

Smart Phone Hacking!

(2) Remote Attack

정구홍@BoB

2013-08-30

강의 내용

- 원격 공격(Remote Attack)이란?
 - Remote Attack Vectors
- 실습 환경 구축
 - Android SDK, ADT
- 공격 예제
 - Webkit Remote Exploit
 - ~~Flash Remote Exploit~~
- 스마트폰 Debugging

원격 공격(Remote Attack)이란?

- 대상 시스템의 접근 권한을 획득하는 공격
- Active Attack
 - 공격 당하는 사람의 Action 없이 바로 공격
 - 열려 있는 포트 이용
- Passive Attack (Drive-by-download)
 - 공격 당하는 사람의 Action이 필요한 경우
 - 이메일 클릭
 - 문서 파일 오픈
 - 악성코드 삽입 홈페이지 URL 접속

Remote Attack Vectors

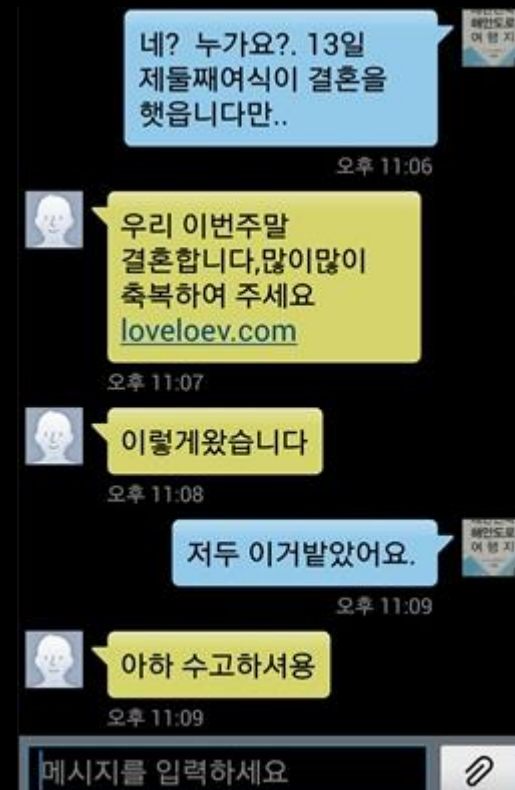
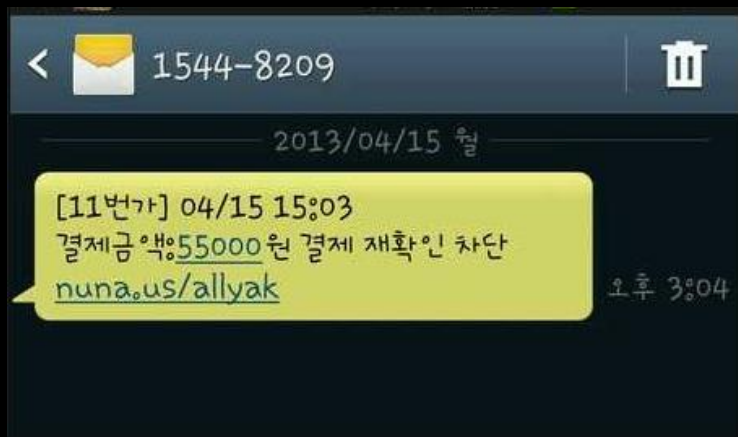
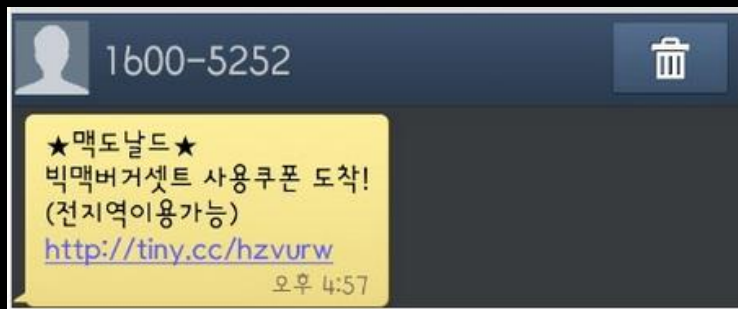
Remote Attack Vectors

