중소기업융합학회 논문지 제5권 제1호 pp. 37-40, 2015 ISSN 2234-4438

모바일 앱 위·변조 공격 및 대응방안

정현수
숭실대학교

Countermeasure of Mobile App tamper attack

Jung Hyun Soo[1*]
[1*]Information Security, Songsil University

요 약 최근 스마트폰 사용자가 늘어남에 따라 사소한 개인적 용도에서부터 금전거래까지 사용범위 또한 늘어나는 추세이다. 이에 발 맞춰서 해킹기술들도 다양해지는데 그중에서도 이슈화되고 있는 것이 위·변조이다. 모바일 앱 위·변조는 본래의 앱과 비슷하게 만들어내 사람들을 속이는 용도로 사용하는 해킹기술이다. 이 기술은 보안의 3대요소인 기밀성, 무결성, 가용성중에 무결성을 침해한다. 이 경우 신뢰성이 저하되고 앱 자체의 위험성 증가와 가치가 떨어지게 된다. 이를 통해 정보에 권한이 없는데도 마음대로 개인정보를 가져갈 수 있게 되고 심하게는 자산손실에 까지 사회적 영향을 끼치게 된다. 즉, 본 내용에는 위·변조의 정의와 원리, 마지막으로 대응방안까지 정리하였다. 이를 통해 현재 상황과 어떻게 예방할 수 있는지 알게 한다.

주제어 : 위·변조, 금융 피해, 앱, 방지 기술

Abstract Recently range of use also being increase along with smart phone users growing. And keep pace with hacking technician is increasing inter alia tamper technician has issued. This technician infringe integrity on three element of security of data. In this case reliability has deteriorated, the app itself has increased danger, and it's value has reduced. This can affect like take information even though don't have any authority to information or hemorrhage at large in this country. In other words, I has been arranged tamper's definition to

Key Words : tamper, finance demage, App, prevention technique

1. 서론

스마트 폰 사용자가 급격히 늘어남에 따라 다양한 앱들이 개발되고 모바일 시장에 나오고 있다. 하지만 이용률에 비해 그에 따른 해킹위협에 대해서는 아직 대책이 부족하고, 인식 또한 낮다. 즉, 스마트 폰 보안에 허점이 생기게 되고 피해자 스마트 폰을 통해 공격자는 각종 이득을 손쉽게 취할 수 있다. 너무 공급에만 치중하다 보니 사람들의 윤리의식이나 피해에 대한 위험 인식이 제대로 되지 못했기 때문이다. 특히, 공격 중에서도 앱에 관련된 공격이 손쉽게 사용자의 각종 정보를 탈취하는데 사용된다. 그리고 위·변조는 모바일 악성코드를 유포하는데 매우 일반적으로 사용되는 기법이다.

2. 앱 위·변조

2.1 앱 위·변조 기술

Received 2015-02-16 Revised 2015-02-27 Accepted 2015-03-10
*Corresponding author: Jung Hyun Soo (hsj6552@hanmail.net)

앱 위변조 기술은 위에서 말한 공격의 종류 중 하나로 불법적인 악성코드를 넣어 만든 앱을 원래 기존의 앱과 바꿔놓고, 마치 해당 앱이 맞는 것처럼 깔린 뒤, back space에서 여러 정보들과 사생활 관련된 것들을 빼내온다. 설치되는 경로는 주로 웹하드나 블랙마켓 등의 불법 사이트를 통해 다운로드 하는 경우, 보안 검증이 되어있는지 모르는 게임 앱, 블로그 등에 올라와있는 덤프파일들을 다운 받을 때로 볼 수 있다. 그리고 한 번 해킹이 되면 이후 해당 스마트폰으로 사용하는 기록들이 다 넘어가게 되므로 뱅킹을 사용할 경우 금전적인 피해로 까지 이어질 수 있다. [1]

대표적으로는 2011년에 Droid-Dream 악성 앱을 들 수 있다. 이 앱은 정상적인 앱을 위·변조해서 악성코드를 심고 안드로이드 마켓에 재배포 한 뒤 시스템 정보를 빼내고, 외부 애플리케이션 원격 설치하는 등의 공격을 한다.

