정보처리기사 필기

소프트웨어 개발 10 인터페이스 구현 ②

양문자 선생님

출처 : ncs 학습모듈(NCS능력단위 인터페이스구현)

차례

1 데이터 입출력 구현

2 통합 구현

3 제품소프트웨어 패키징

4 애플리케이션 테스트 관리

5 인터페이스 구현

1) 인터페이스 설계 확인

2) 인터페이스 기능 구현

3) 인터페이스 구현 검증

인터페이스 기능 구현

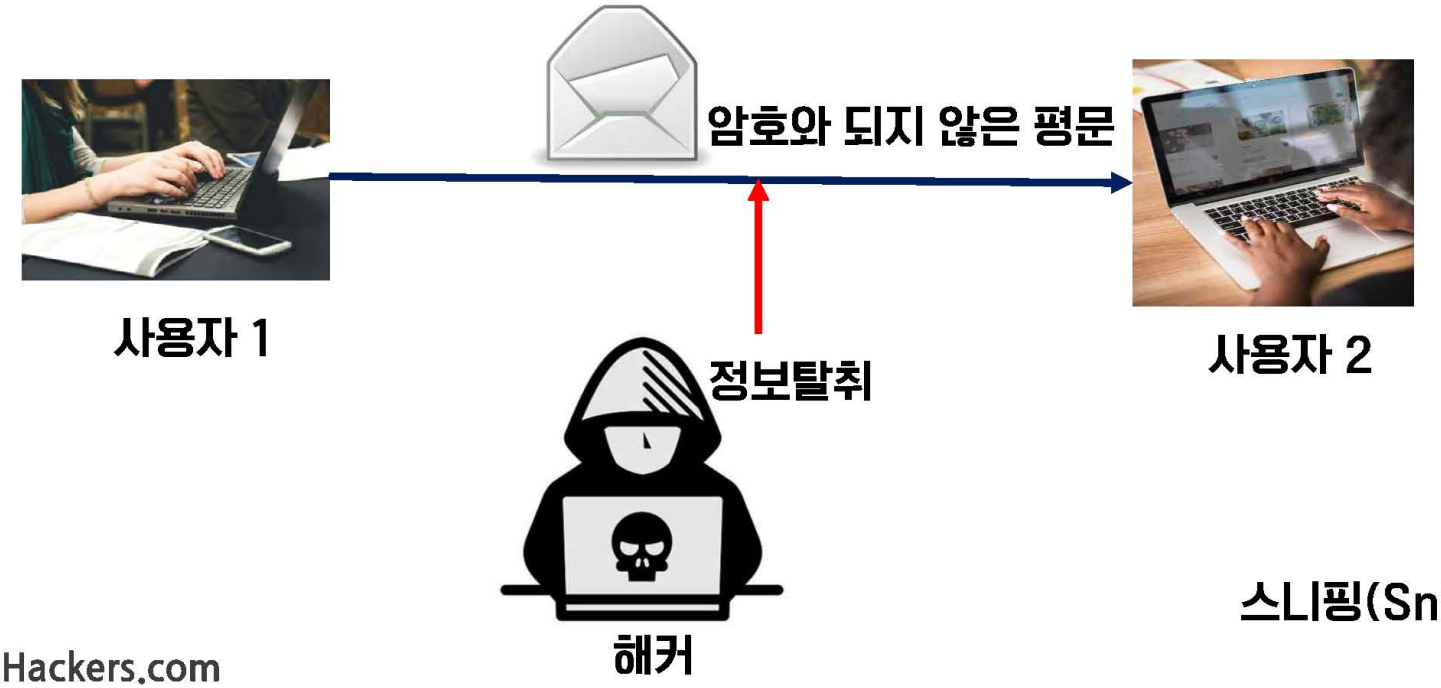
인터페이스 보안

- 구현된 인터페이스의 주요 보안 취약점

  인터페이스는 시스템 모듈 간 통신 및 정보 교환을 지원하므로 데이터 변조·탈취 및 인터페이스 모듈 자체의 보안 취약점이 있을 수 있다.

데이터 통신 시 데이터 탈취 위협

- 데이터 통신 내역을 중간에서 감청하여 기밀성을 훼손할 수 있는 기법(스니핑(Sniffing))
- 도청할 수 있도록 중간에 설치되는 도구를 스니퍼(Sniffer)라고 한다.
- 주로 패킷 분석기 같은 툴을 통해서 진행된다.



