情 報 保 護 學 會 誌
第21卷 第3號, 2011. 5

스마트기기 보호를 위한 스마트기기 보안취약점과 보안방안 검토

김 기 영*, 이 동 훈**

요 약

준비되지 않은 상황에서 스마트기기의 국내 도입은 비정상적인 형태의 보안을 초래하게 되었다. 그러나 이미 1년이 넘게 지난 상황에서도 크게 달라지지 않고 초기에 도입한 보안이 적용되고 있으나 현재 발생하고 있는 위협이나 예측되는 위협을 막기에는 크게 부족한 상황이다. 게다가 새로운 스마트기기들이 속속 등장하고 있기에 지금부터라도 스마트기기와 그 환경 특성을 고려한 적절한 보안 환경이 구축될 필요가 있다. 본 논문에서는 지금까지 도입된 스마트기기를 위한 보안 환경과 현재 발생하고 있거나 예측되는 위협을 검토하고 또한 이러한 스마트기기의 특성이 안전한 스마트기기 사용환경 구축을 위하여 어떻게 활용되어야 할지 검토하고자 한다.

Ⅰ. 서 론

2009년 11월 30일에 한국에 애플(Apple Inc.)의 아이폰(iPhone)이 소개된 이후 급속한 양적 팽창이 이루어졌다. 아이폰이 도입된 직후인 2009년 12월 10일 다양한 보안이 적용된 스마트폰 뱅킹이 시작되었다. 이때 적용된 보안들은 기본적으로 스마트폰을 PC와 동일한 것으로 간주하고 기존에 PC환경에서 금융서비스에 적용되었던 보안 요소들을 그대로 적용하였다.

그로 인하여 스마트폰의 사용은 매우 불편해졌다. 규제당국으로서는 기존에 PC환경에 적용된 규제보다는 훨씬 완화된 형태를 허용한 것이기는 하였으나 제약된 화면에서 적용하는 것은 보안에 아무런 도움이 되지 않는 것도 있었고 최근에 알려진 공격을 전혀 막지도 못한다.

이러한 결과는 스마트기기의 특성과 그것이 사용되는 행태에 대한 고려가 전혀 이루어지지 않은 상태에서 PC환경에 기존에 적용된 방식을 변형하여 적용하다 발생한 것이라고 할 수 있다.

이제는 스마트기기가 iOS를 기반으로 한 아이패드(iPad)와 갤럭시탭을 비롯한 안드로이드 운영체제를 탑재한 태블릿PC들 외에도 블랙베리의 플레이북이 있으며, 스마트폰도 iOS가 2011.4월 기준으로 4.3.2, 안드로이드도 3.0이 나온 상태이다. 게다가 윈도우폰도 이미 해외에는 판매되고 있으며 국내에도 조만간 도입될 예정이다. 그 외에도 리모(LIMO)[1]와 바다(Bada)[2] 등이 출현할 예정이다. 또한 이러한 운영체제(OS)들은 다양한 형태의 화면 크기를 지원하면서 PC와 다르기도 하지만 PC보다도 훨씬 많은 운영체제와 화면사이즈를 개별적으로 지원 해야만 하는 상황이다.

따라서 솔루션을 제공하는 입장에서도 다양한 운영체제에 다양한 버전을 지원하는 일이 더 이상 쉽지 않은 일이 됐으며 서비스를 구축하는 입장에서도 보안 솔루션이 지원되지 않아 서비스를 제공할 수 없는 상황이 발생할 수 있으며 이미 발생하고 있는 상황이다.

