정 보 보 호 학 회 지
제26권 제3호, 2016. 6

안전한 모바일 게임서비스를 위한 보안 점검항목

강 선 명*

요 약

최근 다양한 모바일 기기의 보급과 함께 향상된 모바일 컴퓨팅 환경으로 인해 게임서비스의 트렌드가 빠르게 모바일 환경으로 변화하고 있다. 그러나 모바일 게임서비스의 그 역사가 아직까지 길지 않아 보안 수준평가를 위한 점검항목들이 충분히 개발되지 못하였다. 이에 기존에 알려진 모바일 게임 관련 보안 취약점을 분석하여 게임 내 보안수준을 평가하기 위한 사전 점검항목을 열거해보도록 한다.

Ⅰ. 서 론

온라인게임 서비스의 경우 다양한 해킹기법과 이를 막기 위한 여러 가지 보안기술이 빠르게 진화하고 있는 영역이라고 할 수 있다. 그러나 PC 온라인 게임에 비해서 모바일 게임의 경우 그 역사는 짧아 해킹기법은 빠르게 발전하고 있으나, 이에 비해 해킹대응 기술은 공격기술의 진화속도를 따라가지 못하고 있어 많은 모바일 게임 서비스 사업자들은 다양한 해킹에 노출되고 있고 이로 인해 많은 사업적 손실을 입고 있다.

또한 모바일 플랫폼의 주류를 차지하고 있는 안드로이드 의 경우 OS자체의 취약점이 꾸준히 보고되면서 이를 이용하는 해킹시도로 모바일 서비스의 위협이 지속적으로 증대하고 있다.

하지만 모바일 게임의 경우 PC게임에 비해 개발기간이나 게임 서비스의 라이프사이클이 상대적으로 짧기에 보안을 고려하여 설계를 하거나 별도의 비용을 들여 3rd party 보안솔루션을 도입하는 경우도 드물다. 또한 모바일 게임 서비스만을 전문으로 하는 퍼블리싱 기업이 대두 되면서 게임 개발과 운영의 주체가 구분되어 게임 내 보안정책이 제대로 적용되지 않는 경우도 허다하다.

최근 들어서는 개인정보의 중요성이 강조되면서 게임서비스에서 모바일플랫폼 내의 개인정보의 이용이나 시스템 내 민감한 시스템자원(예: 위치정보, 주소록) 을 이용 하는 것에 대해서 매우 제한적이며 민감하게 취급하고 있다.

Ⅱ. 모바일게임 내의 보안위협의 분류

모바일게임 내의 보안적 위협유형을 분류하면 OS를 포함하는 플랫폼의 취약성을 이용하는 보안위협과 게임 내의 보안위협으로 분류 할 수 있다.

2.1. 모바일 환경의 보안위협

OWASP에서 제시하고 있는 Mobile Application의 보안위협은 아래와 같다.

〔표 1〕 OWASP mobile top10 Risk 2014

유형	설 명
M1	Weak Server Side Controls
M2	Insecure Data Storage
M3	Insufficient Transport Layer Protection
M4	Unintended Data Leakage
M5	Poor Authorization and Authentication
M6	Broken Cryptography
M7	Client Side Injection
M8	Security Decisions Via Untrusted Inputs
M9	Improper Session Handling
M10	Lack of Binary Protections

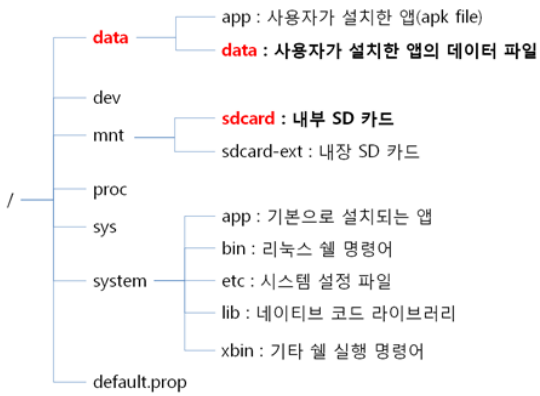
* 펜타시큐리티시스템

이중 모바일 게임과 연관성이 높은 항목에 대해서 세부적인 내용을 살펴본다.