여러 운영체제들 중에서도 특히 안드로이드기반이 공격의 표적이 되기 쉬운데 이는 APK(애플리케이션 패키지 파일)를 설치 하는 것이 가능해서 악성 앱의 배포가 쉽고, 자바언어로 구현되어 역공학이 쉬워서 앱 위·변조를 통한 악성코드 삽입과 재배포까지 쉽기 때문이다. 위·변조 공격의 최적의 장소라고 볼 수도 있다. 이러한 문제는 금융 관련된 앱에서 더욱 심각한 위험요소가 된다. 만약 금융 앱을 악성코드를 삽입해 재배포 할 경우 개인 뿐만 아니라 기업 손실에 까지 영향을 끼치게 된다.[2]

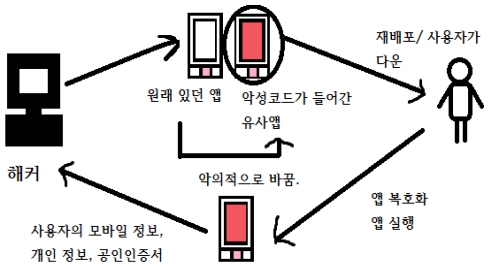

Fig. 1. 모바일 앱 위조 공격

Fig. 1은 앱 위조 공격의 예다. 위의 설명을 한 번에 볼 수 있게 첨부 하였고, 대부분의 위·변조 공격이 이런 형태로 이루어짐을 알 수 있다.

2.2 위 · 변조 기술 순서

앱을 위조하기 위해서 우선 해커는 대상 앱을 하나 선정해서 마켓이나 웹하드 등에서 입수한다. 그리고 unzip, ded, dex2jar, JDGUI 같은 도구를 사용해서 앱의소스 코드를 추출하고, 분석한다. 이후 기존 앱의 특정부분에 작성해둔 악성코드를 삽입하거나 특정 파일에 대한 존재 유무를 체크하는 코드를 무력하게 만다는 등 보안 기능을 하지 못하게 작업을 한 뒤, 변경된 전체 소스 코드를 서명해서 위·변조 된 새로운 앱으로 생성한다. 특히, 공격자들은 악성코드의 빠른 유포를 위해 앵그리버드 등의 모바일 게임 나 인스타그램, 페이스 북등의 sns와 같은 인기있는 앱이나 구글 월렛과 같은 신뢰성 있는 앱 위·변조 대상으로 정하고 있다. Arxan 보고 자료에 따르면, 안드로이드 마켓 상위 100개의 유료 앱에 위·변조 애플리케이션이 모두 존재하는 것으로 밝혀졌고, 구글 플레이에 뱅킹 앱을 위·변조한 앱이 다수 등록된 것으로 보고되었다.

2.3 피해 사례

주로 위·변조는 무결성의 침해 여부에서 일어난다. 앱의 디자인이나 겉의 기능들이 아무리 뛰어나도 취약점이 들어 있거나 무결성이 침해된다면 보안사고가 일어날 수 있고 기업 손실로 이어지기 때문에 무결성 확보는 꼭 필요하다.

피해를 받은 사례를 보면 '아이러브커피'라는 모바일 게임을 그대로 복제해서 소스코드를 해킹해 '커피러버'라는 게임을 불법적으로 만들어낸 사건을 들 수 있다. 이뿐만 아니라 모바일게임의 결제모듈을 해킹해서 불법으로 게임머니를 빼내려는 시도가 셀 수 없이 발생하고 있다. 여기서 봐도 알 수 있듯이 주로 금전적인 목적이 주를 이루고 있다. 이로 인해 금융권에서 제일 이 공격에 대해 민감하게 반응하고 있다. 전자 금융 앱 서비스등을 제공하고 있기 때문에 현실적으로 피해들이 속출하기 때문이다. 그리고 한 음악 사이트에서도 발생했는데 음악 파일의 불법 복제를 방지하는 DRM(Digital Rights Mamagement)을 해제하는 위·변조 앱도 있었다. 뿐만 아니라 이 공격은 개인정보유출에 큰 위험이 생길 수 있다. 아이디 비밀번호를 도용당하고, 개인적인 정보들이 공격자의 모바일이나 컴퓨터로 넘어가게 된다. 오히려 이 부분이 더 위험할 수 있다. 금융앱은 보안이 강해졌다 해도 앱 마켓에 나오는 앱의 수는 너무나 많아서 그 중 어떤 앱을 통해 악성코드가 섞인 위·변조 앱인지 알기가 어렵다.

3. 대응 방법

3.1 위 · 변조 탐지 기법

공격에 대응하기 위해서는 위·변조 앱이 어떤 것인지 일단 알아내야 한다. 그러기 위해서 탐지 기법에 대한 연구들이 많아지고, 세 가지 정도가 알려져 있다.

