[정답 및 해설] [소프트웨어 개발 보안 구축>SW개발 보안 설계]

SW개발 보안 설계 1. 기밀성, 가용성, 무결성

* SW 개발 보안의 세 가지 요소
- 기밀성(Confidentiality)
- 가용성(Availability)
- 무결성(Integrity)

SW개발 보안 설계 2. ① 무결성 ② 인증 ③ 가용성 ④ 부인 방지 ⑤ 기밀성

* SW 개발 보안 항목
- 기밀성(Confidentiality)
- 무결성(Integrity)
- 가용성(Availability)
- 인증(Authentication)
- 부인 방지(non-repudiation)

SW개발 보안 설계 3. ① (ㅁ) ② (ㄹ) ③ (ㄷ) ④ (ㄱ) ⑤ (ㄴ)

* SW개발 보안 용어
- 자산(Asset)
- 위협원(Threat agents)
- 위협(Threat)
- 취약점(Vulnerability)
- 위험(Risk)

SW개발 보안 설계 4. 행정 안전부, 한국인터넷진흥원(KISA), 발주기관, 사업자, 감리법인

* 소프트웨어 개발 보안 관련 기관
- 행정 안전부
- 한국인터넷진흥원
- 발주기관
- 사업자
- 감리법인

SW개발 보안 설계 5. ① 행정 안전부 ② 한국 인터넷 진흥원 ③ 발주기관 ④ 사업자 ⑤ 감리법인

* 소프트웨어 개발 보안 관련 기관
- 행정 안전부
- 한국인터넷진흥원
- 발주기관
- 사업자
- 감리법인

SW개발 보안 설계 6. ① 유지보수 ② 설계 ③ 요구사항 분석 ④ 구현 ⑤ 테스트

* SecureSDLC: 각 단계별로 요구되는 보안활동을 수행함으로써 안전한 소프트웨어를 만들 수 있도록 함.
- 1단계: 요구사항 분석 단계 주요 보안활동
- 2단계: 설계 단계 주요 보안 활동
- 3단계: 구현 단계 주요 보안 활동
- 4단계: 테스트 단계 주요 보안 활동
- 5단계: 유지보수 단계 주요 보안 활동

[정답 및 해설] [소프트웨어 개발 보안 구축>SW개발 보안 설계]

SW개발 보안 설계 7. ① SW 개발 보안 ② SecureSDLC

- SW 개발 보안: SW개발 과정에서 보안업무를 수행하며 안전한 보안요소를 만족하는 소프트웨어를 개발·운영하기 위한 목적으로 수행하는 개발 방법이다.
- Secure SDLC: 각 단계별로 요구되는 보안활동을 수행함으로써 안전한 소프트웨어를 만들 수 있도록 한다.

SW개발 보안 설계 8. 개인정보 보호법, 정보통신망 이용촉진 및 정보보호 등에 관한 법률, 신용정보의 이용 및 보호에 관한 법률, 위치정보의 보호 및 이용 등에 관한 법률, 표준 개인정보보호 지침, 개인정보의 안전성 확보 조치 기준, 개인 정보 영향평가에 관한 고시

* 개인정보보호 관련 법규
- 개인정보 보호법
- 정보통신망 이용촉진 및 정보보호 등에 관한 법률
- 신용정보의 이용 및 보호에 관한 법률
- 위치정보의 보호 및 이용 등에 관한 법률
- 표준 개인정보보호 지침
- 개인정보의 안전성 확보 조치 기준
- 개인정보 영향평가에 관한 고시

SW개발 보안 설계 9. ① (ㄴ) ② (ㄹ) ③ (ㄷ) ④ (ㄱ)

* 특정IT 기술관련 규정
- RFID 프라이버시 보호 가이드라인
- 위치정보의 보호 및 이용 등에 관한 법률
- 위치정보의 관리적, 기술적 보호조치 가이드
- 바이오 정보 보호 가이드라인
- 뉴미디어 서비스 개인정보보호 가이드라인

SW개발 보안 설계 10. 보안등급

물리적, 기술적 접근 허용범위와 정보유출 시 예상되는 피해를 기준으로 보안등급을 결정할 수 있다.