일반 웹 서핑만 하더라도 스마트기기에 문제를 초래할 수 있는 컨텐츠나 링크를 접근하는 것을 막거나 보호할 방법도 없는 상황이며 iOS환경에서는 백신도 지원할 수가 없다. 안드로이드의 경우 백신이 공급되고 있지만 운영체제의 특성상 그 기능이 제한적이며 dalvik

* 안철수연구소 전략제품개발실 (kiyoung.kim@ahnlab.com)
** 고려대학교 정보경영공학전문대학원 (donghlee@korea.ac.kr)

FreeType 'ft_var_readpackedpoints()' Buffer Overflow Vulnerability
FreeType TrueType Font Handling 'ttinterp.c' Remote Code Execution Vulnerability
libxml2 'XPATH' Memory Corruption Vulnerability
libpng Memory Corruption and Memory Leak Vulnerabilities
WebKit 'removeChild()' Remote Code Execution Vulnerability
WebKit CVE-2010-1783 Remote Memory Corruption Vulnerability
WebKit CVE-2011-0155 Unspecified Memory Corruption Vulnerability
WebKit CVE-2011-0142 Unspecified Memory Corruption Vulnerability
WebKit CVE-2011-0149 'HTMLBRElement' Style Memory Corruption Vulnerability
...
WebKit 'Runin' Box CVE-2011-0132 Use-After-Free Memory Corruption Vulnerability
WebKit CVE-2011-0129 Unspecified Memory Corruption Vulnerability
...
WebKit CVE-2011-0119 Unspecified Memory Corruption Vulnerability
WebKit Range Object Remote Code Execution Vulnerability
WebKit CVE-2011-0116 'setOuterText()' Method Memory Corruption Remote Code Execution Vulnerability
WebKit Resource Load Callback Information Disclosure Weakness
WebKit CSS 'format()' Arguments Memory Corruption Vulnerability
Apple iPhone and iPod touch Safari Referer Header Information Disclosure Vulnerability

〔그림 1〕 공개된 아이폰 취약점 목록

(Java 유사 기술기반의 미들웨어)[3]을 기반으로 어플리케이션이 동작을 하면서 멀트프로세싱(Multi-Processing)까지 지원하여 사용자의 체감 성능이 매우 나쁜 상태이다.

이에 다양한 클라이언트의 의존성을 가능한 낮추고 스마트기기의 가용성을 높이며 스마트기기의 특성에 따라 막기 힘들거나 불가능한 공격까지 막을 수 있는 검토방법을 제안하고자 한다.

그러기 위해서 II 본론에서는 스마트기기의 특성과 사용자 행동 패턴검토하고 현재까지 발견된 스마트기기 운영체제의 취약점과 악성코드에 대하여 알아봄으로써 스마트기기에 현실적으로 필요한 보안에 대하여 검토하고 스마트기기를 위한 보안을 제공할 때 추가적으로 고려해야할 사항을 검토한 후에 III. 제안 방안에서는 안전한 스마트기기 사용 환경을 구축하기 위해 필요한 보안 방식을 제안할 것이다. 그리고 끝으로 VI. 결론에서는 현실적인 범위 안에서 좀더 나은 보안을 제공하기 위해서 필요한 노력들을 검토함으로써 스마트기기 보호를 위한 보안 제공방안을 마무리 하도록 할 것이다.

Ⅱ. 본론

스마트기기에 적용되는 보안이 스마트기기에 가해지는 위협을 막을 수 없을 경우 불필요한 보안이 되며 스마트기기에 가해지는 위협을 막을 수는 있으나 성능이나 조작 편의성에 문제가 있으면 그 보안은 사용자들이 외면하게 되기 때문에 아무런 역할을 할 수 없게 된다. 그렇기 때문에 먼저 현실적이고 효과 있는 보안 방안을 제안하기 위해서 스마트기기에 발생하고 있거나 발생할 가능성이 높은 공격 유형을 중심으로 스마트기기에 대한 보안 위협을 검토할 것이다. 이러한 공격유형이 PC의 경우와 어떤 차이점이 있는지, 또 왜 발생하는지 비교할 것이다. 그리고 이러한 공격 유형을 분석하여 방어하기 위한 필요 요소를 도출하고 스마트기기의 특성에 대하여 고려한 뒤에 스마트기기에 가장 적절하다고 판단되는 보안에 대한 검토방법을 제안하도록 하겠다.

