Journal of Knowledge Information Technology and Systems

ISSN 1975-7700

http://www.kkits.or.kr

A Study of the Security assessment methodology for Android Mobile App

Kyoung Gon Kim[1], Huy Kang Kim[1], Eunjin Kim[*2]

[1]*Graduate School of Information Security, Korea University*
[2]*Department of International Industrial Information, Kyonggi University*

ABSTRACT

Apple iPhone was released on 2007, and Android 1.0 with alpha version was released in November of the same year. After seven years, in 2013, about 50 billion apps were downloaded from Android which tells that the mobile apps users were dramatically increased. Company also have developed and distributed mobile app to provide their service to users. As the number of mobile apps rapidlly increased, many mobile apps are still developed with vulnerability and distributed in markets due to the limitations of security assessment. Hackers usually repackag Apps and distribute the malicious Apps via Appstore or Googleplay in order to infect many devices. In this paper, we selected four mobile app security assessment methodologies. Local government, local private company, global security research institution and global consulting firm's methodologies were selected. Android-based mobile app security assessment methodology was developed for the security personnel to develop and operate in their organization. Mobile app security assessments methodology consists of 3 areas and 9 sub items and added menu assessments approach. We conducted the assessment using this methodology for the major domestic tele-communication company and found out that the assessment methodology developed for Android mobile app was efficiently assessed without missing any items compared to existing assessment methodologies.

KEYWORDS: Smart phone, Android App, Mobile Security threat, Security assessment, BYOD

ARTICLE INFO: Received 23 October 2014, Revised 13 February 2015, Accepted 13 February 2015.

[*]Corresponding author is with the Department of International Industrial Information, Kyonggi University

E-mail address: ejkim777@kgu.ac.kr

1. 서 론

모바일 폰의 역사는 2007년을 기점으로 전과 후의 패러다임이 급격하게 달라졌다. 2007년 6월 29일 아이폰 1.0이 출시되고, 2007년 11월 안드로이드 1.0 알파 버전이 공개되면서 본격적인 스마트폰의 역사가 시작되었다. 기존의 피처폰과 스마트폰의 가장 큰 차이점은 소위 '마켓(Market)'이라고 불리는 곳에서 원하는 애플리케이션(앱)을 언제 어디서든 다운로드 받아 모바일 폰에 설치할 수 있다는 점이었다.

애플리케이션 마켓의 활성화는 수많은 앱의 개발과 배포를 가능하게 하였다. 구글의 안드로이드 마켓인 구글 플레이는 2008년 10월 처음 오픈한 이후 2010년 7월 처음으로 10억 건 앱 등록을 넘어섰고, 2011년 3월에는 30억 건을 넘어섰다. 2013년 7월에는 안드로이드 구글 플레이에는 97만 5000개 이상의 앱이 등록되어 누적 다운로드 수는 500억 건을 돌파했다[1]

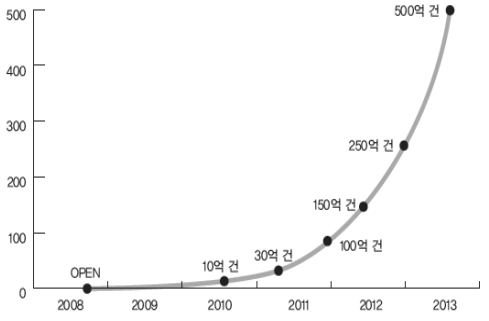

그림 1. 안드로이드 마켓 성장
Figure 1. Growth of Android market (accumulation of download counts)

기하급수적으로 모바일 앱이 증가함에 따라 모바일 앱의 보안 위협들도 증가하게 되었다. 트랜드마이크로는 2013년 악성앱과 고위험군 앱이 100만 개를 돌파하였다고 발표하였다[2]. 스마트 폰 보안 위협으로 플랫폼 공격, 네트워크 공격, 애플리케이션 공격, 단말기 공격으로 분류를 하고 있다[3].

표 1. 스마트폰 모바일 위협 유형
Table 1. Mobile Threat Types of Smart-Phone

분류	공격유형	공격방법
플랫폼 공격	바이러스/웜, 시스템 Unlock, 키보드 해킹	WiFi/블루투스/Web 이용 전파 PC 동기화 Jailbreak(iPhone) Rooting(Android) Security Off(WM) 플랫폼 취약점 Rootkit(백도어, 트로이 목마 등 해킹프로그램)
네트워크 공격	Wi-Fi 도청/변조 DoS 공격	Wi-Fi/블루투스 네트워크 공격
애플리케이션 공격	Malicious App. Fishing App.	Web 다운로드 PC 동기화
단말기 공격	도난 및 분실 Malicious App.	도난 및 분실 이동 저장매체 감염

운영체제의 안정성을 바탕으로 보안에 대한 위험이 상대적으로 낮았던 아이폰에 비해 보안 위협이 높은 안드로이드 진영이 시장 점유율을 급격히 높여감에 따라 모바일 보안 위협은 더욱 더 많은 사용자들을 보안 위험에 노출시키게 되었다[4].

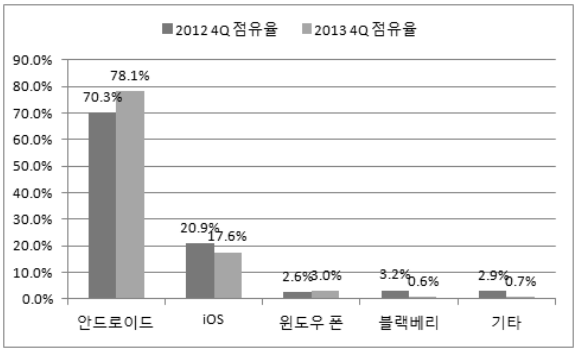

그림 2. 스마트폰 운영체제 시장 점유률, 4Q 2013
Figure 2. Smartphone Operating Systems Market Share, 4Q 2013

증가하고 있는 모바일 앱의 보안 위협에도 불구하고 스마트폰은 네트워크 접속의 편리성을 강점으로 하여 지속적으로 웹 시장의 지배력을 대체하는 매개체로 비중을 높여 가고 있다. 넷크래프트(Netcraft)의 2014년 웹 정기조사에서도 전 세계 웹 사이트(website)의 성장 속도가 뚜렷하게 둔화되고 있음을 알 수 있다[5]. 반면 스마트폰은 기업이 효과적, 효율적으로 고객에게 정보를 제공할 수 있도록 하여 개인의 영역을 넘어 비즈니스 영역까지 역할을 확대하고 있다. 모바일 보안은 개인 수준을 넘어 기업에서 중요한 보안 분야로 고려되어 가고 있다. 가트너에 따르면 2017년까지 기업의 1/4 정도가 기업용 앱 스토어를 가지게 될 것이라고 전망했다[6].

현재 기업에서 활용하고 있는 모바일 앱 보안진단 방식은 단편적으로 보안 위협을 예방하는데 한계가 있어 많은 모바일 앱들이 보안 취약점을 내포한 상태로 개발되고 마켓에 배포되고 있다. 안드로이드 운영체제는 아이폰 운영체제인 iOS에 비해 보안 위협이 높은 편이다. Sophos의 분석에 따르면, 안드로이드는 대중성과 벤더의 낮은 통제 수준으로 인해 아이폰에 비해 위협이 높다고 보고 있으며, 사용자는 안드로이드 앱 마켓에서 악성 앱이 제거되기 전에 약 26만 번 이상의 악성 앱을 다운로드 받는다고 조사하였다[7].