1. 유사성에 기반을 둔 탐지 법

위·변조 앱 경우 기존 앱을 바탕으로 특정 소스 코드를 추가, 삭제, 수정해서 생성시킨 앱이므로 이 두 앱 사이에는 일정 부분의 소스코드를 공유하게 된다. 이 유사성에 기반을 두고 작성자는 다르지만 소스코드가 유사한 앱들을 찾아내는 방법을 주로 연구한다.

그리고 이런 앱은 공식적인 마켓보다는 마켓에 등록된 앱에 대한 검증 작업이 이루어지지 않는 제3자 마켓을 통해 많이 유통된다. 여기서 고안된 방식으로 제3자 마켓에 등록된 앱과 공식 마켓에 등록된 앱 사이에 유사성 측정을 해서 위·변조된 앱을 찾아내는 시스템이 최근에 제안 되고 있다. DroidMOSS라고 하며 위·변조된 앱 작성 시 해커에 의해 수행 될 수 있는 소스코드 내에 함수나 변수 명 변경 등의 회피기술을 고려해서 각 앱의 유사성 비교대상으로 소스코드가 아닌 OP코드(컴퓨터 기술에서 기계어의 일부이며, 수행할 명령어를 지정하는 역할을 함. 한 Byte씩 읽어 들여 명령어를 수행함.)들 만을 추려내서 사용하는 특징을 가지고 있다. 그리고 fuzzy hashing 알고리즘(기존의 해쉬 방법을 통해 찾지 못했던 잠재적인 유죄를 증명할 문서들을 검색하는 알고리즘. 기존 해쉬의 무결성 확보의 측면뿐만 아니라 원본과의 유사도를 파악하기 위해 만들어 졌음.)기반의 fingerprint (각 앱의 fingerprint는 유사성 측정 시간 단축과 정확도 모두를 만족시키기 위해 fuzzy hashing 알고리즘을 통해 생성됨.)를 생성해서 유사도를 비교해 위·변조 앱을 찾는 방법이다.

2. 무결성을 검증하는 방법

이 방법은 앱이 실행되는 시점에 위·변조 여부를 탐지하는 방법으로 무결성 검증 방법이 적용된 안드로이드 앱 경우 앱 내부에 위·변조 여부를 탐지하기 위한 특정코드를 포함하고 있다. 위의 코드는 실행 시점에 앱의 해쉬값을 측정하고 외부서버에 존재하는 올바른 해쉬값과 비교해서 위·변조 여부를 파악하고, 위·변조로 탐지될 경우 앱의 동작을 중지시키거나 제한한다.

현재 모바일 뱅킹 등의 금융권, 보험 증권 및 모바일 게임 앱에 적용되고 있다. 특히, 서버를 통해 서비스가 제공되고 있는 앱에 적합하다. 그러나 이 방법에도 주의할 점이 있다. 무결성을 검증하는 로직도 앱 내부에 들어 있어서 이 로직까지 위·변조 될 위험이 있기 때문이다. 다행인 이 부분은 위 로직을 코드 난독화 기술을 토해 보호하는 것으로 위험성을 줄이고 있다는 점이다. 또 해당 로직을 앱이 아닌 플랫폼에 넣는 것도 하나의 방법으로 보고 있다.

3. 소스코드 추출 및 분석을 방지하기 위한 코드 난독화 기술

난독화는 앱 소스코드가 역공학을 통해 분석되는 것을 어렵게 하기 위한 기법이다. 크게 구획 난독화(소스코드 분석 시 도움이 될 수 있는 정보를 제거하는 기법. ex. 함수/변수 등의 식별자를 의미 없는 값으로 바꿈, 디버깅 정보 제거), 제어 흐름 난독화(수행되지 않는 불필요한 코드를 삽입하는 방식 ex. 프로그램의 제어 흐름을 변형해서 문맥판단을 어렵게 만듦), 데이터 난독화(앱 내에 포함된 데이터를 유추하기 어렵게 하는 기법 ex. 기존 앱에 포함된 텍스트를 암호화하는 방식), 예방 난독화(앱이 역공학에 이용되는 것을 미리 막기 위해 특정 로직을 삽입하는 기법 ex. 디버거(역공학시 사용됨)들의 사용을 탐지해서 앱의 동작을 중지시키거나 제한하는 로직을 삽입)로 나눌 수 있다.

3.2 대응

우선 루팅 및 탈옥 여부를 체크해야 하고, 시스템 구축을 통해 직원들이 검색한 개인 정보 포함 파일이 고객 정보보호 관리시스템과 연동해서 관리 할 수 있도록 하고, 개인정보가 미포함된 파일과 구분될 수 있도록 명확히 한다.