2.1 스마트기기에 대한 취약점과 공격

해외에서는 아이폰이 2007년 6월 29일에 출시되었기 때문에 아이폰에 대한 다양한 공격법이 나왔고 그 중 일부는 아이폰을 사용자가 보다 자유롭게 사용할 수 있는 탈옥(Jailbreak) 기술로 사용되기도 하였다. 최근에 알려진 아이폰의 취약점들은 [그림 1]과 같다. 우선 여기서 눈여겨 볼 것은 애플이 아이폰은 안전하다고 얘기하고 있지만 악성코드의 숫자는 그렇지 않음을 보여주고 있다는 것이다. 2010년 2월까지 알려진 개수는 37개에 지나지 않았었다. 그러나 2011년 3월 한 달 동안 등록된 취약점의 개수가 128개에 이른다. [그림 1]에는 2011년 3월 한 달 동안 SecurityFocus[4]에 등록된 아

Open Handset Alliance Android Web Browser Remote Information Disclosure Vulnerability
Libpng Library ICC Profile Chunk Off-By-One Denial of Service Vulnerability
Libpng Library Multiple Remote Denial of Service Vulnerabilities
Libpng Library Remote Denial of Service Vulnerability
The 'libpng' Graphics Library PNG_SET_SPLT Remote Denial of Service Vulnerability
Android Web Browser BMP File Integer Overflow Vulnerability
Android Web Browser GIF File Heap-Based Buffer Overflow Vulnerability
OpenCore 'pvmp3_huffman_parsing.cpp' Remote Buffer Underflow Vulnerability
Android Web Browser Unspecified Remote Code Execution Vulnerability

〔그림 2〕 안드로이드 취약점 목록

악성코드 안드로이드 앱, 공식마켓 확산

김희연 기자 hee@zdnet.co.kr 2011.03.04 PM 01:36

[지디넷코리아]구글은 지난 2일 공식 안드로이드 마켓에서 악성코드를 내장한 많은 개수의 앱에 대해 제거 조치를 취했다고 밝혔다.

이와 관련해 모바일 보안업체 룩아웃이 자체 조사한 결과에 따르면 'Myoumet', 'Kingmall2010', 'we20090202'라는 개발자들이 공식 마켓에 '드로이드드림'이란 악성코드를 넣은 앱 50개 이상을 올린 것으로 드러났다. 룩아웃은 사용자들이 자사 블로그에 공개된 감염 앱 리스트를 보고 확인하는 것이 좋다고 당부했다.

〔그림 3〕 안드로이드 악성앱 관련 기사[5]

이폰 관련 128개의 취약점 중 일부를 예시하였다. 이 수치는 시간이 지날수록, 사용자가 많아질수록 계속 증가할 것이다.

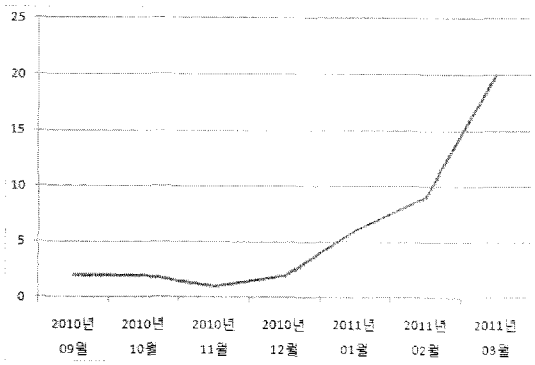
그러나 애플의 어플리케이션 정책에서는 안티바이러스 제품, 개인방화벽 등을 허용하지 않고 있으며 개발을 위해 사용할 수 있는 적절한 API도 SDK에는 공개되어 있지 않다. 이런 상황에서 애플이 완전무결한 운영체제를 만들거나 정책을 변경하여 개인방화벽이나 안티바이러스 제품을 만들 수 있도록 허용해줄 때까지 기다리는 것은 적절하지 못하다고 판단된다. 심지어는 그런 상황이 벌어진다고 해도 사용자가 운영체제를 업그레이드하거나 어플리케이션을 다운로드 받아야 하기 때문에 결과적으로 보안 개선에는 큰 기여를 하지 못하게 될 것이다.