본 논문에서 안드로이드 모바일 앱을 개발하고 운영할 때 점검해야 하는 보안진단 방법론을 제시하고자 한다.

2. 기존 유사 방법론

2.1 유사 방법론 선정 방법

유사 방법론 조사에는 국내와 국외 방법론을 각각 선정하였으며, 공공 성격을 가지고 있는 방법론과 민간 성격을 가지고 있는 방법론을 선정하였다. 국내 방법론으로는 공공기관에는 정보보안 분야의 공신력 있는 기관인 안전행정부와 한국인터넷진흥원에서 발표한 방법론을 선정하였으며, 국내 민간의 경우 회사들 마다 고유한 방법론을 사용 중이나, 모바일 앱 개발이 매우 빈번하고 많이 배포되고 있는 국내 대기업 통신사에서 사용하는 방법론을 선정하였다. 해외의 경우 보안진단을 수행할 때 대표적으로 참고하는 공신력 있는 공개프로젝트조직인 OWASP에서 발표한 방법론을 선정하였으며, 해외 민간의 경우 Gartner, Forrester, Kennedy 등 유수의 리서치 기관으로부터 전 세계 보안업계 중 No.1으로 평가받는 Deloitte에서 사용하는 방법론을 선정하였다[8,9,10].

표 2. 안드로이드 앱 보안진단 방법론
Table 2. Security testing methodologies for Android App

구분	방법론 및 점검기준	세부내용
국내 국가기관 (KISA, 안전행정부)	모바일 전자정부서비스 앱 검증 신청기관을 위한 앱 소스코드 보안성 검증 기준	소스코드 보안약점 세부 점검항목 47개 항목, 기능 보안취약점 9개 영역
국내 민간기관 (대기업 통신사)	모바일 앱 보안 진단 항목	App 권한, 보호를 포함한 6개 영역
해외 연구기관 (OWASP)	OWASP 모바일 주요 10가지 위험 요소	서버통제, 데이터보호를 포함한 10개 위험 요소
해외 민간기관 (Deloitte)	Mobile Application Security Best Practice	인증, 권한관리를 포함한 9개 영역

2.2 국내 국가기관 방법론

한국인터넷진흥원(KISA, Korea Information Security Agency)과 안전행정부는 2011년 8월 모바

일 전자정부서비스 앱 검증 신청기관을 위한 앱 소스코드 보안성 검증 가이드라인을 최초 작성하고 발표하였다[11]. 이후 3번의 개정을 거쳐 2014년 2월 V4.0으로 개정되었다.

가이드라인은 행정기관 및 공공기관 등에서 개발한 모바일 대국민 전자정부서비스 앱에 대해, 검증기관(한국인터넷진흥원)에서 소스코드 보안약점 검증 시 참고하기 위한 절차 및 검증 기준을 소개하고 있다. 가이드라인에는 소스코드 보안약점(SW, Source Code Weakness) 세부 점검항목으로 47개 항목을 포함하고 있다.

표 3. 소스코드 보안 약점 체크리스트
Table 3. Source code security weaknesses checklists

검증 기준	No	세부 보안약점
입력 데이터 검증	1	SQL 삽입
	2	경로 조작 및 자원 삽입
	3	크로스사이트 스크립트
	4	운영체제 명령어 삽입
	5	위험한 형식 파일 업로드
	6	신뢰되지 않는 URL 주소로 자동접속 연결
	7	XQuery 삽입
	8	XPath 삽입
	9	LDAP 삽입
	10	크로스사이트 요청 위조
	11	HTTP 응답분할
	12	정수형 오버플로우
	13	보안기능 결정에 사용되는 부적절한 입력값
	14	메모리 버퍼 오버플로우
	15	포맷 스트링 삽입
보안 기능	1	적절한 인증 없는 중요기능 허용
	2	부적절한 인가
	3	중요한 자원에 대한 잘못된 권한 설정
	4	취약한 암호화 알고리즘 사용
	5	중요정보 평문저장
	6	중요정보 평문전송
	7	하드코드된 비밀번호
	8	충분하지 않은 키 길이 사용
	9	적절하지 않은 난수 값 사용
	10	하드코드된 암호화 키
	11	취약한 비밀번호 허용
	12	사용자 하드디스크에 저장되는 쿠키를 통한 정보 노출
	13	주석문안에 포함된 시스템 주요정보
	14	솔트 없이 일반향 해쉬 함수 사용
	15	무결성 검사없는 코드 다운로드
	16	반복된 인증시도 제한 기능 부재
시간 및 상태	1	경쟁조건: 검사 시점과 사용 시점(TOCTOU)
	2	종료되지 않는 반복문 또는 재귀함수
에러 처리	1	오류 메시지를 통한 정보 노출
	2	오류 상황 대응 부재
	3	부적절한 예외 처리
코드 오류	1	Null Pointer 역참조
	2	해제된 자원 사용
	3	부적절한 자원 해제
	4	초기화되지 않은 변수 사용
캡슐화	1	잘못된 세션에 의한 데이터 정보 노출
	2	제거되지 않고 남은 디버그 코드
	3	시스템 데이터 정보노출
	4	Public 메소드로부터 반환된 Private 배열
	5	Private 배열에 Public 데이터 할당
API 오용	1	DNS lookup에 의존한 보안결정
	2	취약한 API 사용

기능 보안취약점(FV, Function Vulnerability) 기준을 9개 영역으로 구분하였다.

표 4. 기능 보안취약점 체크리스트
Table 4. Function security weaknesses checklists

항목	세부 점검항목	
임의 기능	FV -1.1	명세되지 않은 기능 존재 여부
	FV -1.2	악성행위 기능 존재 여부 (불법 녹음, 임의 데이터 전송, 임의로 위치정보 수집, 전송 등)
최소 권한	FV -2.1	관리자 권한으로 동작하는 기능(권한 상승 포함) 존재 여부
	FV -2.2	인가되지 않은 API 사용(공통)
	FV -2.3	동일한 개인키로 서명된 다른 앱과 UID 공유 여부(Android)