SW개발 보안 설계 11. 시큐어 코딩

- 시큐어 코딩(Secure coding): 구현 단계에서 발생할 수 있는 보안 취약점들을 최소화하기 위한 코딩

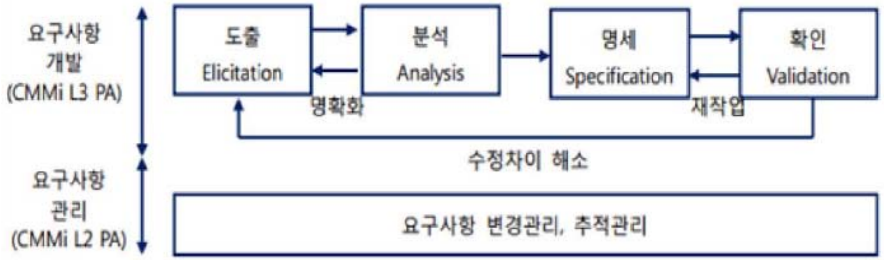
SW개발 보안 설계 12. ① 보안요구사항 개발 ② 보안요구사항 관리

보안요구 공학은 크게 보안요구사항 개발과 보안요구사항 관리로 나눠진다.

[정답 및 해설] [소프트웨어 개발 보안 구축>SW개발 보안 설계]

SW개발 보안 설계 13. ① 요구사항 도출② 요구사항 분석 ③ 요구사항 명세 ④ 요구사항 확인과 검증

* SW 개발보안 요구공학 프로세스



SW개발 보안 설계 14. 추적 매트릭스

서버 보안요구사항의 관리는 서버 보안요구사항 명세서와 함께 서버 보안요구사항 추적 매트릭스를 통해 산출물과 테스트 시나리오 등을 관리하도록 한다.

SW개발 보안 설계 15. ① (ㄴ) ② (ㄷ) ③ (ㄱ)

* SW개발보안 요구사항 명세와 설계 수행 순서
- 1단계: 정보에 대한 보안항목을 환경 분석에 따라 법률적으로 검토한다.
- 2단계: 보안항목의 요구사항을 수집한다.

SW개발 보안 설계 16. ① (ㄷ) ② (ㄱ) ③ (ㄴ)

- 보안취약점 환경 구축: 취약점 진단을 수행하기 위해 필요한 서버, 스토리지 및 네트워크 환경을 구성한다.

SW개발 보안 설계 17. ① (ㄴ),(ㄷ),(ㅁ) ② (ㄱ), (ㄹ)

* 보안성이 강화된 응용프로그램 구현을 위한 환경 구축
- 보안취약점 환경구축 기능
- 취약점 모니터링 환경 기능

SW개발 보안 설계 18. (ㄱ)→(ㄴ)→(ㄹ)→(ㄷ)

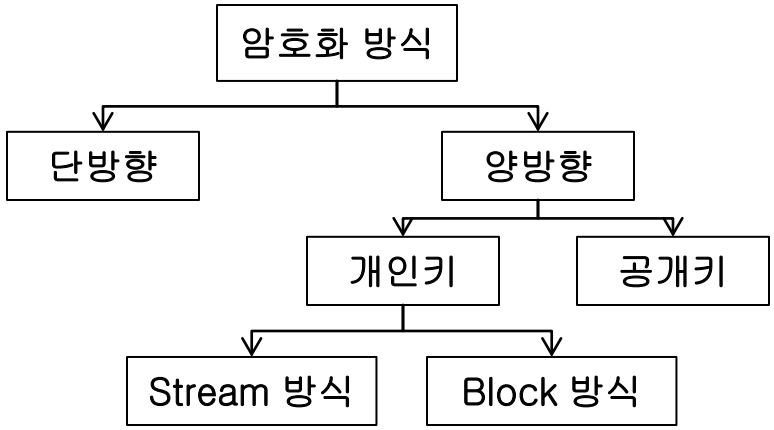
* 보안성이 강화된 응용프로그램 구현을 위한 일정 계획 수립
- 1단계: 일정 계획을 위한 보안 전문가를 선정
- 2단계: 일정 계획을 위한 보안 범주를 식별
- 3단계: 보안요구사항을 분석
- 4단계: 보안성이 강화되었는지 취약점 점검과 함께 독립적인 테스트 계획을 기획

[정답 및 해설] [소프트웨어 개발 보안 구축>SW개발 보안 구현]

SW개발 보안 구현 1. ① 개인키 ② 공개키

* 양방향 암호 알고리즘
- 개인키 암호화(Private Key Encryption)
- 공개키 암호화(Public Key Encryption)