안드로이드는 출현 시기가 늦고 본격적으로 시장에 퍼지기 시작한 시점이 2010년이어서 아직 그 숫자가 많지가 않다. [그림 2]에는 2011년 3월까지 발견된 안드로이드의 취약점들이 나열되어 있다. 안으로이드에서 발견되는 취약점이 적은 이유 중의 하나는 안드로이드는 사전검증 시스템이 없기 때문에 악성코드를 정상적인 어플리케이션의 형태로 만들거나 정상적인 어플리케이션의 패키징을 해제한 후에 악성코드를 포함하여 재패키징을 해서 배포하는 훨씬 쉽고 효과적인 방법이 있기 때문일 것이다.

이러한 추정을 뒷받침하는 것은 지금까지 발견된 취약점의 개수와 무관하게 유포되고 있는 악성코드의 개수를 보면 쉽게 확인할 수 있다. 아이폰의 경우 초기에 탈옥폰 대상으로 한 악성코드가 발견된 이후로 iTunes를 이용한 공격, 탈옥(Jail Break)툴을 가장한 공격 등이 발견되고 있으나 위협적인 악성코드가 발견되지 않고 있다. 이와 달리 안드로이드의 경우 2010년 미국의 주요 은행 어플리케이션으로 위장한 악성프로그램이 등장했었고 어플리케이션의 취약점을 이용하거나 다른 어플리케이션을 재패키징한 악성코드가 지속적으로 등장하여 현재 안드로이드용 백신에서 대응하고 있다.

안철수연구소에서 악성코드의 숫자는 2011년 3월까지 42개에 이른다. [그림 4]에서 보듯이 전체적인 숫자는 적지만 안드로이드 기반 기기가 급속히 확장됨에 따라 악성코드도 급속하게 늘고 있음을 알 수 있다.

더욱 큰 문제는 스마트기기의 사용 패턴에서 기인하는데 이미 널리 알려진 취약점에도 불구하고 대부분의 스마트폰 사용자는 취약점이 패치된 버전으로 업그레이드를 하지 않는 다는 것이다. 그 이유는 탈옥이나 루팅을 해서 사용하기 때문이기도 하고 또 많은 사용자가 업그레이드하는 방법을 모르는 경우가 많다.



〔그림 4〕 안드로이드 악성코드 증가 추이〔6〕

실제 최근 은행에 접근하는 아이폰 사용자의 아이폰 버전을 확인한 결과 특정 은행의 경우 전체 스마트폰 고객의 20%에 다다르는 9만명정도가 iOS 3.1.3을 사용하고 있어 이를 지원하기 위한 어려움을 겪고 있다고 한다. 2011년 4월 현재 iOS의 버전이 4.3.2인 것을 감안하면 이외의 다른 사용자들도 대부분 구매시점의 상태를 그대로 유지하고 있다고 짐작을 할 수 있으며 업그레이드를 자주 해주는 사용자의 경우에도 iOS가 업그레이드되어도 문제가 없다는 것이 검증된 후에야 설치를 하는 분위기여서 대부분의 아이폰이 최신 상태가 아니라는 것을 알 수 있다. 또 사용자들이 이동시에는 3G통신을 주로 사용하지만 장시간 정지 시에는 속도와 요금상의 이유로 WIFI를 즐겨 사용한다.

이러한 사용자들의 사용 패턴은 공격자에게는 매우 쉬운 공격 기회를 제공한다. 즉 취약점의 패치가 거의 이루어지지 않은 상태에서 외부에 노출되기 쉽다는 것이다.

이러한 스마트폰의 특성과 사용자의 행태가 어느 정도의 공격을 유발할 수 있는가를 짐작하기 위해서는 PC와의 비교검토가 이루어질 필요가 있다.

2.2 스마트기기와 PC환경 비교

공격은 특정 대상을 지목해서 일어날 수도 있지만 대부분의 악성코드들은 불특정 다수를 노린다. 이러한 불특정 다수를 공격하는 방법은 기존 PC에서 사용 되었던, 스팸, 브라우저 취약점, 운영체제 취약점 등을 활용하여 다양한 형태로 이루어진다. 당연한 것인지도 모르지만 공격은 대부분 공격 대상의 특성에 맞추어 발생하게 된다.