	FV -2.4	기능사용 요청 권한과 기능사용 여부 적절성 여부(Android)
	FV -2.5	인텐트 권한의 올바른 설정 여부 (Android)
입력값 유효성	FV -3.1	외부 입력값의 유효성 (지정된 길이 초과, 악성코드 포함 등)검증 기능 존재 여부
중요 정보 관리	FV -4.1	주민등록번호, 여권번호, 면허번호, 외국인등록번호, 금융정보에 대한 암호화 저장 여부
	FV -4.2	주민등록번호, 여권번호, 면허번호, 외국인등록번호, 금융정보에 대한 암호화 전송 여부
	FV -4.3	비밀번호 및 바이오정보에 대한 일방향 암호화 저장 여부
	FV -4.4	개인위치정보의 안전한 저장을 위한 암호화 적용 여부
	FV -4.5	개인위치정보의 안전한 전송을 위한 암호화 적용 여부
	FV -4.6	사용자 인증 방법의 적절성 유무
	FV -4.7	비밀번호 조합규칙(영문, 숫자, 특수문자 등 조합 9자리 이상 등)
	FV -4.8	앱과 관련된 앱 서버의 기능, 중요정보 관리기능 등 점검
플랫폼 보안 모델	FV -5.1	루팅, 탈옥 등과 같은 플랫폼 변조 기능 존재 여부
	FV -5.2	플랫폼에서 제공하는 보안기능 사용의 적절성 여부
상용/ 공개용 모듈	FV -6.1	상용 또는 공개모듈 사용 목적 및 기능의 적절성 여부
	FV -6.2	해당 모듈에 대한 개발업체의 안전성 확인 방법 및 결과의 적절성 여부
공개 영역 취약점	FV -7.1	모바일 플랫폼(예, 안드로이드, iOS, 윈도우모바일 등) 등에 대해 알려진 (및 신규) 취약점 존재 여부
모바일 보안공 통기반 적용적 절성	FV -8.1	모바일 공통기반 구현의 적절성 여부
	FV -8.2	보안공통기반에 적용하지 않은 앱의 경우에는 공통기반에서 제공하는 기능에 부합되도록 보안이 제공되어야 한다.
기타	FV -9.1	신청제품의 서비스 특성에 따른 추가적인 보안취약점 존재 여부
	FV -9.2	모바일 앱 개발 및 배포 시 코드 난독화 도구 사용 여부

한국인터넷진흥원과 안전행정부에서 개발한 모바일 전자정부서비스 앱 검증 신청기관을 위한 앱소스코드 보안성 검증기준은 중앙부처, 지자체, 공공기관을 대상으로 개발한 것이며, 소스코드 및 실행파일을 제출하면 한국인터넷진흥원에서 기능 보안취약점과 소스코드 보안약점을 검토해 주는 방식이다.

위 점검항목은 각 항목에 대해 구성과 내용을 설명하고 있을 뿐, 각 항목별로 점검해야 하는 세부 방법과 판단 기준은 공개되어 있지 않다. 위 방법론은 민간에서 개발하는 앱을 점검하는 방법론으로 활용하기에는 제약사항이 있다.

2.3 국내 민간기관 방법론

국내 민간기관 중 모바일 앱을 많이 개발하는 A 통신사를 방법론 조사 대상으로 선정하였으며, 다음 <표 5>와 같이 6개의 영역으로 진단 항목을 구성하고 있다. 앱 스토어에 올라가기 전에 개발된 모든 앱은 아래 보안항목을 기준으로 점검한 이후에 배포하고 있다. 아래 항목은 안드로이드 바이너리(apk) 파일에 대해 동적 분석, 정적 분석 기반으로 점검하고 있다.

표 5. 국내 대기업 통신사에서 활용 중인 모바일 보안진단 항목
Table 5. Mobile app security assessment checklist of Korea Major A Telecommunication Company

진단 유형	진단 항목
App 권한	불필요한 App 권한 포함 여부
App 보호	앱 무결성 검증 기법 적용 여부
	앱 자체 보호기법 적용 여부
사용자 데이터 보호	보안 프로그램 적용 여부
민감한 정보 노출	코드 내 중요정보 포함 여부
	코드 내 저장시 암호화 여부
불안전한 데이터 저장	코드 외 중요정보 포함 여부
	코드 외 저장시 암호화 여부
불충분한 전송 계층	중요정보 전송 여부
	전송 시 암호화 여부

점검자는 앱을 구동하면서 취약점이 발견되면

다른 취약점을 확인하는 구조로 진단하고 있으며, 모든 메뉴에 대해 각 진단 항목을 모두 수행하지는 않고 있다.

2.4 해외 연구기관 방법론

2012년 4월, OWASP(공개 웹 애플리케이션 보안 프로젝트)에서 Advanced Mobile Application Code Review Techniques 보고서를 발표했다[12]. 보고서에는 모바일 보안 진단 테스트 영역으로 'Reading Stored Data', 'Capturing Request', 'Reversing the Application Package', 'Platform Specific Issues'로 구분하였다.

표 6. OWASP 모바일 상위 위험 10가지
Table 6. OWASP Mobile Top 10 Risk

Code	Risk
M1	취약한 서버 측 통제 (Weak Server Side Controls)
M2	안전하지 않은 데이터 저장 (Insecure Data Storage)
M3	충분하지 않은 전송 계층 보호 (Insufficient Transport Layer Protection)
M4	의도하지 않은 데이터 노출 (Unintended Data Leakage)
M5	취약한 인증과 권한 (Poor Authorization and Authentication)
M6	불안전한 암호화 (Broken Cryptography)
M7	클라이언트 측 인젝션 (Client Side Injection)
M8	신뢰할 수 없는 입력을 통한 보안 결정 (Security Decisions via Untrusted Inputs)
M9	안전하지 않은 세션 처리 (Improper Session Handling)
M10	바이너리 보호 미흡 (Lack of Binary Protections)

2013년 OWASP에서 모바일 환경에서 주의해야 할 열 가지 위험을 기술한 '2013년 OWASP Mobile Security Project' 초안을 발표했고, 2014년 현재 version 1.0으로 배포하고 있다. 다음 표는 OWASP에서 조사한 모바일 환경에서 주의해야 할 10가지 위험 요소이다[13].

OWASP에서 발표한 Mobile Top 10 Risk는 한국인터넷진흥원과 안전행정부에서 발간한 가이드라인에 비해 점검해야 하는 항목의 개수는 적으나, 항목별로 확인할 수 있는 방법은 보다 구체적으로 기술하고 있다.

OWASP에서 발표한 모바일 상위 위험 10가지 점검 방법에는 항목 별 취약점 유무에 대한 객관적인 판단기준이 없으며, 모바일 앱이 가지고 있는 메뉴에 대해 개별적으로 취약점 유무를 점검하는 방식은 아니다.

2.5 해외 민간기관 방법론

Gartner, Forrester, Kennedy 등 유수의 리서치 기관으로부터 전 세계 보안업계 중 No.1으로 평가받는 Deloitte에서 조사한 Mobile Application Security Best Practice에는 9개의 영역으로 모바일 앱 보안을 구분하고 있다.

표 7. 딜로이트 모바일 애플리케이션 보안 베스트 프랙티스
Table 7. Deloitte Mobile Application Security Best Practice

No	Mobile Application Security Best Practice 항목
1	Authentication
2	Authorization
3	Configuration Management
4	Sensitive Management
5	Session Management
6	Input Validation
7	Cryptography
8	Exception Management
9	Auditing and Logging

Deloitte Mobile Application Security Best Practice의 항목은 OWASP Mobile Top 10 Risk와 비슷한 항목으로 구성되어 있으나, 진단 방법에 대한 세부적인 내용은 기술되어 있지 않다.

2.6 국·내외 방법론 특징 및 장단점

앞서 설명한 국내 국가기관, 국내 민간기관, 해외 연구기관, 해외 민간기관에서 발표한 방법론의 특징 및 장단점은 다음과 같다.