스니핑(Sniffing)

2. 시큐어 코딩 가이드
SW 보안 취약점, 약점 및 대응 방안이 구체적으로 서술

구분	내용
입력 데이터 검증 및 표현	소스코드 취약점 점검
API 이용	시스템 접근 API 오용
보안 특성	인증, 접근 제어, 기밀성, 암호화, 권한 관리, 취약한 알고리즘, 부적절 인가로 인한 취약점
시간 및 상태	프로세스 동시 수행 시, 잘못된 권한위임 가능성
에러 처리	에러 처리가 부적절하거나 에러에 정보가 과도하게 많이 포함된 경우
코드 품질	복잡한 소스코드가 가독성과 유지 보수성을 저하함.
캡슐화	중요 데이터의 불충분한 캡슐화로 악의적 접근 가능

3. 데이터베이스 암호화

데이터베이스의 기밀성을 유지하기 위해 중요 민감 데이터는 다양한 암호 알고리즘을 활용하여 암호화한다.

(1) 데이터베이스 암호화 알고리즘

구분	내용
대칭 키 암호 알고리즘	ARIA 128/192/256, SEED
해시 알고리즘	SHA -256/384/512, HAS-160
비대칭 키 알고리즘 (공개키 알고리즘)	RSA, ECDSA

(2) 데이터베이스 암호화 기법

구분	개념
API 방식	APP 레벨에서 암호 모듈(API)을 적용하는 APP 수정 방식
Filter(Plug-in) 방식	DB 레벨의 확장성 프러시저 기능을 이용, DBMS에 plug-in 모듈로 동작하는 방식
Hybrid 방식	API 방식과 Filter 방식을 결합

- 인터페이스 보안 기능 적용하기

1] 인터페이스의 보안 취약점을 분석한다.

인터페이스 구현이 어떻게 되어 있는지를 분석하고 각 구간에 어떤 보안 취약점이 있는지를 다양한 관점에서 분석한다.

2] 분석된 보안 취약점을 근거로 인터페이스 보안 기능을 적용한다.

1. 네트워크 구간에 보안 기능을 적용한다.

   - 인터페이스 송수신 간 중간자에 의한 데이터 탈취 위변조를 막기 위해서는 네트워크 트래픽에 대한 암호화가 필요하다.

   - 네트워크 구간 암호화를 위해서는 인터페이스 아키텍처에 따라 다양한 방식으로 보안 기능을 적용한다.

네트워크 구간 보안 기능 적용 시 고려 사항 예시

단계	고려사항	보안 기능 적용
Transport Layer Network 보안	상대방 인증을 적용	IPSec AH(Authentication Header) 적용, IKE(Internet Key Exchange) 프로토콜 적용
	데이터 기밀성 보장 필요	IPSec ESP(Encapsulation Security Payload) 적용
	End-to-End 보안 적용	IPSec Transport mode 적용
Application Layer Network 보안	서버만 공개 키 인증서를 가지고 통신(위협 분산)	SSL(Secure Soket Layer)의 서버 인증 모드 운영
	연결 단위 외 메시지 단위로도 인증 및 암호화 필요	S-HTTP 적용하여 메시지 암호화(상호 인증 필요, 성능 일부 저하됨)

문제풀이

- **인터페이스 보안을 위해 네트워크 영역에 적용될 수 있는 것으로 거리가 먼 것은?**

① IPSec　② SSL
③ SMTP　④ S-HTTP

(2020년 4회 정보처리기사 필기 기출문제 소프트웨어 개발)

- **인터페이스 보안을 위해 네트워크 영역에 적용될 수 있는 솔루션과 거리가 먼 것은?**

① IPSec　② SSL　③ SMTP　④ S-HTTP

(2020년 3회 정보처리기사 필기 기출문제 소프트웨어 개발)

- **인터페이스 보안을 위해 네트워크 영역에 적용될 수 있는 솔루션과 거리가 먼 것은?**

① IPSec　② SMTP
③ SSL　④ S-HTTP

(2020년 1, 2회 정보처리기사 필기 기출문제 소프트웨어 개발)

2. 애플리케이션에 보안 기능을 적용한다.

- 애플리케이션 구현 코드상에 보안 취약점을 보완하는 방향으로 애플리케이션 보안 기능을 적용한다. 주로 시큐어 코딩 가이드를 참조하여 보안 기능을 적용한다.

- 비인가자 접근 권한 관리, 악의적 코드 삽입 금지, 악의적 시도 시 에러 처리

3. 데이터베이스에 보안 기능을 적용한다.

- 데이터베이스의 접근 권한 및 데이터베이스 동작 객체(sql, 프러시저, 트리거 등)의 보안 취약점을 보완하기 위해 보안 기능을 적용한다.

- 민감 데이터의 경우에는 데이터 자체의 보안 방안(암호화, 익명화 등)도 고려한다.