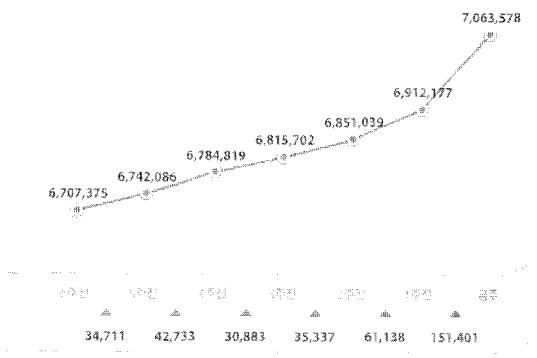
〔표 1〕 스마트기기와 PC 환경 비교

	이동성	계정 권한	전원 공급	OS 업데이트	주통신
PC/노트북	낮음	높음	어댑터	잦음	LAN/WIFI
스마트기기	높음	낮음	충전지	드믐	WIFI/3G

[표 1]에서 보면 사소하게 간주할 수 있는 차이점들이다. 그러나 이 사소하게 보이는 것들이 새로운 공격방식을 만들어 낸다. 먼저 이전의 PC환경에서는 대부분의 공격은 사용자의 방문과 행동을 요청을 했다. 그 이유는 PC가 공격 가능한 위치로 이동하는 것이 아니라 대부분 사용자가 공격 가능한 행동을 해주어야 하기 때문이다. 그러나 스마트기기의 경우는 이동성이 크고 노트북과 달리 이동 중에도 켜져 있어서 사용자의 정상적인 행동으로 정상적인 서비스를 이용하는 상황에서도 [그림 1], [그림 2]에서 본 수많은 취약점을 활용하여 업그레이드되지 않은 스마트기기의 WIFI 통신을 조작하여 다양한 공격을 시도해볼 수 있다. 스마트기기의 특성상 사용자가 의도적으로 PC의 iTunes와 같은 스마트기기 관리프로그램에 연동하여, 있을지도 모르는 운영체제의 업그레이드를 해주어야 한다. PC의 운영체제들의 경우 1년에 1회 정도 있는 업그레이드 보다는 수시 패치를 통해 취약점을 빠르게 없애고 있다. 이런 상황에서 취약점이 없어지기는 매우 어려운 상황이다. 따라서 위의 공격방식은 특히 대도시의 많은 사람이 모이는 곳이라면 효과가 클 수 있다.

예를 들어 통신사가 제공하는 무선 AP로 위장한 불법(rogue) AP[7]를 사용할 경우 스마트폰 사용자는 인식을 못하거나 정상적인 AP로 인식하고 사용을 하게 될 것이다. 그런 경우 그 AP를 경유해서 가는 보안이 적용되지 않은 모든 통신은 도청과 변조가 가능해져서 해당 통신을 발생시키는 어플리케이션의 취약점을 통하여 다양한 공격이 가능하게 된다. 일반적으로는 브라우저의 취약점을 이용한 공격이 가장 범용적인 공격이 될 것이다. 불법 AP를 사용하지 않더라도 현재 대부분의 무선 AP에 사용되는 보안 방식은 그에 대한 공격방식이 알려진 상태여서 무선 AP를 사용하는 상황에서 암호화 되지 않은 통신에 대해서는 보호가 불가능한 상황이 된다.

PC에서와는 다르게 스마트기기에서는 화면과 입력의 제약으로 불특정 웹서핑을 많이 하지는 않는다. 결국 악



〔그림 5〕 PC에서의 악성코드 진단 건수[8]

성코드를 포함하고 있는 웹페이지를 접근할 확률이 그만큼 낮아진다는 것이다. 따라서 WIFI를 많이 사용하는 스마트기기의 특성상 위와 같이 중간자공격(MITM, Man in the Middle Attack)에 노출되기가 쉽게 되어 동일한 취약점을 공격자가 활용할 수 있게 된다.

이미 널리 알려진 이야기이지만 어떤 방식으로든 스마트기기를 공격자가 제어할 수 있는 상태가 되면 정보유출이나 삭제, 배터리 고갈, DDoS 공격 등 다양한 피해로 확산될 수가 있게 된다. 특히 스마트기기는 PC와 달리 24시간 켜져 있기 때문에 DDoS 공격에 이용될 경우 그 피해는 지금보다도 막대하게 된다.