표 8. 각 모바일 앱 보안진단 방법론의 장단점
Table 8. Pros and Cons for each mobile app security assessment methodology

구분	방법론 및 점검기준	특징 및 장단점
국내 국가기관 (KISA, 안전행정부)	모바일 전자정부서비스 앱 검증 신청기관을 위한 앱 소스코드 보안성 검증 기준	- 국가기관 및 공공기관 대상으로 작성 - 소스코드 보안 약점, 기능 보안 취약점으로 세분화 됨 - 민간에서 진단방법론으로 사용하기에는 세부 진단 방법 및 항목 판단 기준이 없음
국내 민간기관 (대기업 통신사)	모바일 앱 보안 진단 항목	- 세부 진단 방법 및 항목 판단 기준이 없음
해외 연구기관 (OWASP)	OWASP 모바일 주요 10가지 위험 요소	- 각 항목별 접근 방식은 기술되어 있음 - 메뉴별 접근 방식이 아니며, 항목 판단 기준이 없음
해외 민간기관 (Deloitte)	Mobile Application Security Best Practice	- 세부 진단 방법 및 항목 판단 기준이 없음

3. 제안 방법론

3.1 설계방법

본 논문에서는 국내 국가기관, 국내 민간기관, 해외 연구기관, 해외 민간기관에서 제시한 모바일 앱 보안진단 방법론의 단점을 보완하여 최적화된 범용 보안진단 방법론을 제안한다.

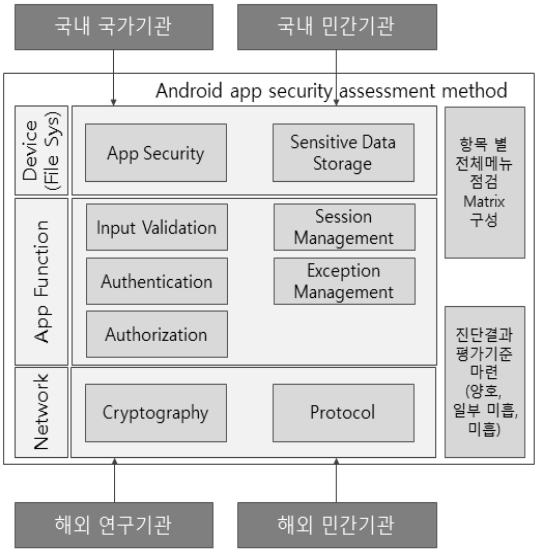

그림 3. 안드로이드 모바일 앱 보안진단 영역 설계 방법
Figure 3. Design approach of major categories for Android mobile app security assessment

제안하는 모바일 앱 보안진단 방법론 항목으로 Layer 관점에서 대분류 3개(Device, App Function, Network)요소와, 3개 요소에 대한 세부 9개 영역 (1)App Security, (2)Sensitive Data Storage, (3)Input Validation, (4)Authentication, (5)Authorization, (6)Session Management, (7)Exception Management, (8)Cryptography, (9)Protocol을 개발하였다. 각 진단 영역별로 앱 메뉴 별 진단 필요성 여부를 구체화 하고 진단 항목에 대한 객관적인 평가 기준을 설계하였다.

표 9. 제시한 모바일 앱 보안진단 세부 영역
Table 9. Detailed items for mobile app security assesment

항목	세부 진단 영역	메뉴별 점검 여부
1. App Security	1-1 앱 자체 보호 및 소스 난독화 여부	N
	1-2 앱 무결성 검증 여부	N
	1-3 루팅 체크 여부	N
2. Sensitive Data Storage	2-1 앱 구동 파일 내 중요정보 존재 및 암호화 여부	N
	2-2 코드 내 중요정보 존재 및 암호화 여부	N
	2-3 안전한 암호화 알고리즘 및 키 길이 사용 여부	N
3. Input Validation	3-1 특수문자 필터링 여부	Y
	3-2 악성파일 업로드 차단여부	Y
	3-3 파일다운로드 경로 검증여부	Y
	3-4 매개변수 조작 검증 여부	Y
4. Authentication	4-1 강화된 인증 방식 사용 여부	Y
	4-2 안전한 패스워드 사용 여부	Y
	4-3 인증 실패 통보 여부	Y
	4-4 중요기능 인증 강화 여부	Y
	4-5 자체서명 인증서 탐지 여부	Y
5. Authorization	5-1 앱 퍼미션 설정 여부	N
	5-2 타 사용자 도용 통제 여부	Y
	5-3 기능제한 우회 금지여부	Y
	5-4 불필요하거나 사용하지 않는 Activity 제거 여부	N
6. Session Management	6-1 세션 보호 여부	Y
7. Exception Management	7-1 안전한 에러처리 구현 여부	Y
8. Cryptography	8-1 중요정보 전송시 암호화 여부	Y
	8-2 전송 시 안전한 암호화 알고리즘 및 키 길이 사용 여부	Y
9. Protocol	9-1 안전한 프로토콜 사용 여부	Y

모바일 앱은 서비스 제공을 위해 여러 메뉴로 구성되어 있는 경우가 많다. 기존 점검 방법론들은 한 메뉴에서 취약점이 발견되면, 나머지 메뉴를 진단하지 않는 경향이 있다. 본 논문에서 설계한 방법론에는 모든 메뉴에 대해 전체적인 진단결과를 할 수 있도록 항목 별 점검 Matrix를 구성하였다.

기존 진단 방법들은 항목 위주로 설명 되어 있기 때문에 진단자가 보유한 수준에 따라 진단 결과가 상이할 수 있는 문제가 있다. 진단 결과에 대한 판단을 최대한 객관적으로 하기 위해 진단 결과에 대한 양호, 일부 미흡, 미흡에 대한 기준을 마련하였다. 다음은 본 논문에서 제시하는 세부 9개 진단 영역에 대해 개발한 모바일 앱 보안진단 요소이다.

3.2 세부항목 및 평가기준

3.2.1 앱 보호(App Security)

표 10. 앱 보호 세부 항목
Table 10. Detailed items for app security

번호	세부 진단 항목	진단항목 확인 방법 / 항목별 평가 기준
1-1	앱 자체 보호 및 코드 난독화 여부	❒ **확인 방법** 앱 자체 암호화 및 보호기능이 없어 공격자가 앱을 디컴파일 후 앱 소스코드에 대해 난독화가 되어 있지 않아 소스코드 획득 및 분석이 가능한지 확인 * 디컴파일 방법 1. 안드로이드폰에서 apk 추출 2. 7-zip을 이용해서 apk압축해체 3. dex2jar를 통해 classes.dex 파일 jar로 변환 4. jad를 이용해 jar파일 압축해제
		❒ **평가 기준** **양호**: java 파일 내 함수, 클래스 등이 이름이 변환되어 있거나, 소스코드 형태가 보이지 않은 형태로 되어 있어 분석이 난해한 경우 **일부 미흡**: 난독화가 적용되어 있으나 일부 내용이 미적용인 경우 **미흡**: 소스코드 난독화가 적용되지 않아 소스코드 분석이 쉬운 경우