M2 - 안전하지 않은 데이터 저장

모바일 단말환경에서 저장된 중요정보는 물리적 도난이나 공격자에 의한 권한 탈취 시 외부로 유출될 우려가 있다.

특히나 게임플레이를 위한 환경설정파일이나 유저의 플레이 저장파일(save file)을 조작하거나 복제할 수 있다면 게임서비스 전반에 악영향을 미칠 수 있다.



〔그림 1〕 안드로이드 환경의 데이터 저장 구조

M3 - 취약한 전송계층의 보호

네트워크를 통한 데이터 전송 계층에서 보호가 이루어지지 않는다면 전송정보가 노출될 수 있다.

최근 모바일 게임서비스의 추세가 web을 기반으로 하다 보니 서버와 통신 시 http 프로토콜을 사용하는 추세가 늘고 있다. 하지만 모바일 게임의 환경적 제약(통신 데이터 증가, 서비스 성능저하 등)으로 https등 보안 채널(secure channel)를 설정하는 것을 꺼리기도 한다.

일부 게임에서는 중요 데이터를 암호화 하고 있다고 하고 있으나 실제로 확인해보면 XOR 방식을 채택하거나 취약한 암호 알고리즘을 쓰는 경우가 허다하다. 아무래도 게임 개발자가 암호학에 대해 관심 있는 경우가 적기에 보안을 적용했으나 그 내용은 취약한 경우가 많다.

M7 - 클라이언트환경의 injection

클라이언트 환경이 탈옥(jailbreak)이나 루팅(rooting)으로 취약해진다면 application의 주요 정보를 가로채거나 조작할 수 있다.

이는 뒤에서 다시 보게 될 memory hacking tool 사용이나 매크로 사용에 시작점이 될 수 있다.

이러한 공격을 하기 위해서는 우선적으로 단말환경에서 상승된 권한을 획득해야 한다. 이미 다양한 형태의 탈옥이나 루팅에 대한 정보가 인터넷에 나와 있기에 사용자는 마음만 먹으면 손쉽게 본인의 휴대폰을 탈옥/루팅을 통해 조작할 수 있다.

M9 - 부적절한 세션관리

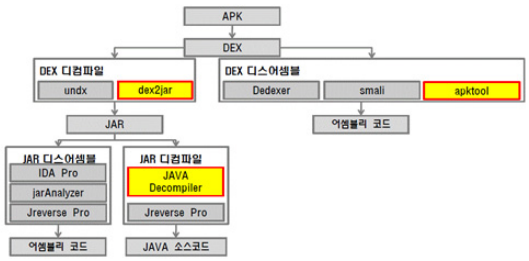
많은 류의 게임들이 경험치와 아이템이라는 차별화 요소를 포함하고 있기에 다른 사용자의 세션정보를 취득할 수 있다면 계정도용 등의 위협이 발생 할 수 있으며 추가적인 부정적인 사례들이 발생 할 수 있다.

특히나 모바일 게임의 경우 개인이 다수의 모바일 기기를 가지고 있는 경우가 많기에 중복 로그인들을 통제하지 못하면 아이템의 부정사용 등으로 이어질 수 있기에 일반적인 웹서비스에서 보다 민감하게 관리되어야 한다.

M10 - 취약한 바이너리 보호

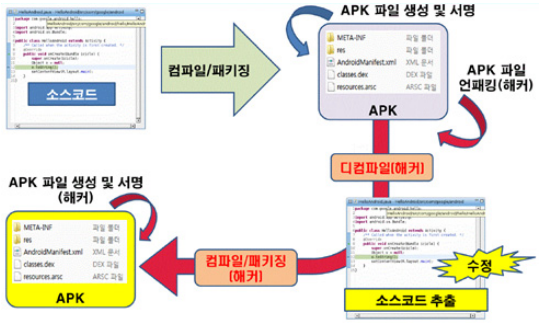
실행파일의 보호대책이 미흡하면 소스코드 유출이나 주요기능의 변조를 할 수 있다. 특히 안드로이드 OS의 경우 실행파일인 apk파일이 decompile 과 repackaging에 취약하여 불법적으로 조작된 실행파일(crack 된 실행파일)이 난무하고 있다.