SW개발 보안 구현 2. SHA-256 등 SHA 시리즈, MD5, SNEFRU, N-NASH



SW개발 보안 구현 3. ① DES, AES, SEED, ARIA ② LFSR, RC4

* 개인키 암호화 기법
- 블록 암호화 방식: DES, AES, SEED, ARIA
- 스트림 암호화 방식: LFSR, RC4

SW개발 보안 구현 4. DES(Data Encryption Standard)

- DES(Data Encryption Standard): 56비트의 암호/복호키를 이용하여 64비트의 평문(블록)을 암호화, 복호화 하는 방식

SW개발 보안 구현 5. ① AES(Advanced Encryption Standard) ② 128

- AES(Advanced Encryption Standard)
: DES의 한계를 느낀 미국 표준 기술 연구소(NIST)에서 2001년에 발표한 개인키 암호화 알고리즘

SW개발 보안 구현 6. 입력 데이터 검증 및 표현, 보안 기능, 시간 및 상태, 에러 처리, 코드 오류, 캡슐화, API 오용

* SW 보안약점 항목 (시큐어 코딩 점검 내용)
- 입력 데이터 검증 및 표현
- 보안 기능
- 시간 및 상태
- 에러 처리
- 코드 오류
- 캡슐화
- API 오용

[정답 및 해설] [소프트웨어 개발 보안 구축>SW개발 보안 구현]

SW개발 보안 구현 7. ① SHA-256 ② RSA ③ ARIA ④ SEED

* 암호화 알고리즘 종류
- DES(Data Encryption Standard)
- AES(Advanced Encryption Standard)
- SEED
- ARIA (아리아)
- RSA
- SHA-256 암호 알고리즘

SW개발 보안 구현 8. ① 시간 및 상태 ② 캡슐화 ③ 입력 데이터 검증 및 표현 ④ 보안 기능 ⑤ 코드 오류 ⑥ API 오용 ⑦ 에러 처리

* SW 보안약점 항목 (시큐어 코딩 점검 내용)
- 입력 데이터 검증 및 표현
- 보안 기능
- 시간 및 상태
- 에러 처리
- 코드 오류
- 캡슐화
- API 오용

SW개발 보안 구현 9. SQL 주입, 경로 조작 및 자원 삽입, 크로스사이트 스크립팅, 운영체제 명령어 삽입, 위험한 형식 파일 업로드, 신뢰되지 않는 URL 주소로 자동 접속 연결 등

위 나열된 보안약점 외 신뢰성이 낮은 URL주소로 자동으로 접속되는 LDAP 삽입, 연결 크로스사이트 요청 위조, XQuery 삽입, XPath 삽입, HTTP 응답분할 등이 있다.

SW개발 보안 구현 10. SQL 주입(SQL injection)

- SQL 주입(SQL injection): 입력란에 SQL을 삽입하여 무단으로 DB를 조회하거나 조작하는 보안 약점

SW개발 보안 구현 11. 크로스사이트 스크립팅(XSS)

- 크로스사이트스크립팅(XSS): 웹페이지에 악의적인 스크립트를 삽입하여 방문자들의 정보를 탈취하거나, 비정상적인 기능 수행을 유발하는 보안 약점

SW개발 보안 구현 12. 화이트리스트

- 화이트리스트: 블랙 리스트와 반대 의미를 가진 허용할 목록

SW개발 보안 구현 13. (ㄱ)

(ㄴ) SQL 주입의 해결 방안이다.
(ㄷ) 크로스사이트스크립팅의 해결 방안이다.

[정답 및 해설] [소프트웨어 개발 보안 구축>SW개발 보안 구현]

SW개발 보안 구현 14. TOCTOU(경쟁조건: 검사시점과 사용시점)

- 경쟁조건: 검사시점과 사용지점(TOCTOU): 검사 시점과 사용 시점을 고려하지 않고 코딩하는 경우 발생하는 보안약점

SW개발 보안 구현 15. return

안전하지 않은 코드에서 무한 재귀가 발생한다. 재귀 호출을 조건문(if~else) 블럭 안에서만 수행하도록 하면 무한 재귀를 해결할 수 있다.