1-2	앱 무결성 검증 여부 (코드 수정 후 리패키징)	❒ **확인 방법** 앱 무결성 검증이 미흡하여 리패키징된 앱이 구동될 수 있는지 여부 확인
		❒ **평가 기준** **양호**: 프로그램 변조 후, 실행 시 위변조 여부를 검증하여 App이 비정상 종료되는 경우 **일부 미흡**: App 위변조 여부를 소스 코드 단에서 구현하여 이를 위변조 가능하거나 단순 네트워크 패킷 변조로 우회가 가능한 경우 **미흡**: 프로그램 변조 후, 실행 시 App이 정상 실행이 되는 경우
1-3	루팅 체크여부	❒ **확인 방법** 앱을 구동할 때 기기가 루팅 되어 있는지 체크하는 코드가 있는지 확인 루팅 체크하는 방법: 아래 파일이 있는지 검사하는 것이 일반적 /system/bin/su /system/xbin/su /system/app/superuser.apk /data/data/com.noshufou.android.su
		❒ **평가 기준** **양호**: 루팅 된 스마트폰에서 앱을 실행하였을 때 루잍 여부를 탐지하는 경우 **미흡**: 루팅 된 스마트폰에서 앱을 실행하였을 때 루팅 여부를 탐지하지 못하는 경우

3.2.2 민감한 데이터 저장(Sensitive Data Storage)

표 11. 민감한 데이터 저장 세부 항목
Table 11. Detailed items for sensitive data storage

번호	세부 진단 항목	진단항목 확인 방법 / 항목별 평가 기준
2-1	앱 구동 파일 내 중요정보 존재 및 암호화	❒ **확인 방법** 앱을 구성하는 파일 내에 중요한 정보가 암호화되어 있지 않은 채 노출되는지 확인
	여부	❒ **평가 기준** **양호**: 단말기 리소스 파일 내 중요 정보가 없거나, 암호화되어 저장된 경우 **일부 미흡**: 단말기 리소스 파일 내 중요 정보가 인코딩(Base64등) 되어 저장된 경우 **미흡**: 단말기 리소스 파일 내 중요 정보가 평문으로 노출된 경우
2-2	코드 내 중요정보 존재 및 암호화 여부	❒ **확인 방법** 앱의 소스코드(Java 등) 내에 계정정보, 시스템접속정보, 개인정보 등 중요한 정보가 포함되어 있는지 확인
		❒ **평가 기준** **양호**: 소스 코드 내 중요 정보가 없거나, 암호화되어 저장된 경우 **일부 미흡**: 소스 코드 내 중요 정보가 인코딩(Base64등) 되어 저장된 경우 **미흡**: 소스 코드 내 중요 정보가 평문으로 노출된 경우
2-3	안전한 암호화 알고리즘 및 키 길이 사용 여부	❒ **확인 방법** 중요 데이터 암호화 시, 안전한 암호화 알고리즘 및 키 길이 사용 여부 확인
		❒ **평가 기준** 자체 암호화 알고리즘을 사용한 경우, 검증된 암호화 알고리즘인지를 확인함.(인터뷰를 통해서 확인하고 관련 알고리즘이 안전한 암호화 알고리즘인지를 반드시 체크해야 함.)

3.2.3 입력 값 검증(Input validation)

표 12. 입력 값 검증 세부 항목
Table 12. Detailed items for input validation

번호	세부 진단 항목	진단항목 확인 방법 / 항목별 평가 기준
3-1	특수문자 필터링 여부	❒ **확인 방법** SQL, XML 인젝션 공격문자 또는 XSS 공격문자와 같은 특수문자 입력을 필터링 하는지 확인
		❒ **평가 기준** **양호**: 입력된 문자열에 대해서 검증 절차가 있어 별다른 에러 없이 데이터 처리가 정상적으로 이루어지는 경우 **미흡**: 의도하지 않는 결과가 도출되거나 비정상 동작이 발생하는 경우

3-2	악성파일 업로드 차단여부	❐ **확인 방법** 악성 스크립트가 포함된 파일을 업로드 할 수 있는지 확인
		❐ **평가 기준** **양호**: 악성 스크립트가 포함된 위험한 확장자를 가진 파일을 업로드 하였을 때 확장자 검증절차가 존재하여 악성파일 업로드가 되지 않는 경우 **미흡**: 악성 스크립트가 포함된 위험한 확장자를 가진 파일을 업로드 하였을 때 확장자 검증절차가 없어 성공적으로 악성 파일이 업로드 되는 경우
3-3	파일다운로드 경로 검증여부	❐ **확인 방법** 파일다운로드 시 경로조작을 통해 다른 파일을 다운로드 받을 수 있는지 확인
		❐ **평가 기준** **양호**: 의도하지 않은 파일을 받으려고 할 때 다운로드 경로 및 파일명에 대한 필터링이 되어 있는 경우 **미흡**: 의도하지 않은 파일을 받으려고 할 때 다운로드 경로 및 파일명에 대한 필터링이 되어 있지 않아 임의의 시스템 및 애플리케이션 파일을 다운로드 받을 수 있는 경우
3-4	매개변수 조작 검증 여부	❐ **확인 방법** 매개변수 조작을 시도했을 때 별도의 에러를 발생시키지 않는지 여부 확인
		❐ **평가 기준** **양호**: 입력된 문자열에 대해서 검증절차가 있어 별다른 에러 없이 데이터 처리가 정상적으로 이루어지는 경우 **미흡**: 의도하지 않는 결과가 도출되거나 비정상 동작이 발생하는 경우

3.2.4 인증(Authentication)

표 13. 인증 세부 항목
Table 13. Detailed items for authentication

번호	세부 진단 항목	진단항목 확인 방법 / 항목별 평가 기준
4-1	강화된 인증 방식 사용	❐ **확인 방법** Device identifiers 값으로 단순 인증을 수행하는지 여부 확인.
	여부	❐ **평가 기준** **양호**: 개인식별을 위한 절차가 적절하게 존재하고, 해당 인증값을 적절하게 관리되고 있음 **미흡**: 개인식별 자체가 부재하면서 개인정보를 노출하고 있으며, 사용된 인증값이 재사용이 가능하거나 변조가 가능한 상태
4-2	안전한 패스워드 사용 여부	❐ 확인 방법 패스워드 복잡도 규칙이 제대로 되어 있는지 여부
		❐ 평가 기준 **양호**: 회원 가입 시 비밀번호 설정에 대한 길이 제한이 제대로 구현되어 있어야 함. 기준은 고객사의 정책을 1차 기준으로 하나, 개인정보를 수집 또는 다루는 시스템의 경우에는 숫자, 영문자, 특수기호 중 2개 이상 혼합할 경우 10자리 이상, 3개 이상 혼합할 경우 8자리 이상으로 규정하고 있음.(개인정보보호법) **미흡**: 비밀번호 길이에 대한 특별한 제한이 없는 경우
4-3	인증 실패 통보 여부	❐ 확인 방법 여러 번의 로그인 실패 시 잠금 설정 또는 담당자 통보 여부
		❐ 평가 기준 **양호**: 패스워드 5회 이상 틀릴 경우 일정 시간 동안 로그인 시도 제한 **미흡**: 인증 실패 시, 로그인 시도 제한
4-4	중요 기능 인증 강화 여부	❐ **확인 방법** 결제금액 및 포인트와 같이 앱에 보여주는 값에 대해 매개변수 조작으로 변경 가능한지 확인
		❐ **평가 기준** **양호**: 매개변수 조작 등으로 중요 기능 인증이 우회가 되지 않는 경우 **미흡**: 매개변수 조작 등으로 중요 기능인증 우회가 가능한 경우
4-5	자체 서명 인증서 탐지 여부	❐ **확인 방법** 인증서가 요구되는 부분에 임의의 자체 서명(self-signed) 인증서로 대체하였을 때 검증하는지 확인
		❐ **평가 기준** **양호**: 버프스위트(Burp Suite) 툴로 인증서를 조작하였을 경우 탐지하는 경우 **미흡**: 버프스위트(Burp Suite) 툴로 인증서를 조작하였을 경우 탐지하지 못하는 경우