아래 그림을 보면 안드로이드 실행파일의 경우 decompile이 용이하며 이마저도 자동화된 도구들이 있어 java 소스코드를 추출하기에 용이하다.



〔그림 2〕Android decompile 흐름도

추출된 소스코드를 분석하여 게임 내 주요 기능이나 수치를 조작하여 유리한 게임플레이를 가능하게 하는 실행파일을 재배포(repackaging)할 수 있기에 이는 게



〔그림 3〕 Android Repackaging 흐름도

임서비스에 영향만을 주는 것이 아니라 저작권도 위반하는 불법적인 용도로도 사용이 될 수 있다.

2.2 게임 서비스 내의 보안위협

- Desktop용 모바일 에뮬레이터의 사용

최근에는 안드로이드 모바일 app을 PC 환경에서 실행할 수 있는 여러 가지 에뮬레이션(emulation) 프로그램이 있어서 일반인도 손쉽게 pc에서 모바일용 게임을 설치하여 이용할 수 있다.

이러한 프로그램의 원래 용도는 안드로이드 테스트 환경의 제공 이였으나 게임이용자들이 pc에서 모바일 게임을 보다 손쉽게 이용하기 위해서도 사용되며 더욱 문제가 되는 건 기존 pc에서 사용되는 해킹툴을 이용하

〔그림 4〕 에뮬레이터를 이용하여 pc에서 안드로이드 환경을 실행한 화면

여 모바일 게임의 실행환경을 조작하여 게임을 이용할 수 있으며 이미 PC에서 사용 할 수 있는 여러 가지 모바일게임에 특화된 해킹툴들이 보급되고 있는 실정이다.

해외의 경우 무선인터넷환경이 취약한 지역이 있어 이러한 지역을 지원하기 위해 사업적으로 PC에서 에뮬레이터를 사용하는 것을 허용하는 경우도 있지만 보안적으로 보았을 때는 공격자가 훨씬 쾌적한 환경에서 게임에 해킹을 시도할 수 있기에 위협이 되는 요소이다.

- In App 결제 우회

PC나 모바일에서나 게임서비스의 주요 수익원은 게임아이템 판매이다.

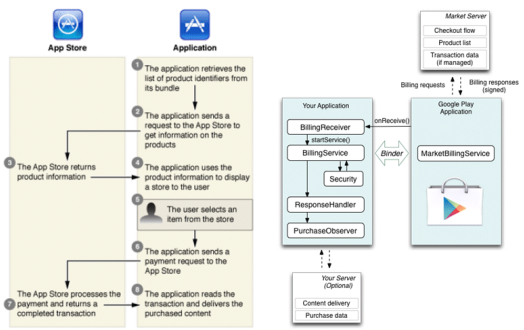
예전에는 서비스 이용료 형태나 패키지 구매비용으로 일정금액을 지불하고 게임을 이용하는 경우가 많았으나 온라인 게임이 되면서 게임서비스의 정액제 운영보다는 부분유료제 형태로 무료로 게임을 즐기면서 추가적인 혜택을 누리기 위해서는 별도의 비용을 지불하는 형태로 바뀌었다.

특히 모바일게임의 경우 게임 app을 유료로 구매하는 경우는 극히 드물며 대다수 무료로 배포 받아 게임 이용도중 유료 아이템 구매 형식으로 그 수익을 내고 있다.

PC 온라인 게임의 경우 서비스 제공자가 여러 가지 형태의 결제/지불 서비스를 연동할 수 있으나 모바일의 경우 IOS/Android 마켓에서 제공하는 결제방식을 채택해야 한다.