SW개발 보안 구현 16. ① (ㄹ)-(b) ② (ㄱ) - (d) ③ (ㄷ)-(c) ④ (ㅂ)-(f) ⑤ (ㄴ)-(e) ⑥ (ㅁ) - (a)

* 보안 기능에 관련한 보안 약점
- 적절한 인증 없는 중요기능 허용
- 부적절한 인가
- 중요한 자원에 대한 잘못된 권한 설정
- 취약한 암호화 알고리즘 사용
- 사용자 중요정보 평문 저장 및 전송
- 하드코드된 비밀번호

SW개발 보안 구현 17. 예외처리

- 예외처리: 일반적으로 프로그램에서 특정한 문제가 일어났을 때 처리를 중단하고 오류 메시지를 띄우는 등 예외 처리를 하도록 흐름을 바꾸는 것

SW개발 보안 구현 18. (ㄹ)

```
#include <stdio.h>
int main(void) {
    int a;
    printf("변수 a의 값은 %d입니다.", a);
    return 0;
}
```

변수 a를 선언하면 기존 메모리에 들어있는 값을 그대로 사용하게 되어 의도하지 않은 결과를 출력하거나 에러가 발생하게 된다. 변수 사용 전 int a = 0; 과 같이 올바른 초기 값을 할당하여 초기화하도록 한다.

SW개발 보안 구현 19. 자원을 획득하여 사용한 다음에는 반드시 자원을 해제하여 반환한다.

- 부적절한 자원 해제 해결 방법
: 자원을 획득하여 사용한 다음 자원 해제 및 반환 코드를 작성하여 방지한다.

[정답 및 해설] [소프트웨어 개발 보안 구축>SW개발 보안 구현]

SW개발 보안 구현 20. ① (ㄱ) ② (ㄴ) ③ (ㄷ) ④ (ㅁ) ⑤ (ㄹ)

* 캡슐화에 관련한 보안 약점
- 잘못된 세션에 의한 정보 노출
- 제거되지 않고 남은 디버그 코드
- 시스템 데이터 정보 노출
- Public 메소드부터 반환된 Private 배열
- Private 배열에 Public 데이터 할당

SW개발 보안 구현 21. ① DNS ② API

* API오용 과 관련한 보안 약점
- DNS lookup에 의존한 보안결정
- 취약한 API 사용

SW개발 보안 구현 22. 영향도, 긴급도

* 보안결함 등급 = 보안사고 발생 시 복구 우선순위 = 영향도 x 긴급도 = 잠재적 손실의 영향 x 해결 시간의 중요성

SW개발 보안 구현 23. ① 정적 분석 ② 동적 분석

* SW개발보안 테스트 종류
- 정적 분석 - 동적 분석

SW개발 보안 구현 24. (ㄴ), (ㄹ)

- 정적 분석: SW를 실행하지 않고, 소스 코드 수준으로 보안 약점을 분석한다.

SW개발 보안 구현 25. (ㄹ)

* SW개발보안 테스트의 일반 원칙
- SW보안 테스트 계획서에 따른 실시
- SW보안 테스트의 적정성 판단
- SW보안 결과의 신뢰성 확인
- SW보안결함 발견 시 피드백 결과 확인

SW개발 보안 구현 26. SW보안결함

- SW보안결함: 소프트웨어 제품의 보안품질이 정의된 특성과 일치하지 않는 모든 행위

SW개발 보안 구현 27. ① (ㄴ) ② (ㄷ) ③ (ㄱ)

* SW보안결함의 종류
- SW보안 결함 - 발견된 보안 결함
- 잠재적 보안 결함 - SW의 특이한 고장

[정답 및 해설] [소프트웨어 개발 보안 구축>SW개발 보안 구현]

SW개발 보안 구현 28. (ㄱ)→(ㄹ)→(ㄴ)→(ㅂ)→(ㄷ)→(ㅅ)→(ㅁ)→(ㅇ)

* SW개발보안 테스트와 결함관리 수행 순서
1. 구현된 응용프로그램의 결함 여부를 테스트한다.
   1) 테스트(시험) 단계를 수행한다.
   2) 취약점 테스트를 실시할 주기와 대상을 선정한다.
   3) SW 취약점 테스트의 변경요청 제기 및 테스트 일정을 수립한다.
   4) SW 취약점 테스트를 수행한다.
2. 테스트 결과에 따라 발견된 결함을 관리한다.
   1) SW 취약점 테스트 보고서를 발행한다.
   2) SW 취약점의 지적사항을 해결한다.
   3) SW 보안결함 해결 사항을 보고한다.
   4) 점검이 완료된 소스코드를 관리한다.
   5) 전반적인 보안 상황을 보고 한다.