3.2.5 권한(Authorization)

표 14. 권한 세부 항목
Table 14. Detailed items for authorization

번호	세부 진단 항목	진단항목 확인 방법 / 항목별 평가 기준
5-1	앱 퍼미션 설정 여부	❐ **확인 방법** 앱 퍼미션이 적절하게 설정되지 않아 앱이 의도하지 않은 권한으로 특정 행위를 할 수 있는지 확인
		❐ **평가 기준** **양호**: App 용도에 맞는 권한이 적절히 설정된 경우 **미흡**: App 용도와 무관하게 과다한 권한이 설정된 경우
5-2	타 사용자 도용 통제 여부	❐ **확인 방법** 사용자 정보를 전달하는 매개변수를 타 사용자로 변경하였을 때 타 사용자 도용 통제를 수행하고 있는지 여부
		❐ **평가 기준** **양호**: 사용자 식별 정보 변조 시, 로그인 사용자의 권한으로 서비스 사용이 가능한 경우 **미흡**: 사용자 식별 정보 변조 시, 타사용자의 권한으로 서비스 사용이 가능한 경우
5-3	기능제한 우회 금지여부	❐ **확인 방법** 매개변수 조작을 통해 제한된 기능을 우회할 수 있는지 확인
		❐ **평가 기준** **양호**: 기능제한 부분에 매개변수 변조 후, 제한된 기능이 우회 되는 경우 (예: 한번 만 등록할 수 있는 앱 평점 부분을 제한 기능 우회하여 여러 번 높은 평점을 등록하는 경우) **미흡**: 기능제한 부분에 매개변수 변조 후, 제한된 기능 우회가 되지 않는 경우
5-4	불필요하거나 사용하지 않는 Activity 제거 여부	❐ **확인 방법** Androidmanifest.xml에 불필요한 activity나 service가 있는지 여부
		❐ **평가 기준** **양호**: 사용자 서비스용 Activity만 존재하는 경우 **미흡**: 서비스 용도 외 테스트나 관리 용도의 불필요한 Activity가 존재하는 경우

3.2.6 세션 관리(Session Management)

표 15. 세션 관리 세부 항목
Table 15. Detailed items for session management

번호	세부 진단 항목	진단항목 확인 방법 / 항목별 평가 기준
6-1	세션 보호 여부	❐ **확인 방법** 사용자 세션이 암호화되지 않거나, 추측 가능한 값으로 구성되어 있는지 확인 (토큰 암호화 여부 확인)
		❐ **평가 기준** 양호: 세션이 암호화 되어 있거나 추측하기 어렵게 구성되어 있는 경우 미흡: 세션이 암호화되어 있지 않거나, 암호화 되어도 쉽게 유추가 가능하도록 되어 있는 경우

3.2.7 예외 관리(Exception Management)

표 16. 예외 관리 세부 항목
Table 16. Detailed items for exception management

번호	세부 진단 항목	진단항목 확인 방법 / 항목별 평가 기준
7-1	안전한 에러 처리 구현 여부	❐ **확인 방법** 고의로 에러를 발생시켰을 때 시스템정보가 노출되는지 확인
		❐ **평가 기준** **양호**: 존재하지 않는 페이지 접근 또는 에러가 발생하도록 특수문자를 입력하였을 때 에러처리를 정상적으로 한 경우 **미흡**: 에러처리가 제대로 되어 있지 않아 시스템 정보가 노출되는 경우

3.2.8 암호화(Cryptography)

표 17. 암호화 세부 항목
Table 17. Detailed items for cryptography

번호	세부 진단 항목	진단항목 확인 방법 / 항목별 평가 기준
8-1	중요 정보 전송시 암호화 여부	❐ **확인 방법** 인증 패킷의 암호화 등의 안전한 송수신 여부 (매개변수를 통해 중요정보 전달 여부

		❒ **평가 기준** **양호**: 중요정보 부분이 암호화가 적용되어 전달되는 경우 **미흡**: 중요정보 부분이 암호화가 되지 않고 전달되는 경우
8-2	전송 시 안전한 암호화 알고리즘 및 키 길이 사용 여부	❒ **확인 방법** 중요 데이터 전송 시, 안전한 암호화 알고리즘 및 키 길이 사용 여부 확인
		❒ **평가 기준** **양호**: 안전한 암호화 알고리즘 및 키 길이에 맞게 전달되는 경우 **미흡**: 취약한 암호화 알고리즘 또는 키 길이가 적절하지 않은 경우

3.2.9 프로토콜(Protocol)

표 18. 프로토콜 세부 항목
Table 18. Detailed items for protocol

번호	세부 진단 항목	진단항목 확인 방법 / 항목별 평가 기준
9-1	안전한 프로토콜 사용 여부	❒ **확인 방법** 안전하지 않은 프로토콜을 이용하여 데이터를 전송하는지 확인
		❒ **평가 기준** **양호**: SSL과 같이 안전한 프로토콜을 사용하여 중요 데이터가 전달되는 경우 **미흡**: 중요정보 전달시 SSL이 적용되어 전달되지 않는 경우

개발된 모바일 앱 보안진단 항목을 개별 앱에 적용시킬 때는 모바일 앱 보안요소와 개별 메뉴를 Matrix로 구성하여 점검 여부를 확인할 수 있다. 가령 아래와 같이 구성할 수 있다.

표 19. 모바일 앱 보안영역을 이용한 진단 예시
Table 19. assessment example using Mobile app security category

보안요소 진단항목	기능/메뉴별 점검 여부	기능/메뉴 외 취약점 여부	대기능 / 소기능1
1-1 앱 자체 보호 여부	불필요	N	N/A
1-2 코드 난독화 여부	불필요	N	N/A
1-3 앱 무결성 검증 여부	불필요	N	N/A
1-4 루팅 머신 인지 체크 여부	불필요	N/A	Y
2-1 앱 구동 파일 내 중요정보 암호화 여부	불필요	N/A	Y
2-2 코드 내 중요정보 암호화 여부	불필요	N/A	Y
2-3 안전한 암호화 알고리즘 및 키 길이 사용 여부	불필요	N/A	Y
3-1 특수문자 필터링 여부	필요	N/A	N
3-2 악성파일 업로드 차단여부	필요	N/A	Y
3-3 파일다운로드 경로 검증여부	필요	N/A	Y
3-4 매개변수 조작 검증 여부	필요	N/A	N
4-1 강화된 인증 방식 사용 여부	필요	N/A	Y
4-2 안전한 패스워드 사용 여부	필요	N/A	Y
4-3 인증 실패 관리자 통보 여부	필요	N/A	Y
4-4 중요기능 인증 강화 여부	필요	N/A	Y
4-5 자체서명 인증서 탐지 여부	필요	N/A	Y
5-1 앱 퍼미션 설정 여부	불필요	N	N/A
5-2 타 사용자 도용 통제 여부	필요	N/A	Y
5-3 기능제한 우회 금지여부 (평점제한, 평가횟수제한 등)	필요	N/A	Y
5-4 불필요하거나 사용하지 않는 Activity 제거 여부	불필요	N	N/A
6-1 세션 보호 여부	필요	N/A	Y
7-1 안전한 에러처리 구현 여부	필요	N/A	Y
8-1 중요정보 전송시 암호화 여부	필요	N/A	Y
9-1 안전한 프로토콜 사용 여부	필요	N/A	Y

3.3 실증 테스트

본 논문에서 제시한 방법론의 실증 테스트를 위해 국내 대기업 통신사에서 개발한 모바일 앱 15종에 대해 점검을 수행하였다. 본 방법론의 범용성을 확인하기 위하여 다양한 특성을 가진 15개의 앱

을 대상으로 하였고 진단에 사용된 모바일 앱의 유형은 일반 서비스(회원 가입, 로그인 등)를 제공하는 앱, 결재 모듈을 포함하고 있는 앱, 게임 앱, 백그라운드로 동작하는 앱, 다른 앱에 의해 호출되는 앱 등으로 구성하였다.