이는 app 배포 마켓에서 app 배포를 제공하고 결제 시 수익을 공유하는 방식이기에 이를 준수해야 한다.

만약 IOS/Android 마켓에서 제공하는 결제 프로세스에 취약성이 있다면 게임서비스에는 막대한 손실을 가져올 수 있다.



〔그림 5〕 IOS/Android In-App 결제 흐름도

〔그림 6〕 결제 우회 app 사례

위의 [그림 5]에서 알 수 있듯이 게임 app에서 결제 시 단말 내 마켓 app이 실행되면서 결제내역을 마켓서버와 통신하고 결제승인내역을 게임 app에 전달하는 방식이며 게임 app에서는 결제성공시 아이템 지급 등 다음 프로세스를 이어 나간다.

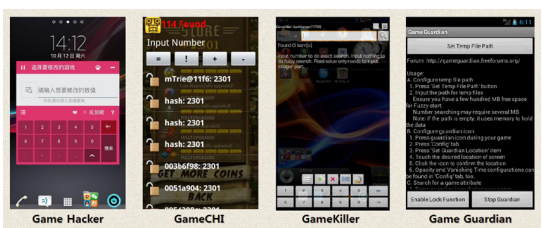
이러한 결제 프로세스를 우회하기 위한 app이 나와 있으며 버젓이 마켓에서 배포가 되고 있다

이와 같은 결제 우회 프로그램을 사용할 경우 단말 app에서 발생하는 결제내역을 서버를 거치지 않고 단말 내에서 결제 성공 정보를 생성하여 전달하여 주며 게임에서는 실제 결제가 이루어 지지 않았음에도 결제가 성공한 것으로 인식하고 아이템을 지급하게 된다.

IOS/Android 개발자 가이드에서는 이러한 결제우회를 막기 위하여 영수증 검증을 권고하고 있으나 실제 서버 개발자들이 이러한 부분을 간과하는 경우가 많다.

- 메모리 치팅 (memory cheat)

게임서비스도 단말 실행환경에서 메모리상에서 실행되는 구조이기에 프로세스의 메모리에 접근하여 게임 내 주요 수치나 코드를 조작하면 게임플레이를 쉽게 진행 할 수 있으며 게임서버에서 이러한 불법적인 조작을

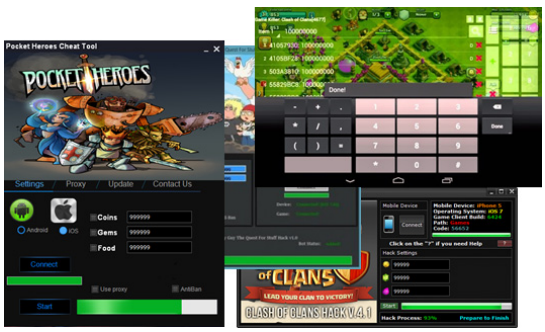


〔그림 7〕 모바일용 메모리 조작 app 사례

감지하기가 어렵다.

PC게임에서도 마찬가지지만 이러한 메모리 조작은 운영체제와 공격기술에 대해 해박한 지식을 알고 있어야 하는 고급기술이지만 최근에는 일반인들도 손쉽게 이용할 수 있는 해킹툴들이 나와서 기술적으로 이해도가 낮아도 게임을 손쉽게 해킹 할 수 있게 되었다.

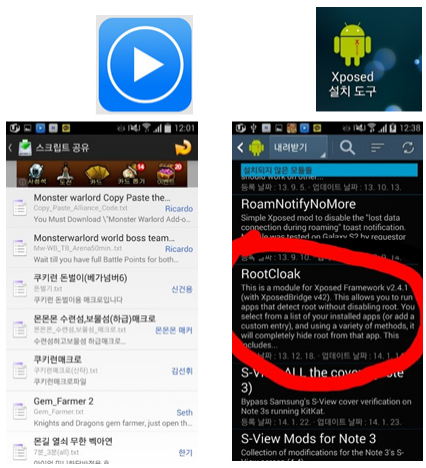
또한 최근 오랜 기간 동안 전 세계적으로 흥행하는 모바일 게임서비스들이 나오면서 특정 게임에 특화된 해킹툴이 출시되고 있는 추세이다.