사실 위와 같은 공격은 PC에서도 발생할 수 있다. 그런데 PC에서 이러한 공격을 막을 수 있는 이유는 PC에서는 다양한 공격이 용이한 만큼 백신도 PC의 대부분의 자원에 접근하여 방어가 가능하였기 때문이다. 문제는 동일한 공격에 대하여 스마트기기에서는 막기가 어렵다는 것이다. 일반적인 어플리케이션의 형태를 하고 있는 악성코드의 경우에는 대부분 막을 수 있으나 시스템이나 어플리케이션의 취약점을 공격하는 경우에는 권한문제 즉 악성코드는 시스템 권한을 확보하지만 안티바이러스는 정상적인 어플리케이션이기 때문에 시스템 권한을 확보하지 못해 악성코드를 대응해야 하기 때문에 충분한 방어를 제공하지 못하게 된다.

[그림 5]는 2011년 4월 18일 기준 안철수연구소의 V3에서 진단하는 악성코드의 수를 나타낸다. 이 그래프를 통하여 악성코드가 지속적으로 증가하는 것처럼 그에 대한 방어도 꾸준히 이루어지고 있음을 알 수 있다. 그러나 스마트기기에서는 운영체제나 어플리케이션의 취약점을 이용하여 발생하는 공격에 대해서는 PC와 달리 위와 같이 안티바이러스 제품이 진단해 낼 수 없다는 문제점이 발생하게 된다. 게다가 iOS환경에서는 안티바이러스 제품조차도 존재할 수 없는 환경이다.

2.3 스마트기기에 대한 공격 분석

이러한 상황을 극복하기 위해서는 먼저 스마트기기에 발생할 수 있는 공격 위치와 방식 등에 대해서 검토해볼 필요가 있다. 즉 공격으로 인해 발생할 수 있는 일보다도 어떤 방식으로 어떻게 공격이 가능한가를 먼저 점검해야 한다. 그리고 그러한 공격을 방어할 수 있는 요소 기술들을 검토해야 한다.

[표 2]처럼 공격이 일어나는 위치와 유형을 정리해보면 의외로 전제적인 대응 방법이 단순해짐을 느낄 수 있다. [표 2]의 위협에 대한 방어에는 몇 가지 전제가 있다.

- 기존에 발생하던 웹 사이트에 대한 공격은 지속적으로 발생하며 현재도 그렇지만 앞으로도 쉽게 해결되지 않을 것이다.
- 애플은 악성 어플리케이션이 앱스토어에 등록되지 않도록 관리를 지속적으로 잘 해나갈 것이다.
- 두 운영체제 모두 샌드박스 개념과 어플리케이션의 권한이 크게 달라지지 않을 것이다.

〔표 2〕 스마트기기에 대한 위협 및 방어

	iOS	안드로이드
분실	파기 / 잠금	파기 / 잠금
악성코드 / 악성 URL	사전차단	사전차단 / 백신
악성어플리케이션	사전검열	백신
도청/MITM(불법 AP / AP 해킹)	단대단 암호	단대단 암호

이런 전제로 [표 2]의 내용을 항목별로 분석을 하겠다.

먼저 분실에 대해서는 지금까지 언떤 스마트기기도 안전한 것이 없다. 그래서 나온 것이 애플의 Mobile Me와 같은 서비스로 원격에서 삭제/파기를 하는 방법이다.[9] 이런 서비스 이용 시 기본적으로 스마트기기의 개인정보와 일부 컨텐츠가 동기화 되어 백업되어 있기 때문에 삭제를 하여도 사용자에게 큰 피해는 없게 되고 기밀이나 개인정보가 유출되는 것을 막을 수 있다. 이 부분은 이미 대안들이 나와 있기 때문에 꼭 필요한 기능이지만 더 이상 다루지는 않겠다.[10]

다음으로는 운영체제나 어플리케이션의 취약점을 노리는 악성코드나 악성 URL이다. 이미 앞에서 검토를 했듯이 PC에서와 다르게 스마트기기의 특성으로 인해 시스템 권한을 확보한 악성코드를 제한된 권한을 가진 백신으로 대응을 해야 한다. 따라서 그 대응은 할당된 권한만큼이나 제한적일 수밖에 없다. 따라서 가능하면 스마트기기에 진입하기 전에 차단을 해야만 한다는 결론이 나온다.