다음 <표 20>, <표 22>는 본 논문의 방법론을 적용하여 국내 대기업 통신사에서 개발한 모바일 앱 15 종 각각에 대해 각 항목 별 취약점을 분석한 결과이다.

표 20. 모바일 앱 보안영역을 이용한 진단 결과(앱1 ~ 앱5)
Table 20. assessment example using Mobile app security category (App1 ~ App5)

보안요소 진단항목	앱1	앱2	앱3	앱4	앱5
1-1	X	X	X	X	X
1-2	X	X	X	X	X
1-3	X	X	X	X	X
1-4	X	X	X	X	X
2-1	X	X	X	-	X
2-2	X	X	X	-	X
2-3	O	-	-	-	-
3-1	X	-	-	-	-
3-2	-	-	-	-	-
3-3	-	-	-	-	-
3-4	X	-	-	-	-
4-1	O	-	-	-	-
4-2	O	-	-	-	-
4-3	X	-	-		-
4-4	X	-	-	-	-
4-5	X	-	-	-	-
5-1	X	X	X	X	X
5-2	X	X	X	X	X
5-3	X	-	-	-	-
5-4	X	X	X	X	X
6-1	O	-	-	-	-
7-1	X	-	-	-	-
8-1	X	-	-	-	-
8-2	X	-	-	-	-
9-1	X	-	-	-	-
O: 양호, X: 취약, -: 해당 없음					

앱1: 회사 대표 서비스 앱(사용자 로그인, 결제기능, 관리자 기능 모두 존재)
앱2~앱5: 백그라운드로 동작하는 앱(별도 화면이 존재하지 않음)

표 21 모바일 앱 보안영역을 이용한 진단 결과(앱6 ~ 앱10)
Table 21 assessment example using Mobile app security category (App6 ~ App10)

보안요소 진단항목	앱6	앱7	앱8	앱9	앱10
1-1	X	X	X	X	X
1-2	X	X	X	X	X
1-3	X	X	X	X	X
1-4	X	X	X	X	X
2-1	-	-	-	-	-
2-2	X	X	-	X	-
2-3	-	-	-	-	-
3-1	O	O	X	-	O
3-2	O	-	-	-	-
3-3	-	X	O	-	-
3-4	O	O	O	-	O
4-1	X	X	-	-	O
4-2	-	-	-	-	-
4-3	-	-	-	-	-
4-4	-	O	O	-	-
4-5	X	X	X	-	X
5-1	-	-	-	-	-
5-2	X	X	-	-	X
5-3	-	-	X	-	-
5-4	O	O	O	-	O
6-1	X	X	O	-	X
7-1	O	O	-	-	-
8-1	X	X	-	-	X
8-2	-	-	O	-	X
9-1	O	O	O	-	O
O: 양호, X: 취약, -: 해당 없음					

앱6~앱12, 앱15: 일반 서비스를 제공하는 앱(사용자 로그인, 기본 서비스)
앱13: 결제 모듈을 제공하는 앱
앱14: 다른 앱을 보호하는 기능을 가지고 있는 앱 (별도 화면이 존재하지 않음)

조사 결과, 모바일 보안 요소 중, “1 앱보안”이 취약율이 100%로 가장 낮았으며, 상대적으로 많이 알려진 “3 입력값 검증”이 평균 19.25%로 매우 높음을 알 수 있었다.

분석 결과 기존 4가지 방법론으로 점검 시 누락되었던 부분을 상당 수 발견할 수 있었다. 15종의 앱 중 하나에서 상품 결제 기능을 수행하는 메뉴가 5개가 있는 경우, 앞서 설명한 다른 방법론으로

는 1개 메뉴에 대한 취약점만 발견 가능하였으나 본 진단 방법론을 통해서는 5개 메뉴 중 4개는 취약한 것으로, 1개는 안전한 것으로 확인되었다.

표 22 모바일 앱 보안영역을 이용한 진단 결과(앱11 ~ 앱15)
Table 22 assessment example using Mobile app security category (App11 ~ App15)

보안요소 진단항목	앱 11	앱 12	앱 13	앱 14	앱 15	취약률
1-1	X	X	-	X	X	100
1-2	X	X	-	X	X	100
1-3	X	X	-	X	X	100
1-4	X	X	-	X	X	100
2-1	-	X	-	-	-	100
2-2	X	X	-	-	X	100
2-3	-	-	-	-	-	0
3-1	X	O	O	-	O	33
3-2	-	-	-	-	-	0
3-3	O	-	-	-	-	33
3-4	O	O	O	-	O	11
4-1	O	O	O	-	X	37
4-2	-	-	-	-	-	0
4-3	-	-	-	-	-	0
4-4	X	-	X	-	-	60
4-5	-	X	X	-	X	100
5-1	-	-	-	-	-	0
5-2	O	O	X	-	X	83
5-3	-	-	-	-	-	100
5-4	O	O	O	-	-	41
6-1	O	-	-	-	X	57
7-1	O	O	O	-	O	14
8-1	X	O	O	-	X	75
8-2	-	-	X	-	-	75
9-1	X	O	O	-	O	22
O: 양호, X: 취약, -: 해당 없음						

이와 같이 기존 방법론에서는 누락될 수 있는 부분이 본 방법론을 통해서는 누락 없이 진단이 가능하였다. 또한, 기존에는 진단자들의 주관적인 판단으로 취약성 유무를 판단하였으나, 본 연구에서 제시한 방법론은 보다 객관적인 진단 결과를 도출 가능하게 하였다.

아래 표는 기존 4가지 방법론과 본 연구의 방법론의 적용 결과 항목 별 취약점 발견 여부를 비교한 표이다.