인터페이스 구현 검증

인터페이스 구현 검증도구, 감시 도구

1. 인터페이스 구현 검증도구

- 인터페이스 구현을 검증하기 위해서는 인터페이스 단위 기능 및 시나리오에 기반한 통합 테스트가 필요하다.
- 테스트 자동화 도구를 이용하여 단위 및 통합 테스트의 효율성을 높일 수 있다.

소프트웨어 개발

- 인터페이스 구현 검증도구

도구	설명
xUnit	java(Junit), C++(Cppunit), .Net(Nunit) 등 다양한 언어를 지원하는 단위 테스트 프레임워크
STAF	서비스 호출, 컴포넌트 재사용 등 다양한 환경을 지원하는 테스트 프레임워크
FitNesse	웹 기반 테스트 케이스 설계/실행/결과 확인 등을 지원하는 테스트 프레임워크
NTAF	Naver 테스트 자동화 프레임워크이며, STAF와 FitNesse를 통합
Selenium	다양한 브라우저 지원 및 개발언어를 지원하는 웹 애플리케이션 테스트 프레임워크
watir	Ruby 기반 웹 애플리케이션 테스트 프레임워크

문제풀이

- **인터페이스 구현 검증 도구가 아닌 것은?**

① ESB ② xUnit
③ STAF ④ NTAF

(2020년 4회 정보처리기사 필기 기출문제 소프트웨어 개발)

- **인터페이스 구현 검증도구 중 아래에서 설명하는 것은?**

> ● 서비스 호출, 컴포넌트 재사용 등 다양한 환경을 지원하는 테스트 프레임워크
> ● 각 테스트 대상 분산 환경에 데몬을 사용하여 테스트 대상 프로그램을 통해 테스트를 수행하고, 통합하여 자동화하는 검증 도구

① xUnit ② STAF
③ FitNesse ④ RubyNode

(2020년 1, 2회 정보처리기사 필기 기출문제 소프트웨어 개발)

2. 인터페이스 감시 도구

- 인터페이스의 동작이 잘 진행되는지 확인하기 위해서는 애플리케이션 모니터링 툴 (APM:Application Performance Management)을 사용하여 동작 상태를 감시할 수 있다.
- 상용 제품 및 오픈소스를 이용한 애플리케이션 모니터링 툴이 있다.
- 데이터베이스, 웹 애플리케이션의 트랜잭션과 변수값, 호출 함수, 로그 및 시스템 부하 등 종합적인 정보를 조회하고 분석할 수 있다.

인터페이스 구현 검증하기

1] 인터페이스 명세서를 활용한 방법

1. 구현된 인터페이스 명세서를 참조하여 구현 검증에 필요한 감시 및 도구의 요건 분석을 한다.

기능 구현 정의	검증도구 요건	감시 도구 요건
1. 송신 측에서 인터페이스 대상 선택 전송	- 입력한 대상과 생성된 인터페이스 객체의 정보가 일치 하는 지 확인	- 데이터베이스 SQL 모니터링 - 조회 Transaction 모니터링 - JSON 생성 객체 모니터링
2. 인터페이스 객체 전송	- 암호화된 통신으로 올바른 수신 측에 전달되었는지 확인 - 전달된 정보가 수신된 정보와 일치하는지 확인 - 파싱된 정보가 송신된 정보와 일치하는지 확인	- 통신 암호화 모니터링 - 패킷정보 모니터링 - 연결된 Transaction 변수 모니터링
3. 수신 후 수신측 트랜잭션과 결과 반환	- 수신된 데이터와 연관 있는 이후 트랜잭션의 기댓값과 일치 여부	- 객체 입력, 출력값 모니터링 - 객체 동작 성공, 실패 여부

2. 구현된 인터페이스 명세서를 참조하여 구현 검증에 필요한 감시 및 도구를 준비한다.

- 구현 검증 및 감시에 필요한 도구의 요건을 확인 후 시장 조사 및 솔루션 조사를 통해서 적절한 감시 및 검증에 필요한 도구를 선택하여 구매할 수 있다.
- 최근에는 오픈소스 감시 도구도 많이 활용되고 있으므로 기능을 분석하여 도입을 검토한다.

2] 외부 시스템과의 연계 모듈 상태를 활용한 방법

1. 외부 시스템과 연계 모듈의 동작 상태를 확인하다.

- 인터페이스 구현 검증을 위하여 외부 시스템(송신 또는 수신)과 연계 모듈(수신 또는 송신)의 동작 상태를 인터페이스 구현 검증도구를 통해서 진행한다.
- 최초 입력값과 입력값에 의해 선택되는 데이터와 생성되는 객체의 데이터 등 전반적인 인터페이스 동작 프로세스상에서 예상되는 결과와 검증값을 비교한다.
- 각 단계별 에러 처리도 적절하게 구현되어 있는지 검증도구를 통해 확인한다.