〔그림 8〕 특정 모바일 게임 전용 해킹툴

- 자동플레이 (매크로)

PC게임시대부터 자동플레이는 게임보안에 큰 골칫거리였다. 많은 시간을 투자하여 게임플레이를 하는 유저의 입장에서는 자동플레이는 불공평한 것이었다. 일부의 경우 이러한 게임용 매크로만을 전문으로 제작 배포하는 유료사이트가 성행하며 게임서비스 업체보다 더 나은 수익모델을 가져가고 있기도 하였다.



〔그림 9〕 모바일용 매크로 프로그램 사례

모바일게임의 경우 자동플레이에 대해서 보다 관대하다. 모바일 환경 특성상 이동 중에도 게임을 할 수 있으며 세부적인 조작을 하기가 번거롭기 때문에 자동플레이를 게임 내에서 지원하는 경우도 있고 이러한 기능을 유료로 판매하는 경우도 있다.

하지만 최근 모바일용 매크로의 경우 사용자의 키조작을 모두 기억하고 그대로 재생해주는 형태로 제공되고 있으며 이러한 내용을 인터넷상에서 공유하는 형태로 발전해 나가고 있다.

Ⅲ. 보안성 점검항목

앞에서 알아본 여러 가지 해킹유형에 대해서 게임 배포 이전에 점검할 수 있는 점검항목을 도출하려 한다.

앞단원에서도 살펴보았지만 대다수의 게임해킹의 시도는 클라이언트 환경에서 이루어지며 모바일게임의 경우 단말환경의 취약성을 이용하는 경우도 매우 많다.

게임의 보안성 점검 시 가장 우선적으로는 취약한 단말환경이나 해킹시도를 감지했을 때 정상적으로 게임서비스가 종료되고 이러한 이벤트가 서버까지 잘 전달되느냐를 점검하는 것이다. 이때 사용자에게 관련 사항을 알려주는 것도 중요하지만 사용자에게 보여주는 화면이 또다시 취약한 공격의 요소로 이용되지는 말아야 한다.

〔그림 10〕 해킹탐지 시 게임종료 화면

3.1. 단말환경의 보안성 점검항목

- 취약한 단말환경 점검

모바일 게임 해킹의 경우 그 시작점이 탈옥이나 루팅인 경우가 많기에 이러한 환경에 대해서 게임실행 전 점검할 수 있어야 하며 이때 사업적 판단에 따라 서비스 진행여부를 검토할 수 있어야한다. 하지만 PC의 경우와 다르게 모바일의 경우 게임유저가 직접 루팅을 진행하는 경우가 더 많기에 유저의 클레임 우려가 있을 수 있으며 이에 대해서는 사전 운영부서와 검토가 선행되어야 한다.

- 알려진 해킹툴의 탐지

게임서비스 운영인력들도 지속적으로 게임 내 커뮤니티나 시장동향을 주시하며 최신 해킹툴에 대해서 모니터링 업무를 수행하고 있다. 이러한 경우 게임에 치명적 영향을 줄 수 있는 해킹툴을 탐지했을 때 이에 대한 정보를 게임서버에서 내려줘서 게임 app이 이러한 해킹툴의 사용이 탐지되면 게임서비스를 차단하는 기능을 확인해야 한다.

- 실행파일 위변조 탐지

안드로이드의 경우 특히 decompile & repackaging에 취약하기에 게임 실행파일이 조작되지 않았는지 검증할 수 있는 기능이 포함되어야 한다.

이는 실행파일 전체에 대한 무결성 검증도 필요하지만 게임 내 불특정 주요 로직에 대한 코드 검증을 수시로 한다면 그 실행환경의 신뢰성은 더욱 향상 될 수 있다.