표 23 각 진단방법론 사용결과 비교 표
Table 23 Compare result for each security assessment methodology

보안요소 진단항목	본 방법론	A 방법론	B 방법론	C 방법론	D 방법론
1-1	O	O	O	O	O
1-2	O	O	O	O	O
1-3	O	O	X	O	O
1-4	O	O	O	O	O
2-1	O	O	O	O	O
2-2	O	O	O	O	O
2-3	O	O	X	O	O
3-1	O	△	△	△	△
3-2	O	△	△	△	△
3-3	O	△	△	△	△
3-4	O	△	△	△	△
4-1	O	△	△	△	△
4-2	O	△	△	△	△
4-3	O	△	△	△	△
4-4	O	△	△	△	△
4-5	O	△	△	△	△
5-1	O	O	O	O	O
5-2	O	△	△	△	△
5-3	O	X	X	X	X
5-4	O	X	X	X	X
6-1	O	△	△	△	△
7-1	O	△	△	△	△
8-1	O	△	△	△	△
9-1	O	△	△	△	△
메뉴별 진단	O	X	X	X	X

△: 일부 발견 가능
X: 발견 불가능
A방법론: 국내 국가기관(KISA, 안전행정부 기준)
B방법론: 국내 민간기관(통신사의 모바일 앱 진단항목)
C방법론: 해외 연구기관(OWASP)
D방법론: 해외 민간기관(Deloitte)

3.4 본 연구의 한계

본 연구에서 제시한 모바일 앱 보안진단 방법론은 각각의 모바일 앱의 고유 기능이나 특성을 고려하기에는 어려운 점이 있어 이를 바탕으로 한 고도화된 공격 등은 detection하기 어려운 점이 있

다. 이러한 한계는 본 논문에서 개발한 범용 진단 방법론을 기본으로 하여, 해당 앱 고유 기능에 따른 시나리오 별 심화진단을 통해 보완할 수 있을 것으로 보인다.

4. 결론

본 논문은 기존 모바일 앱 보안진단 방법론들의 단점을 보완하여 보다 객관적이고 적용이 용이한 범용 진단방법론을 제시하였다. 다양한 유형의 앱에 대한 진단을 위해 모바일 앱 보안진단 항목으로는 대분류 3개(Device, App Function, Network)요소와, 3개 요소에 대한 세부 9개 영역(App Security, Sensitive Data Storage, Input Validation, Authentication, Authorization, Session Management, Exception Management, Cryptography, Protocol)을 개발하였고 각 진단 영역별 앱에 대해 메뉴 별 진단과 진단 항목에 대한 객관적인 평가 기준을 도출하였다.

또한 진단방법론의 실증테스트를 위하여 국내 대기업 통신사에서 개발한 다양한 특성을 가진 15개의 앱을 대상으로 분석을 수행하였고 기존 4개 방법론 대비 진단 결과의 우수성을 확인할 수 있었다. 본 방법론은 범용성, 실무적용 용이성과 취약점 발견의 우수성을 바탕으로 다양한 기업들의 모바일 앱 취약점 점검에 활용될 수 있을 것으로 기대된다.

References

[1] Google app market, app down 50 billion, http://www.edaily.co.kr/news/NewsRead.edy?SCD=JE41&newsid=02309126602875176&DCD=A00504&OutLnkChk=Y.

[2] TrendMicro, *The volume of malicious and high-risk Android apps will hit 1 million in 2013,* http://blog.trendmicro.com/trendlabs-security-intelligence/predictions-for-2013.

[3] National institution's information security management on the smart phone use environment, 2010.

[4] IDC, Android and iOS Continue to Dominate the Worldwide Smartphone Market with Android Shipments Just Shy of 800 Million in 2013, According to IDC, http://www.idc.com/getdoc.jsp?containerId=prUS24676414http://www.idc.com/getdoc.jsp?containerId=prUS24676414.

[5] January 2014 Web Server Survey, http://news.netcraft.com/archives/2014/01/03/january-2014-web-server-survey.html.

[6] Gartner Says That by 2017, 25 Percent of Enterprises Will Have an Enterprise App Store, http://www.gartner.com/newsroom/id/2334015.

[7] Why iOS is safer than Android, http://www.sophos.com/en-us/security-news-trends/security-trends/malware-goes-mobile/why-ios-is-safer-than-android.aspx.

[8] Market Share Analysis: Information Security Consulting, Worldwide, 2013, http://www.gartner.com/technology/reprints.do?id=1-1XCWAT0&ct=140714&st=sb.

[9] The Forrester Wave™: Information Security Consulting Services, Q1 2013, http://www.forrester.com/The+Forrester+Wave+Information+Security+Consulting+Services+Q1+2013/fulltext/-/E-RES77361.

[10] Deloitte ranked #1 in Global Consulting by Kennedy for the fourth consecutive year, http://www.investobharat.com/deloitte-ranked-1-in-global-consulting-by-kennedy-for-the-fourth-consecutive-year/

[11] KISA, Security assessment guidelines for

mobile e-government service app source code
[12] OWASP Advanced Mobile Application Code Review Techniques, https://www.owasp.org/images/4/40/OWASP_Advanced_Mobile_Application_Code_Review_Techniques.pptx.
[13] OWASP Mobile Security Project, https://www.owasp.org/index.php/OWASP_Mobile_Security_Project.
[14] Deloitte Mobile Application Security Best Practice – internal resource.

안드로이드 모바일 앱 보안진단 방법에 관한 연구

김경곤[1], 김휘강[1], 김은진[2]
[1]*고려대학교 정보보호대학원*
[2]*경기대학교 국제산업정보학과*

요 약

2007년 애플에서 아이폰(iPhone)이 출시되고, 같은 해 11월 안드로이드 1.0 알파 버전이 출시되었다. 2013년 안드로이드 앱 다운로드 수가 500억 건을 넘을 정도로 모바일 앱을 사용하는 인구가 폭발적으로 증가하였다. 모바일 앱이 급격하게 증가하는 것에 반해 기존에 모바일 앱 보안진단 방식의 한계로 인해, 여전히 많은 모바일 앱들이 취약한 상태로 개발되고 마켓에 배포되기에 이르렀다. 본 논문에서는 모바일 앱을 개발 및 운영할 때 점검해야 하는 안드로이드 기반의 모바일 앱 진단 방법론을 개발하였다. 그리고 개발한 안드로이드 모바일 앱 진단 방법론을 국내 대기업 통신사에 적용해본 결과, 기존 방법론 대비 진단 항목에 대한 누락 없이 효율적으로 진단할 수 있었다. 이를 통해 본 논문에서 소개한 방법론이 모바일 앱 보안 진단을 수행하기 위한 효과적인 방법임을 알 수 있었다.

Kyoung Gon Kim received the bachelor's degree in the Department of Computer Science from the Soongsil University in 2008. He has finished courses the master degree in Graduate school of Information Security, Kora University. He is senior manager in Deloitte Anjin, and in charge of cyber security technical leader in Deloitte Korea. Prior to join Deloitte, he worked for Samil PwC as an information security consulting, internal auditor, and IT auditor.
E-mail address: anesra@korea.ac.kr

Huy Kang Kim received his B.S. in Industrial Management, M.S. and Ph.D. in Industrial Engineering from KAIST (Korea Advanced Institute of Science and Technology) in 1998, 2000 and 2009, respectively. He was a technical director (TD) and head of Information security department in NCSoft and he is now an assistant professor in Graduate school of Information Security, Korea University.
E-mail address: cenda@korea.ac.kr

Eunjin Kim received her B.S., M.S. and Ph.D degrees in Management from Korea Advanced Institute of Science and Technology (KAIST) in 2001. She is an associate professor at Kyonggi University. Her current research interests include economic analysis of digital content, information system and effects of the digital divide.
E-mail address: ejkim777@kgu.ac.kr