- 악성 앱 설치 유도
 - 스미싱(Smishing)
- 웹 브라우저 공격
 - 브라우저 엔진 공격 (webkit)
 - Flash module 공격
- 문서, 멀티미디어 프로그램 공격
 - 사진, 그림, 동영상, 음악, 문서
- 원격 서비스/앱 공격
 - Remote TCP Daemon
- 기타
 - NFC, 블루투스, 3G/4G/LTE, 원격 앱설치(아이폰 아이튠즈)

악성 앱 설치 유도

- 스미싱

- 낚시를 통해 악성 앱 설치 유도



악성 앱 설치 유도

- 악성 앱으로 인한 피해
 - 모바일 결제
 - DDOS Agent
 - 스파이 프로그램
 - SDK의 기본 기능 이용
 - 문자, 사진 데이터 접근
 - 음성 녹음 기능 이용
 - 위치 정보
 - 백그라운드 실행

음성, 영상 처리는 스마트폰의 기본 기능이다



웹 브라우저 공격

- 브라우저 엔진 공격
 - Webkit
 - <http://www.webkit.org>
 - 오픈 소스 기반의 웹 브라우저 엔진
 - Webkit을 사용하는 대표적인 브라우저들
 - 안드로이드, 아이폰(safari), 크롬, Safari(pc용), ...
- 브라우저 모듈 공격
 - Adobe Flash Player
 - Adobe AIR
 - JAVA

웹 브라우저 취약점 예제들

Vulnerability statistics								20/46
• Porting existing win32 exploit to smart phone								
Type	CVE number	win32	Eclair	Froyo	Gingerbread	Honeycomb	ICS	iPhone 3g
Webkit	CVE-2010-1119	O	O	X	X	X	X	O
Webkit	CVE-2010-1807	O	O	X	X	X	X	X
Webkit	CVE-2010-1813	O	O	O	X	X	X	O
Webkit	CVE-2010-1759	O	O	O	X	X	X	X
Adobe	CVE-2011-0611	O	O	O	O	X	X	-
Adobe	CVE-2010-3654	O	O	O	X	X	X	-
Adobe	CVE-2010-1297	O	?	X	X	X	X	-
Adobe	CVE-2011-0609	O	?	X	X	X	X	-
Adobe	CVE-2012-0754	O	X	X	X	X	X	-
Adobe	CVE-2011-2140	O	O	O	O	X	X	-
Adobe	CVE-2010-3653	O	X	X	X	X	X	-
Adobe	CVE-2009-1862	O	X	X	X	X	X	-



출처 : x82님 발표자료

http://research.hackersschool.org/bbs/data/smartphone_hack/x82_android.zip

문서, 멀티미디어 프로그램 공격

- 문서
 - PDF
 - DOC
 - PPT
 - ...
- 멀티미디어
 - JPG
 - PNG
 - MP3
 - ...