사전검증이 없는 안드로이드에서 주로 발생하는 악성 어플리케이션의 경우는 사전에 차단이 가능할 수도 있겠지만 현재 대부분 백신과 같은 방식으로 차단을 하고 있고 구글에서 원격 삭제를 해주는 방법이 있을 수 있다.

끝으로 주로 무선 AP를 경유할 때 발생할 수 있는 도청이나 중간자 공격(MITM)의 경우이다. 무선 AP에 암호화가 설정되어 있을 수도 있지만 그렇지 않은 경우에는 무선 AP를 공격하지 않고도 그 내용에 대한 도청이 가능하게 된다. 그리고 앞에서 언급을 했듯이 이미 표준으로 나온 무선 AP의 보안 통신은 양단이 디지털인증서를 사용하지 않는 이상 대부분의 경우 공격이 가능하기 때문에 해당 무선 AP를 경유하는 모든 통신은 암호화 되어 있을 필요가 있다. 문제는 일반 웹페이지에 HTTPS와 같은 보안이 적용되어 있지 않기 때문에 이에 대한 해결 방법이 필요하다. 또한 최근에는 무선환경에서 HTTPS에 SSL Strip이나 서버를 가장하는 중간자 공격까지 진행이 되고 있어서 채널에 대한 보안 강화가 절실하다고 할 수 있다.

2.4 추가적인 고려사항

스마트기기를 사용하면서 느끼는 가장 중요한 것 중의 하나가 바로 기기의 반응성이다. 응답속도로도 표현될 수도 있는데 이것이 단순히 CPU성능을 의미하지 않는다는 것이다. iOS야 기술적으로나 정책적으로 두 개 이상의 어플리케이션이 동시에 실행되는 것을 제안함으로써 실현을 하고 있는데 안드로이드의 경우는 사용자가 현재 사용하지 않는 어플리케에션도 서버로부터 오는 메시지를 받기 위해 해당 어플리게이션이 모두 동작하고 있으며 두 개의 어플리케이션을 같이 실행도 할 수 있어 백신이 동작할 수 있는 기반을 제공해 주기도 하지만 동시에 스마트기기의 응답성을 떨어뜨리고 배터리 소모 속도를 가중시키게 된다.

백신이 스마트기기에서 적극적으로 활동하여 다양한 보안을 제공해 줄 수도 있지만 이는 스마트기기 사용자의 만족도를 떨어뜨리는 역할을 하게 될 가능성이 높게 된다.

다른 측면으로는 스마트기기 기반으로 다양한 서비스를 제공하려는 사업자들이 있는데 그 구현 방법은 웹일 수도 있고 어플리케이션일 수도 있다. 문제는 보안통신을 위해 웹과 어플리케이션에 일일이 보안모듈을 적용하고 백신 동작여부를 점검하기는 어렵다는 것이다. 결국 보안이 비즈니스적인 측면에서 도움이 되는 방향으로 작용을 해야지 이를 방해하는 방향으로 작용한다면 좋은 방안이라고 할 수 없을 것이다.

Ⅲ. 보안 구축시 고려 사항

지금까지 본론에서 스마트기기에 발생할 수 있는 공격 유형과 방어요소, 그리고 방어를 하기위해 고려해야 할 사항들을 검토하였다. 여기서는 그러한 상황과 조건에 적합한 보안에 대한 검토방법을 제안하도록 하겠다.

우선 최대한 스마트기기에 설치되어 보안을 위해 스마트기기의 사용성을 크게 떨어뜨리는 방식은 최소화할 필요가 있다. 그리고 가능하면 스마트기기의 종류에 따른 제약이 가장 적은 방법을 선택 해야 하며, 끝으로 가장 효율적인 방법을 선택 하여야 한다.

