 포인터이며, 실행을 이어 갈 위치는 **반환주소**가 나타낸다.

빈칸별 해설

㉠ 프레임 포인터: 함수의 스택 프레임에서 지역변수 등에 접근할 때 기준이 되는 위치를 가리킨다. 함수 진입 시 이전 프레임 포인터를 저장하고, 종료 시 이를 복원하여 호출한 함수의 프레임을 다시 사용한다. 현재 스택의 끝을 가리키는 스택포인터와는 **역할이 다르다**.

㉡ 카나리아(canary): 스택의 손상 여부를 감지하기 위해 넣는 검사값이다. 지문에서는 지역변수 영역과 저장된 이전 프레임 포인터 사이에 배치한다. 버퍼를 넘어서 연속으로 덮어쓰는 전통적인 공격이 이전 프레임 포인터나 반환주소에 도달하려면 이 검사값도 지나게 된다. 함수 종료 코드는 **카나리아가 처음 기록한 값과 달라졌는지** 확인하고, 달라졌으면 정상 복귀 대신 프로그램을 종료한다.

㉢ 반환주소: 호출된 함수가 끝난 뒤 **실행을 이어 갈 명령의 주소**이다. 카나리아 검사에 이상이 없으면 이전 프레임 포인터를 복원하고 반환주소로 제어를 넘긴다. 카나리아는 변조를 검사할 데이터이고, 스택포인터는 스택 위치를 관리하는 포인터이므로 이 복귀 목적지에 해당하지 않는다.

㉠ 프레임 포인터, ㉡ 카나리아, ㉢ 반환주소의 조합은 ③과 일치한다.

NoteBase

문 17. 응용 보안에 대한 설명으로 옳지 않은 것은?

① HTTPS는 웹 브라우저와 웹 서버 간의 안전한 통신을 구현하기 위해 HTTP와 SSL을 결합한 것이다.

② SET는 웹 보안을 위하여 메시지 기밀성은 제공되지만, 메시지 무결성은 제공되지 않는다.

③ 공개키기반구조(PKI)는 전자서명, 전자상거래 등이 안전하게 구현되기 위하여 구축되어야 할 기반 기술이다.

④ 스팸메일의 문제점은 인터넷망을 통해 무차별로 전송되어 원하지 않는 사람이 읽거나 처리하는 데 많은 시간과 비용을 낭비하게 된다는 것이다.

정답 2번

간단한 해설

SET는 전자상거래의 카드 결제를 보호하는 프로토콜로, 메시지의 **기밀성과 무결성을 모두 제공**한다. 따라서 '메시지 무결성은 제공되지 않는다'는 설명이 잘못되었다.

선지별 해설

① O: HTTPS는 HTTP 통신을 **SSL로 보호**하여 웹 브라우저와 웹 서버가 안전하게 데이터를 주고받도록 하는 방식이다. 암호화로 통신 내용의 노출을 막고, 무결성 검증으로 전송 중 데이터의 변조를 탐지한다.

② X: SET(Secure Electronic Transaction)는 전자상거래에서 카드 결제를 안전하게 처리하기 위한 보안 프로토콜이다. 암호화로 결제정보의 기밀성을 보호하고, **전자서명으로 메시지의 무결성도 제공**한다. 기밀성은 권한 없는 사람이 내용을 읽지 못하게 하는 성질이고, 무결성은 내용이 변조되었는지를 확인할 수 있게 하는 성질이다. 따라서 '메시지 무결성은 제공되지 않는다'는 '메시지 무결성도 제공된다'로 고쳐야 한다.

③ O: 공개키기반구조(PKI)는 **공개키와 소유자의 신원을 연결하는 인증서**를 발급·관리·검증하는 기반이다. 전자서명을 검증할 때 사용한 공개키가 실제 서명자의 것인지 확인할 수 있게 하므로, 전자서명과 전자상거래를 안전하게 구현하는 데 필요한 신뢰 기반이 된다.

④ O: 스팸메일은 수신자가 원하지 않는 메일을 대량으로 전송하므로, 수신자가 이를 확인하거나 분류·삭제하는 데 시간과 비용을 쓰게 한다. 메일의 전송과 보관에도 네트워크·서버 자원이 소모되어 **수신자와 서비스 운영자의 부담**이 커진다.

NoteBase

문 18. 트립와이어(tripwire)에 대한 설명으로 옳지 않은 것은?

① 파일의 무결성을 검사하는 도구이며, 해쉬 알고리즘을 이용하여 시스템에 존재하는 파일에 관한 정보를 데이터베이스화한다.

② 해커의 침입으로 인한 시스템 파일이나 디렉터리의 변경을 쉽게 검출할 수 있도록 도와준다.

③ 데이터베이스에 저장된 해쉬 결과 값과 현재 파일의 해쉬 결과 값을 비교하여 무결성 여부를 판단한다.

④ 트립와이어의 데이터베이스에는 파일의 해쉬 결과 값이 저장되어 있어서 물리적 보안 대책이 필요 없다.

정답 4번

간단한 해설

해시값은 파일의 변경을 확인하는 비교 자료이며, 저장 매체에 대한 무단 접근을 막아 주지는 않는다. 따라서 기준 데이터베이스를 신뢰할 수 있도록 **물리적 보안 대책도 필요하다**.

선지별 해설

① O: 트립와이어는 파일이 허가 없이 변경되었는지 확인하는 무결성 검사 도구이다. 정상 상태에서 검사 대상 파일의 **해시값과 파일 속성을 기준 데이터베이스에 저장**하고, 이후 검사에 활용한다.