원격 서비스/앱 공격

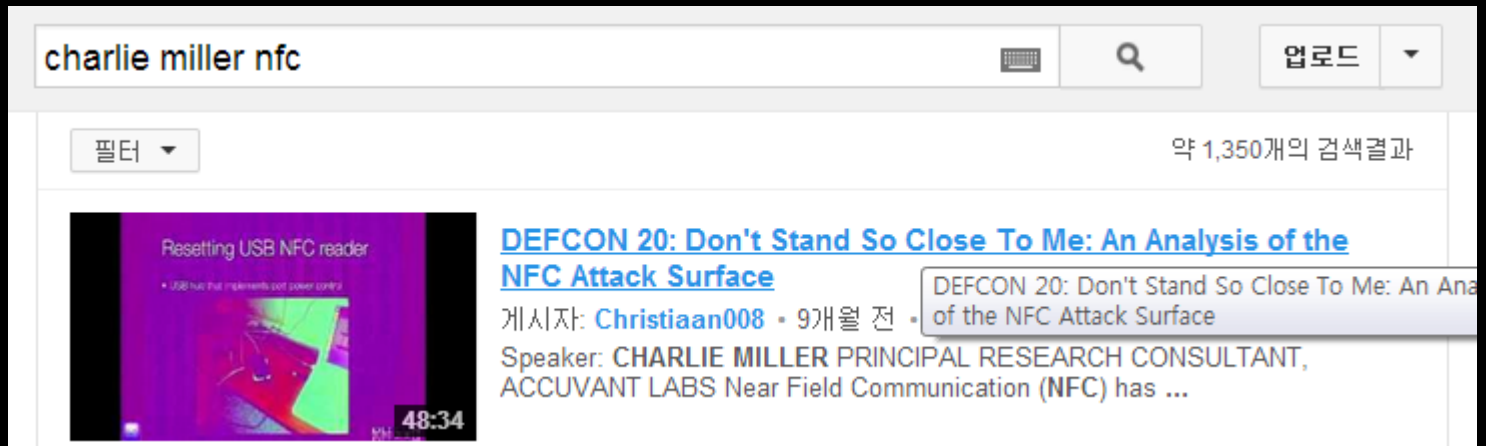
- SSHDroid(TCP 22)
 - Use-after-free, BufferOverflow 등
 - <http://www.cvedetails.com/cve/CVE-2012-0920/>
 - <http://exploitsdownload.com/search/dropbear/>
- FTP(TCP 21)
 - DoS
 - <http://www.exploit-db.com/exploits/18630/>
- 그 외 포트를 여는 모든 서비스/앱들

기타

- 기타
 - NFC
 - DEFCON 20: Don't Stand So Close To Me
 - <http://www.youtube.com/watch?v=bpiuKEy0SkY>
 - 블루투스
 - http://www.youtube.com/results?search_query=BLUETOOTH+HACKING
 - 3G/4G/LTE
 - http://www.youtube.com/results?search_query=3G+HACKING
 - 원격 앱 설치
 - 아이폰 아이튠즈를 이용한 원격 설치

Charlie Miller's NFC attack

- 유튜브 검색



시연 동영상 : 5분 50초부터 시작

실습 환경 구축

우리에게 필요한 건?

- 실습환경
 - Android Emulator



실습 환경 구축

- Android SDK란?
 - Software Development Kit
 - 안드로이드 개발을 위한 API 라이브러리 + 컴파일, 테스트 및 디버깅 환경 제공
 - Android 가상 에뮬레이터 제공 (QEMU 기반)
- ADT란?
 - Android Developer Tools
 - Android SDK + Eclipse + ETC... 통합 버전
 - <http://developer.android.com/sdk/index.html>
 - JRE 안 깔려 있는 분 설치 필요
 - <http://www.oracle.com/technetwork/java/javase/downloads/jre7-downloads-1880261.html>

ADT 설치

Get the Android SDK

The Android SDK provides you the API libraries and developer tools necessary to build, test, and debug apps for Android.

If you're a new Android developer, we recommend you download the ADT Bundle to quickly start developing apps. It includes the essential Android SDK components and a version of the Eclipse IDE with built-in **ADT (Android Developer Tools)** to streamline your Android app development.

With a single download, the ADT Bundle includes everything you need to begin developing apps:

- Eclipse + ADT plugin
- Android SDK Tools
- Android Platform-tools
- The latest Android platform
- Android Platform-tools
- Android SDK Tools
- Eclipse + ADT plugin



Download the SDK
ADT Bundle for Windows

ADT 설치

SDK
NDK

2.1 In order to use the SDK, you must first agree to this License Agreement. You may not use the SDK if you do not accept this License Agreement.

2.2 By clicking to accept, you hereby agree to the terms of this License Agreement.

2.3 You may not use the SDK and may not accept the License Agreement if you are a person barred from receiving the SDK under the laws of the United States or other countries including the country in which you are resident or from which you use the SDK.