또한 고려 되어야 할 내용은 PC와 같이 보안 프로그램이 많은 권한을 가지고 있지 못하므로 가능하면 스마트기기에서 악성 코드를 차단하는 방식이 아닌 스마트기기 내부로 악성코드가 들어오기 전에 차단함으로써 기존의 안티바이러스 제품의 제약을 극복할 필요가 있다. 다만 외부에서 차단하지 못하거나 어려운 컨텐츠에 대해서는 스마트기기 안에서 대응하는 방식을 취함으로써 스마트기기의 사용성 훼손이 최소가 되도록 하므로

써 스마트기기로 제공되는 다양한 서비스들이 보안을 특별히 고민하지 않고도 사용자가 안전하게 서비스를 이용할 수 있도록 하여야 한다. 그렇지 않을 경우 스마트기기를 사용하는 근본적인 이유나 목적이 사라질 수 있기 때문이다.

Ⅳ. 결 론

지금까지 다루어진 방식이 모든 공격방식을 다룬 것은 아니다. 앞에 제시된 고려사항만 반영을 하여도 초기에는 대부분의 공격이 차단될 것으로 예상된다. 그러나 일반적으로 사용되는 채널이 모두 차단되었다고 생각할 경우 사용할 수 있는 방식은 많은 사람이 사용하면서 보안을 적용하기 어렵거나 취약점이 존재하는 곳을 찾아낼 것으로 예상 된다. 가장 유력한 방식중의 하나는 메일의 첨부파일을 통한 악성코드 전달이다.

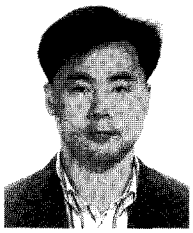
한 가지 방식으로 모든 문제를 해결할 수는 없겠지만 사용자가 불편함을 느끼지 않으면서 보안을 제공한다는 기본 개념을 바탕으로 스마트기기에 적합한 보안 방식을 취사선택하여야 한다.

또한 스마트기기를 위한 보안을 적용하기 위해서는 스마트폰뱅킹에 적용된 사례처럼 기존 PC에서 사용하던 방식을 그대로 사용할 것이 아니라 스마트기기에 적합한 방식을 고민해서 적용할 것을 권고한다.

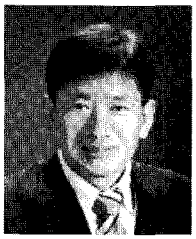
참고문헌

[1] http://www.limofoundation.org/en/what-is-the-platform.html
[2] http://www.bada.com/
[3] http://code.google.com/p/dalvik/
[4] http://www.securityfocus.com/bid vendor : Apple, Title : iPhone
[5] http://www.zdnet.co.kr/news/news_view.asp?artice_id=20110304102951
[6] 안철수연구소 플랫폼개발실 2011.4 통계
[7] http://www.freepatentsonline.com/7181 530.html
[8] http://www.ahnlab.com/kr/site/securtycenter/statics/popStatistics.do? svccode=aa1001&content
[9] http://www.apple.com/mobileme/ http://www.me.com/
[10] S. Ryan, C. J. Kolodgy and S. D. Drake, "World wide Mobile Security 2010 - 2014 Forecast and Analysis," IDC #222348, Volume:1, Mar. 2010.

〈著者紹介〉



김 기 영 (Kiyoung Kim)
종신회원
1997년 2월 : 한양대학교 전자공학과 졸업
2009년 8월 : 고려대학교 정보보호대학원 석사과정 수료
2011년 1월~현재 : 안철수연구소 전략개발실장
관심분야 : 정보보호, 데이터베이스, PKI, 클라우드 컴퓨팅, 가상화, 스마트그리드 보안



이 동 훈 (Dong Hoon Lee)
종신회원
1983년 8월: 고려대학교 경제학사 졸업
1987년 12월: Oklahoma University 전산학과 석사 졸업
1992년 5월: Oklahoma University 전산학과 박사 졸업
1993년 3월 ~ 1997년 2월: 고려대학교 전산학과 조교수
1997년 3월 ~ 2001년 2월: 고려대학교 전산학과 부교수
2001년 3월 ~ 현재: 고려대학교 정보경영공학전문대학원 교수
관심분야 : 암호프로토콜, 암호이론, USN이론, 키 교환, 익명성 연구, PET 기술