그리고 현재 대응 상태를 보면 각 보안회사에서 저마다의 방지 솔루션을 내놓고 있고, 공공기관이나 금융권에서 적극적으로 도입하고 있다. 현재 나와있는 제품으로는 일단 NSHC의 앱프로텍트(앱 위·변조 방지기능과 앱 위·변조 탐지 및 대응 두가지로 나누어져 있어 보다 정확하게 보호가능)나 제큐어 앱쉴드 등이 있다.

여기에 더해서 일반 사용자들의 위험에 대한 인식까지 함께 바뀌어야 한다. 아무리 솔루션이나 대응 기술이

나온다 해도 사용자들이 제3의 경로를 통해 안전여부가 판별되지 않은 앱을 다운받아 실행시킨다면 제대로 보호를 받기 힘들고 피해또한 줄어들지 않을 것이다.

4. 결론

많은 기관과 금융회사, 보험회사에서 솔루션을 도입하고 있지만 그럼에도 불구하고 아직 많은 앱들이 보호되지 않고 있다. 현재 나와 있는 앱들이 너무 많아서 검증여부가 어렵고, 결제방식이 보호되고 있는지도 알 수 없다. 그리고 많은 사람들이 사용하고 있는 앱 일수록 안전할 거라는 방심도 할 수 있는데 마켓에 올라와 있는 앱들은 얼마든지 해킹을 통해 가짜로 둔갑할 수 있다. 물론 메신저를 통해 보내진 URL을 가지고 앱을 실행시키게 하는 등 우리 도처에는 많은 위험성이 있다.

이에 대한 올바른 인식과 대처방안을 가져야 제대로 막을 수 있다고 생각한다.

REFERENCES

[1] Symantec Corp. (2011). Symantec Internet Security Threat Report - 2010, Internet Security Threat Report Volume 16.Technical Report. Available: http://www.symantec.com/ business/threatreport/

[2] Nataraj, L., Karthikeyan, S., Jacob, G., And Manjunath,B. (2011). Malware images: Visualization and automatic classification. Proceedings of Visualization for Cyber Security (VizSec). 2011.

[3] Quist, D.A., and Liebrock, L.M. (2009). Visualizing compiled executables for malware analysis. 6th International Workshop on Visualization for Cyber Security, 2009 (VizSec 2009). pp. 27-32.

[4] Jiang, X., Wang, X., And Xu, D. (2007).Stealthy malware detection through vmm-based "out-of-the-box" semantic view reconstruction. Proceedings of the 14th ACM conference on Computer and communications security (CCS '07). 2007. New York, NY, USA: ACM. 128-138.

[5] Nair, V.P., Jain, H., Golecha, Y.K., Gaur, M.S., And Laxmi, V. (2010).MEDUSA: MEtamorphic malware dynamic analysis using signature from API. Proceedings of the 3rd international conference on Security of information and networks (SIN '10). 2010. New York, NY, USA: ACM. 263-269.

[6] Trinius, P., Holz, T., Gobel, J., And Freiling, F.C. (2009).Visual analysis of malware behavior using treemaps and thread graphs. 6th International Workshop on Visualization for Cyber Security, 2009 (VizSec 2009). Oct 2009. 33-38.

[7] Zhang, F.Y., Qi, D.Y., And Hu, J.L. (2010).Using IRP for Malware Detection.Recent Advances in Intrusion Detection in Lecture Notes in Computer Science.Springer Berlin / Heidelberg.514-515; 2010.

[8] Ahmed, I., And Lhee, K.S. (2011).Classification of packet contents for malware detection. Journal in Computer Virology, 279-295.

[9] Skrzewski, M. (2011). Flow Based Algorithm for Malware Traffic Detection. Computer Networks in Communications in Computer and Information Science.Springer Berlin Heidelberg.271-280; 2011.

저 자 소 개

정 현 수(Jung Hyun Soo) [정회원]

- 1982년 2월 : 숭실대학교 전자계산학과 학사
- 1991년 2월 : 숭실대학교 컴퓨터학과 석사
- 1995년 2월 :숭실대학교 컴퓨터학과 박사
- 1982년 2월 ~ 2005년 11월 : ETRI 책임연구원
- 2006년 2월 ~ 2011년 3월 : TANC CTO
- 2009년 2월 ~ 2012년 2월 : 한남대학교 경영정보학과 겸임교수
- 2012년 4월 ~ 현재 : 숭실대학교 정보보안학과 교수

<관심분야> : 정보보호 정책, 컴퓨터 비젼