2.4 If you are agreeing to be bound by this License Agreement on behalf of your employer or other entity, you represent and warrant that you have full legal authority to bind your employer or such entity to this License Agreement. If you do not have the requisite authority, you may not accept the License Agreement or use the SDK on behalf of your employer or other entity.

☒ I have read and agree with the above terms and conditions

☒ 32-bit ☐ 64-bit

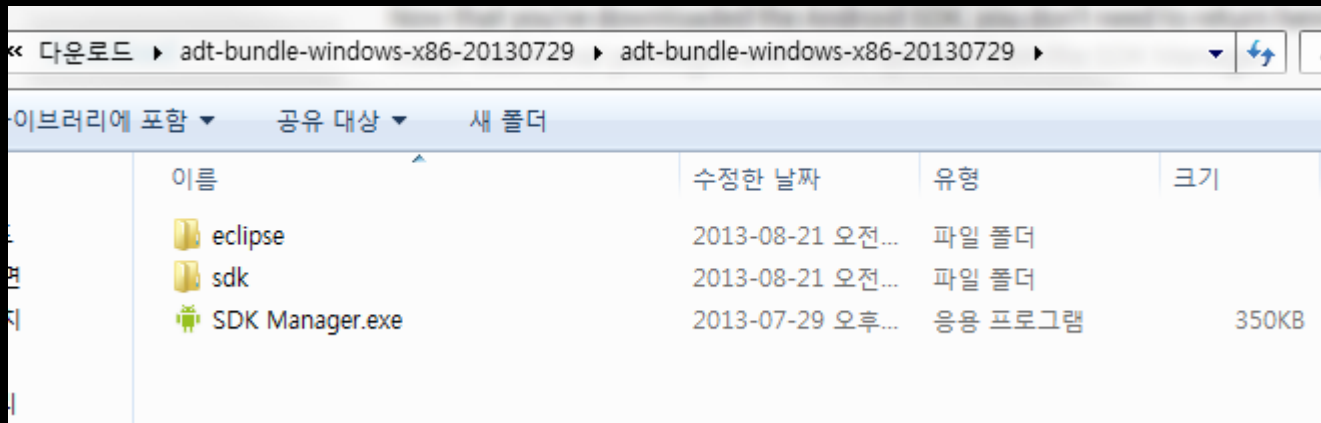
Download the SDK ADT Bundle for Windows

Download the SDK ADT Bundle for Windows

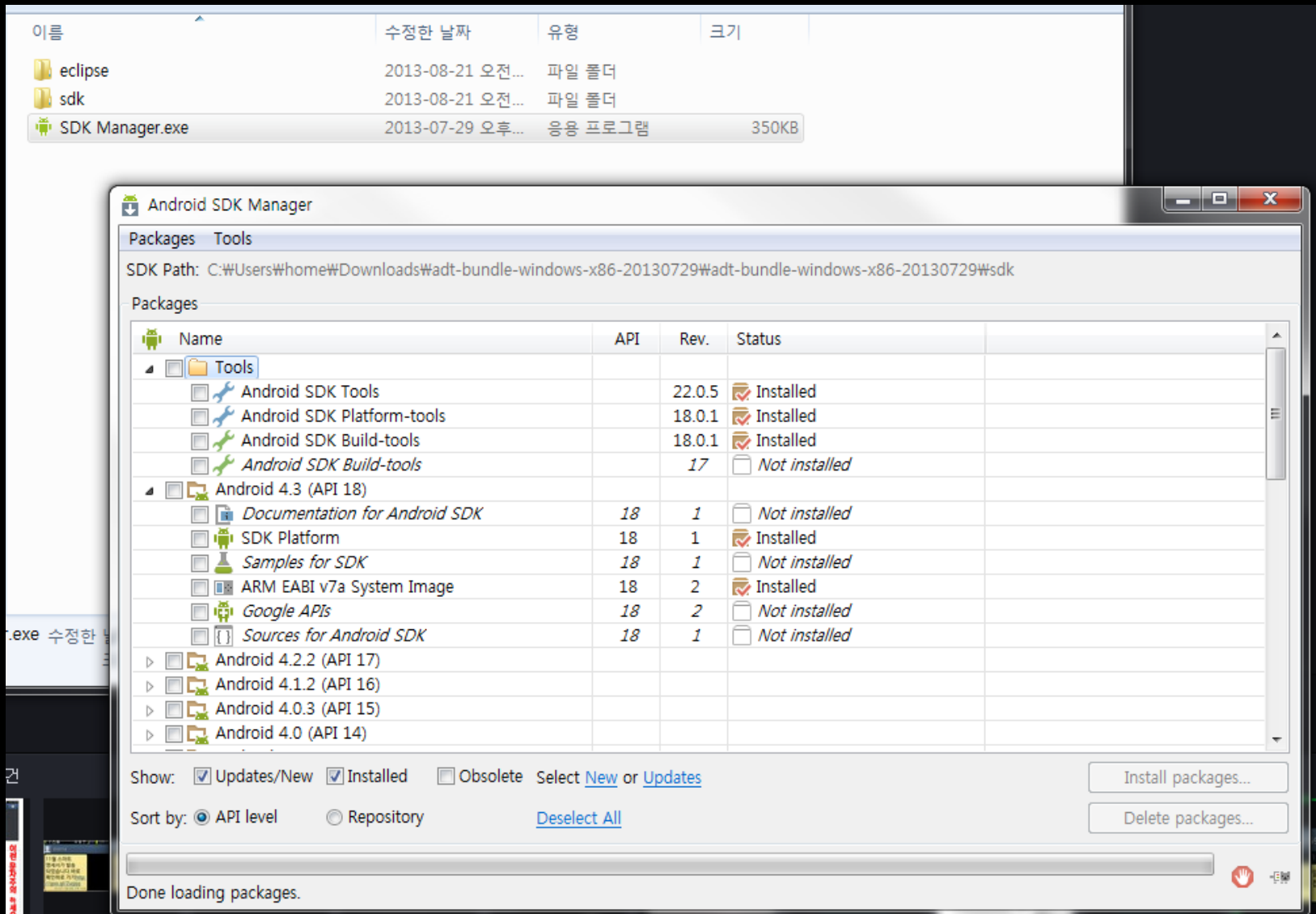
32-bit 64-bit

ADT 설치

- 압축 해제 후 나오는 파일들
 - Eclipse
 - 통합 개발 환경
 - SDK
 - 개발 라이브러리 + 디버깅 환경
- SDK Manager.exe : 업데이트/추가설치