- 난독화

안드로이드 게임의 경우 실행파일 난독화는 필수적이다. 최근에는 여러 보안 제품들이 binary wrapping 방식을 채택하고 있어 최종적으로 마켓에 배포하기 전에 한번만 작업하면 난독화를 반영할 수 있어 손쉽게 적용을 할 수 있다. 하지만 실행파일 난독화는 몇 가지 제약사항을 가지고 있다. 그중 가장 첫 번째가 호환성 이슈이다. 난독화 기술 특성상 실행파일의 동작방식을 변경하기에 모바일 백신 같은 기존의 보안솔루션에서 오탐 이슈가 있고 모든 단말환경에서 제대로 동작한다고 보장하기에 어렵다. 때문에 게임 내 신규버전 배포 시 가장 민감하게 테스트해야 하는 항목 중 하나가 난독화 후 정상실행여부이다. 행여나 OS 버전이라도 업데이트 되고나면 크게 이슈화가 되기도 한다.

최근 실행파일 난독화를 무력화 하는 해킹시도와 툴이 지속적으로 나오고 있으며 실행파일 난독화는 특성상 실행환경에서는 정상적인 코드 형태로 실행되어야 하기에 메모리 덤프 등에 취약할 수 있기에 최근에는 실행파일 난독화 보다 소스코드 난독화로 방향을 잡아가고 있다.

하지만 소스코드 난독화도 우려사항이 있기는 마찬

가지다. 소스코드 난독화의 경우 초기 개발 시부터 반영되어야 하기에 개발부서에서는 여러모로 부담이 되어 꺼리게 된다. 또한 일부 무료버전의 난독화 제품들은 일부 변수정도에만 난독화가 적용되기에 그 기대수준이 높지 않다.

난독화가 불독화는 아니기에 모든 해킹시도를 다 차단해주지는 못하지만 일반 유저가 단순 해킹시도를 하는 경우 그 첫 번째 장벽으로 역할을 하는 것이 난독화이기에 이 부분은 개발부서와 보안부서 모두 민감하게 고려해야 하는 항목이다.

- 게임 app 실행권한의 점검

최초 app을 설치할 때와 실행할 때 실행권한을 체크하게 된다. 게임에서 혹시라도 불필요한 시스템 자원을 요청하거나 민감한 정보를 수집하지 않나 점검해야 한다. 이는 단지 해킹 방지차원을 넘어서 개인정보보호법 등 관련 법규 준수를 위해서라도 사전 점검되어야 한다.

3.2. 게임 내 보안성 점검

- 게임 내 중요파일 암호화

게임사용자의 유저정보나 게임 플레이 정보(save file) 등은 사용자의 등록된 단말환경에서만 사용할 수 있게끔 특정키로 암호화 되어야 하며, 기타 게임 내 중요한 리소스 정보나 게임 내 데이터 등에 대해서는 암호화 되어 저장되어야 한다. 이때 암호화에 사용되는 키는 소스코드 차원에서 제공되지 말고 사용자와 단말에 종속적인 키를 생성하여 사용되어야 한다.

- 안전한 통신구간 점검

모바일 게임도 최근에는 온라인 게임화 추세로 가고 있으며 서버와 게임도중 지속적인 통신이 발생하고 있다. 모바일 환경 특성상 데이터 통신의 경우 성능과 데이터 사용량에 민감하여 암호화 통신을 안하거나 사용하여도 보안강도가 낮게 하는 경우가 많다. 이를 위해서 관련된 고려사항을 감안하여 안전한 통신 채널을 형성한다.

최근 web 환경으로 서버를 구성하는 사례가 많기에 https 등을 사용할 경우 보안설정이 취약하지 않은지 점검한다.

또한 보안통신의 경우 전송 데이터의 기밀성 뿐만 아니라 무결성 검증 기능도 포함되어야 하면 특히 게임서비스에서 취약한 재전송 공격에 대한 대응방안이 세워져 있어야 한다.