② O: 검사 대상으로 지정한 시스템 파일과 디렉터리의 현재 상태를 기준 데이터베이스와 비교하여 변경을 검출한다. 해커가 파일 내용을 바꾸거나 디렉터리의 파일을 추가·삭제하면 그 차이를 보고하여 침입 흔적을 찾는 데 도움을 준다.

③ O: 검사 시 현재 파일의 해시값을 다시 계산하여 데이터베이스에 저장된 기준 해시값과 비교한다. **해시값이 다르면 파일 내용이 변경되었음**을 알 수 있다. 다만 변경이 정상적인 업데이트인지 침입에 따른 것인지는 별도로 확인해야 한다.

④ X: 해시값을 저장한다는 이유로 물리적 보안 대책이 불필요해지는 것은 아니다. **기준 데이터베이스 자체도 변조·삭제로부터 보호**해야 신뢰할 수 있는 비교가 가능하다. 따라서 서버와 저장 매체에 대한 물리적 접근 통제 등 보안 대책이 필요하다.

NoteBase

문 19. 괄호 안에 들어갈 말로 옳은 것은?

운영체제의 구조는 이중 모드(dual mode)로 되어있는데, 이는 사용자 모드(user mode)와 커널 모드(kernel mode, 또는 운영체제 실행 모드)이다. 이 중 사용자 모드는 특권 명령어를 사용할 수 없으며, 이러한 경우에 사용자 프로세스는 운영체제에게 도움을 요청하게 되는데, 이를 ( )(이)라 한다. 즉, ( )는(은) 실행 중인 프로그램과 운영체제 사이에 인터페이스를 제공하는 것이다.

① 시스템 관리(system management)

② 시스템 호출(system call)

③ 프로세스 관리(process management)

④ 스케줄링(scheduling)

정답 2번

간단한 해설

사용자 프로세스가 직접 수행할 수 없는 작업을 운영체제에 요청하는 인터페이스가 **시스템 호출**이다. 요청된 작업은 커널 모드에서 운영체제가 처리한다.

공통 해설

시스템 호출(system call)은 실행 중인 프로그램이 **운영체제의 서비스를 요청하는 인터페이스**이다. 사용자 모드에서는 특권 명령어를 직접 실행할 수 없으므로, 사용자 프로세스는 시스템 호출을 통해 커널의 기능을 이용한다. 따라서 지문의 두 괄호에는 모두 시스템 호출이 들어간다.

시스템 호출이 이루어지면 커널 모드로 전환되어 운영체제가 요청을 처리하고, 처리가 끝나면 사용자 모드로 돌아와 프로그램 실행을 이어 간다. 예를 들어 프로그램이 파일의 내용을 읽으려 할 때, 직접 저장장치를 제어하는 대신 시스템 호출로 읽기를 요청하면 운영체제가 해당 작업을 수행한다. 핵심은 관리 기능의 이름이 아니라 **프로그램에서 커널로 서비스를 요청하는 통로**라는 점이다.

NoteBase

문 20. 코드 보안과 관련된 설명으로 옳지 않은 것은?

① 버퍼 오버플로우 공격은 데이터 길이에 대한 불명확한 정의를 이용한 공격이다.

② gets()는 버퍼 오버플로우 공격에 취약하지 않은 함수이다.

③ 포맷 스트링 공격은 데이터 형태에 대한 불명확한 정의로 발생한다.

④ 버퍼 오버플로우 공격 방어 방법으로는 공격에 취약한 함수를 사용하지 않거나 최신 운영체제를 사용하는 것 등이 있다.

정답 2번

간단한 해설

gets()는 **입력 길이를 버퍼 크기로 제한하지 않는** 함수이다. 버퍼에 저장할 수 있는 양보다 긴 입력을 받으면 버퍼 밖의 메모리까지 덮어쓸 수 있으므로 버퍼 오버플로우에 취약하다.

선지별 해설

① O: 버퍼 오버플로우는 버퍼에 할당된 공간보다 많은 데이터를 기록하여 경계 밖의 메모리까지 덮어쓰는 현상이다. 선지의 길이에 대한 불명확한 정의는 **입력 길이를 버퍼 용량에 맞게 제한·검사하지 않는 것**을 뜻한다. 공격자는 이를 이용해 인접한 데이터나 프로그램의 제어에 쓰이는 정보를 손상시킬 수 있다.

② X: gets()는 **버퍼 크기를 인수로 받지 않아 입력 길이를 제한할 수 없는** 함수이다. 따라서 긴 입력이 들어오면 버퍼 경계를 넘어 기록할 수 있다. 선지의 '취약하지 않은 함수'는 '취약한 함수'로 고쳐야 한다.

③ O: 포맷 스트링은 출력할 데이터의 해석·표시 방식을 지정하는 문자열이다. 외부 입력을 단순한 데이터로 처리하지 않고 **포맷 문자열 자체로 해석**하면 공격자가 넣은 서식 지정자에 따라 의도하지 않은 메모리 접근 등이 발생할 수 있다. 예를 들어 printf(input)처럼 입력을 포맷으로 넘기는 대신 printf("%s", input)처럼 형식을 고정해야 한다. 선지의 '데이터 형태에 대한 불명확한 정의'는 이러한 형식 지정의 통제 부족을 가리킨다.

④ O: 입력 길이를 제한하지 않는 취약한 함수를 피하고, 버퍼 용량에 맞는 길이 검사를 수행하면 버퍼 경계를 넘는 기록을 예방할 수 있다. 최신 보안 패치와 메모리 보호 기능을 갖춘 운영체제를 사용하는 것도 공격 성공 가능성을 낮추는 방어 방법이다. 다만 **최신 운영체제만으로 프로그램의 경계 검사 오류가 없어지는 것은 아니므로** 코드 수준의 안전한 입력 처리를 함께 적용해야 한다.

NoteBase