9장 소프트웨어 개발 보안 구축

109 Secure SDLC

- 보안상 안전한 소프트웨어를 개발하기 위해 SDLC에 보안강화를 위한 프로세스를 포함하 것

대표적 방법론 CLASP / SDL / Seven Touchpoints

단계별 보안 활동

요구사항 분석 단계 -> 설계 단계 -> 구현 단계 -> 테스트 단계 -> 유지보수단계

소프트웨어 개발 보안 요소 – 기밀성 / 무결성 / 가용성 / 인증 / 부인 방지

시큐어 코딩 – 구현 단계에서 발생할 수 있는 보안 취약점들을 최소화하기 위해 보안 요소들을 고려하며 코딩하는 것

110 입력 데이터 검증 및 표현

- 입력 데이터로 인해 발생하는 문제들을 예방하기 위해 구현 단계에서 검증해야 하는 보안 점검 항목들

입력 데이터 검증 및 표현의 보안 약점

SQL 삽입 / 경로 조작 및 자원 삽입 / 크로스사이트스크립팅 / 운영체제 명령어 삽입 / 위험한 형식 파일 업로드 / 신뢰되지 않는 URL 주소로 자동접속 연결 / 메모리 버퍼 오버플로

111 보안기능

- 소프트웨어 개발의 구현 단계에서 코딩하는 기능인 인증, 접근제어, 기밀성, 암호화 등을 올바르게 구현하기 위한 보안 점검 항목들

보안 기능의 보안 약점

적절한 인증없이 중요 기능 허용 / 부적절한 인가 / 중요한 자원에 대한 잘못된 권한 설정 / 취약한 암호화 알고리즘 사용 / 중요정보 평문 저장 및 전송 / 하드코드 된 암호화 키

112 코드 오류

- 개발자들이 코딩 중 실수하기 쉬운 형 변환, 자원 반환 등의 오류를 예방하기 위한 보안 점검 항목들

코드 오류의 보안 약점

널 포인터 역참조 / 부적절한 자원 해제 / 해제된 자원 사용 / 초기화되지 않은 변수 사용

스택 가드 – 주소가 저장되는 스택에서 발생하는 보안 약점을 막는 기술

113 캡슐화

- 정보 은닉이 필요한 중요한 데이터와 기능을 불완전하게 캡슐화하거나 잘못 사용함으로써 발생할 수 있는 문제를 예방하기 위한 보안 점검 항목들

캡슐화의 보안 약점

잘못된 세션에 의한 정보 노출 / 제거되지 않고 남은 디보그 코드 / 시스템 데이터 정보 노출 / Public 메소드로부터 반환된private 배열 / Private 배열에 public 데이터 할당

접근 제어자 – 외부로부터의 접근을 제한하기 위해 사용되는 예약어