- 세션관리 점검

온라인 게임서비스의 특성상 계정도용 시도가 매우 많으며 그러한 공격에 대한 영향도 매우 심각하다. 모바일게임의 경우 PC용 게임과는 다르게 개인용 단말을 이용하여 게임을 이용하는 경우가 많기에 유저가 사용하는 단말과 계정을 연동하는 것도 좋은 방안이다.

또한 최근 유명 SNS와 계정을 연동하고, OAuth등 표준인증방식과 연동하는 경우 인증만료시간 설정에 민감해야 한다. 중복로그인 체크, 단말고유값 체크도 중요한 사항이다.

- 결제내역 서버검증 기능

모바일게임이 서비스되던 초기에는 결제우회를 통해 서비스에 큰 손실을 입는 경우를 많이 보았다. 그 결과 유명한 게임이 하루아침에 문을 닫아야 하는 상황까지도 가게 되었다.

또한 최근에는 결제금액 조작, 영수증 재사용, 결제 후 마켓에서 결제취소 등 다양한 형태의 결제우회를 시도하고 있다.

모바일환경 특성상 클라이언트 단말에서 결제가 발생한다고 해서 게임 app에서 그 결과를 바로 수용해서는 안되며 반드시 결제내역을 서버로 전송하여 서버에서 영수증 검증이나 결제내역을 검증하는 프로세스를 거쳐야 한다.

결제 항목 중 중요한 혜택이나 유료 아이템의 경우 클라이언트가 아닌 서버에서 사용내역을 관리하는 것도 좋은 방안이라 할 수 있다.

- 실행 중 중요 데이터 보호

실행 중 게임 내에서 사용되는 중요 데이터는 메모리상에 존재할 수 밖에 없다. 게임 내 모든 데이터를 암호화 하거나 보호하기에는 많은 시간과 비용이 발생하며 게임 성능에도 영향을 미치기에 중요 데이터를 분류하여 관리해야 한다.

중요 데이터에 대해서는 암호화하거나 분산 저장하는 방식으로 메모리 해킹툴에 대한 저항력을 가지고 있어야 하며 그럼에도 불구하고 클라이언트 데이터는 언

제든 조작될 수 있기에 주기적으로 서버와 데이터 현황을 동기화 하는 것도 좋은 방법이다.

모바일 게임 특성상 실시간 통신이 발생하는 게임이라도 전송 데이터를 최소화 하는 노력을 많이 하기에 플레이가 끝날 때 일괄 전송하는 경우도 많다 이러한 경우 그 데이터의 정합성을 검증할 수 있는 정보를 추가로 작성하여 서버로 전송하고 서버에서 기록된 데이터 값이 적절한지 여부를 검증해야 한다.

- 서버 모니터링 기능

사전적, 기술적 조치로 게임 내 이상현상을 모두 차단하기는 거의 불가능하다. 또한 강도 높은 보안기술의 적용은 게임의 정상 서비스를 방해하는 경우가 빈번하다. 그러기에 최신보안기술을 게임에 적용하지 못하는 경우가 발생할 수 있다.

이러한 이유를 포함해서라도 서버에서는 게임 내 이상현상에 대한 서버 모니터링 기능이 필요하다. 단순한 게임 내 이벤트의 로그를 남기는 수준이 아닌 특정한 유형의 이벤트가 반복되거나 확률적으로 발생하는 이벤트가 너무 빈번하게 발생하거나 한다면 게임 내 부정행위를 의심해야 한다.

특히 결제부정이나 아이템 중복 등은 유저들 사이에는 은밀히 공유되기에 그러한 현상을 모니터링 하기에 쉽지 않다. 요새는 게임서버 로그를 이용한 다양한 형태의 데이터 분석이 시도되고 전문 솔루션을 도입하는 경우도 있으며 그 효과도 기존 사전적 조치로 얻었던 것보다 더 큰 효과를 볼 수 있다.

게임서비스 초기에는 서버에서 데이터 분석을 하지 않는다 하더라도 향후 지속적인 서비스를 위해서는 서버 모니터링 기능이 꼭 필요하다 할 수 있다.