소프트웨어 개발

- 인터페이스 검증도구를 통한 인터페이스 검증 시나리오(발령 시나리오)

인터페이스 기능	시나리오	예상값	검증도구 확인 실젯값
1. 최초 데이터 입력	과장 10명 입사 발령 확정	10명의 사번이 확정 FLAG를 가지고 저장	예상값과 동일
2. DB에서 조회	입사자 과장 10명 정보 조회	10명의 인사 발령 정보, 기본 정보가 Rowset 형태로 선택	예상값과 동일
3. 송신 객체 생성	전송 버튼 클릭	DB에서 조회된 동일 정보가 확정된 10명에 대해서 JSON 형태로 생성	예상값과 동일
4. 송신 객체 전송		10명 과장 입사 발령 확정 내용이 Submit flag를 가지고 송신 이력 적재	예상값과 동일
5. 수신 및 파싱		수신된 객체를 파싱한 결과는 송신한 결과와동일	예상값과 동일
6. 데이터 트랜잭션	수신 회사 인사 정보로 입력	파싱된 결과는 수신 측의 업무 정의에 따라 트랜잭션 진행(인사 기본 정보, 발령 정보에 입력)	
7. 수신 결과 반환	전송 회사 입사자 정보 조회	- 수신 결과가 송신 측 반환(TRUE) - 반환 이력 테이블에 해당 내역 입력	

2. 외부 시스템과 연계 모듈의 동작 상태를 감시(Monitoring)한다.

- 외부 모듈이 서비스를 제공하는 동안 정상적으로 동작하는지 감시 도구를 통해 확인할 수 있다.
- 인터페이스 동작 여부, 에러 발생 여부 등 감시 도구에서 제공해 주는 리포트를 활용한다.

인터페이스 오류 처리 확인 및 보고서 작성

1. 인터페이스 오류 처리 방법

(1) 사용자 화면에서 오류를 인지하게 구현하는 방법
- 가장 직관적으로 오류를 인지 할 수 있어 가장 많이 쓰이는 방법
- 인터페이스 오류가 발생하였을 경우 알람 형태로 화면에 표시되며, 주로 즉시적으로 데이터가 인터페이스되는 경우에 사용된다.

(2) 인터페이스 오류 로그 생성하는 방법
- 시스템 운영 로그에 인터페이스 오류 시 관련 에러 로그가 생성되도록 할 수 있다.
- 인터페이스 오류의 자세한 내역을 알기 위해 사용되며, 시스템 관리자나 운영자가 오류 로그를 확인할 수 있다.

(3) 인터페이스 관련 테이블에 오류 사항 기록

- 테이블을 통한 인터페이스 기능을 구현할 경우나 인터페이스 트랜잭션 기록을 별도로 보관하는 경우 테이블에 오류 사항을 기록할 수 있다.
- 이력을 직관적으로 보기 쉬워 운영자가 관리하기 용이한 장점이 있다.

송신일시	변경	발행번호	사번	발행내용	.....	처리일시	처리상태	오류코드	오류내용
21.1.27	입력	2021-001	21-001, 21-002	신규채용		21.1.27	실패	E-003	수신 데이터베이스 연결실패

소프트웨어 개발

2. 인터페이스 오류 처리 보고서

- 인터페이스에서 오류가 발생 시 관련 사항을 조직에서 정의된 보고 라인으로 인터페이스 오류 처리 보고서를 작성하여 즉각적으로 보고하여야 한다.
- 인터페이스 오류 처리 보고서 형식 정형화된 형식은 없으며 조직 및 상황에 맞는 보고서를 작성하여 활용한다.

장애(발생/진행/완료)보고서

2018.7.27 담당: ... 대리

장애처리 (발생/진행/완료)보고서		보고서 번호	
장애 발생 일시	2018년 7월 27일 09:00	장애환경	XX 시스템
장애 조치 일시	2018년 7월 27일 09:00 ~	종료여부	처리완료
장애 종료 일시	2018년 7월 27일 13:00	장애등급	2등급 ①영향도: B ②중요도: High
장애 내용 및 증상 - 인터페이스 오류			
장애 원인 - 시스템 결함 , 네트워크 장애			
조치 사항 - 조치경과 기록			
재발방지 계획 및 의견 - 향후 재발방지 방안			

장애 보고서(인터페이스 오류 보고서) 양식 예시

문제풀이

- 다음 중 인터페이스 오류 처리 방법으로 알맞지 않은 것은?

① 사용자 화면에서 오류를 인지하게 구현하는 방법
② 인터페이스 오류 시스템 로그를 별도로 작성하여 파일로 보관하는 방법
③ 인터페이스 오류 처리 도구를 사용하는 방법
④ 인터페이스 관련 테이블에 오류 사항 기록