- 개인정보 관리

게임 내 사용자의 개인정보가 중요한 사항이 된지는 이미 오래이다. 개인정보보호법이 발휘되면서 법적으로 꼭 지켜야 하는 준수항목도 많이 생기게 되었다.

이에 게임 내에서 다루고 있는 유저의 개인정보는 출시 전 철저하게 점검되어야 한다. 특히나 약관내용과 수집범위가 적절한 지 등을 검토해야 하며 게임 내에서 수집된 정보가 어떻게 다루어지고 있는지도 점검되어야 한다. 특히 요즘에는 게임개발사와 게임을 서비스하는 주체가 다른 경우에는 개인정보 수집의 동의는 서비스하는 주체에게 하면서 사전에 동의되지 않은 개인정보를 개발사에서 임의로 수집하는 경우도 종종 발생한다.

게임이 단말 내에서 불필요한 개인정보를 수집하고 있지는 않은지, 사용자 정보 및 인증정보가 단말에 저장되는 경우 안전한 조치가 수행되는지, 전송 시 안전하게 전송되고 서버에서는 안전하게 관리되고 있는지, 또한 수집목적이 만료된 경우 적절히 폐기조치 되는지 등을 검토해야 한다.

개인정보의 경우 단순 게임 내 보안적 요소 뿐만 아니라 법규준수와도 연관되기에 중요한 점검항목으로 분류되어야 한다.

Ⅳ. 결 론

모바일게임 서비스의 경우 그 특성상 기존 PC용 온라인 게임과는 위협유형이나 보안조치가 차이가 많다.

모바일의 경우 환경적 제약이나 플랫폼의 취약성 등으로 인해 보안조치를 수행하기에 제약사항이 많다.

또한 아쉽게도 게임을 서비스하는 기업도 모바일게임의 짧은 라이프사이클 등으로 보안에 대한 인식이 부족한 것도 사실이다.

하지만 이미 여러 해 동안의 시행착오로 모바일게임에 대한 보안위협이나 보안기술에 대한 이해와 분석 경험을 축적하게 되었다.

이러한 경험을 바탕으로 보안 점검항목을 개선해 나간다면 보다 안전한 모바일게임 서비스를 할 수 있으며 사용자에게 제공할 수 있을 거라 생각된다.

참 고 문 헌

[1] 찰리 밀러, 디오니소스 발라자키스, 디노 다이 조비, 빈센조 이오조, 스테판 에서, 랄프필립 와인만, “iOS 해킹과 보안 가이드”, 에이콘 출판사, 2014

[2] 쉐런 구나세케라, "안드로이드 앱 보안", 길벗 출판사, 2013

[3] 키이스 마칸, 스콧 알렉산더바운, "안드로이드 해킹과 보안”, 에이콘 출판사, 2015

[4] 신준엽, “모바일 서비스 환경에서의 개발보안 감리방안 연구” 건국대학교 정보통신 대학원, 2012

[5] 이은진, “안드로이드 앱 보안 강화를 위한 프로토타입 구현”, 이화여자대학교 정보과학대학원,

2012

[6] 김휘강, 금영준, “안드로이드 환경에서의 모바일 게임 서비스 보안이슈”, 정보보호학회논문지, 23(2), 2013

[7] OWASP Secure Coding 규칙 ver2, OWASP, 2011

[8] OWASP Mobile Security Project - Top Ten Mobile Risks

[9] IOS Security Guide, https://www.apple.com/business/docs/iOS_Security_Guide.pdf

〈저자소개〉

강 선 명 (Sun Myung Kang)
종신회원

1997년 8월 : 중앙대학교 산업정보학과 학사
2016년 2월 : 동국대학교 국제정보보호대학원 정보보호학과 석사
2015년 10월~현재 : 펜타시큐리티시스템 근무
2014년 1월~2015년 3월 : 게임빌 정보보안 실장 근무
2010년 6월~2013년 11월 : 이니텍 개발본부장 근무
2007년 7월~2009년 8월 : JCE 보안기술 팀장 근무
<관심분야> PKI, 게임보안, 어플리케이션 위협 관리