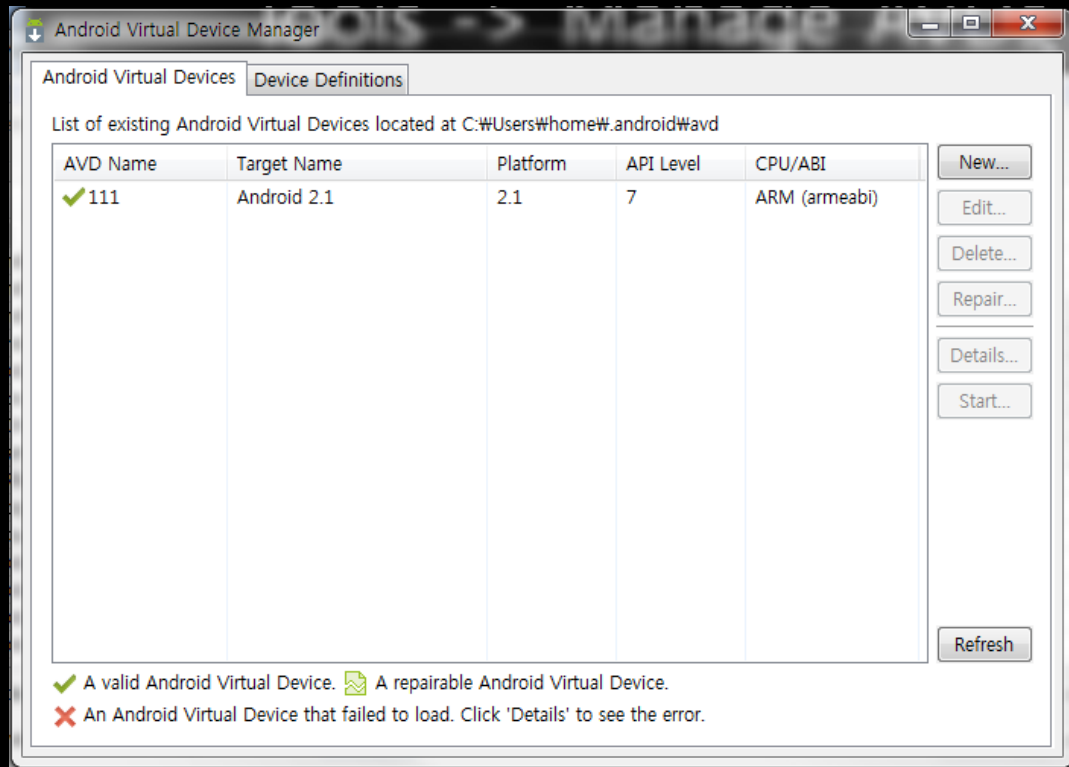
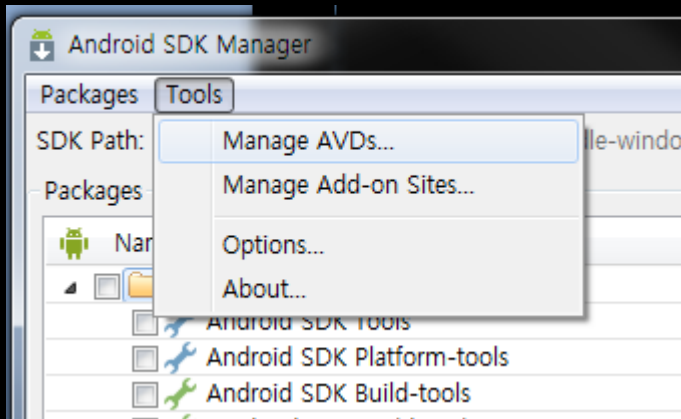


SDK Manager 실행



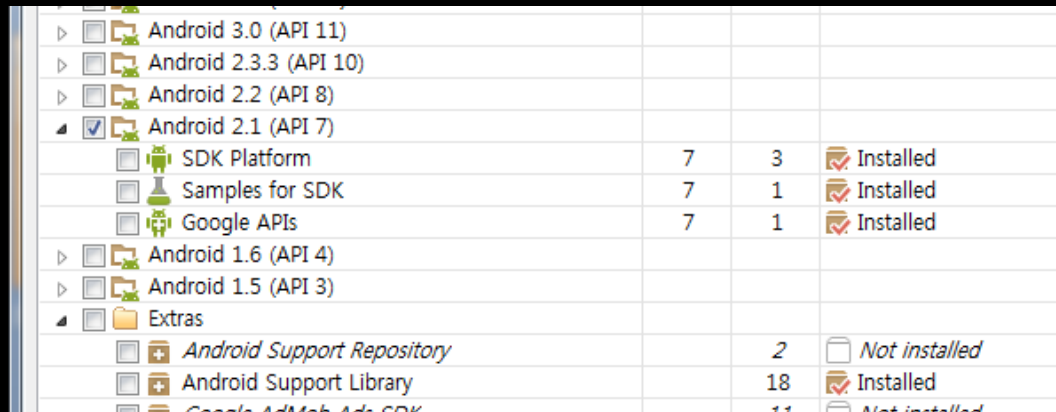
Tools -> Manage AVDs

- AVD = Android Virtual Device



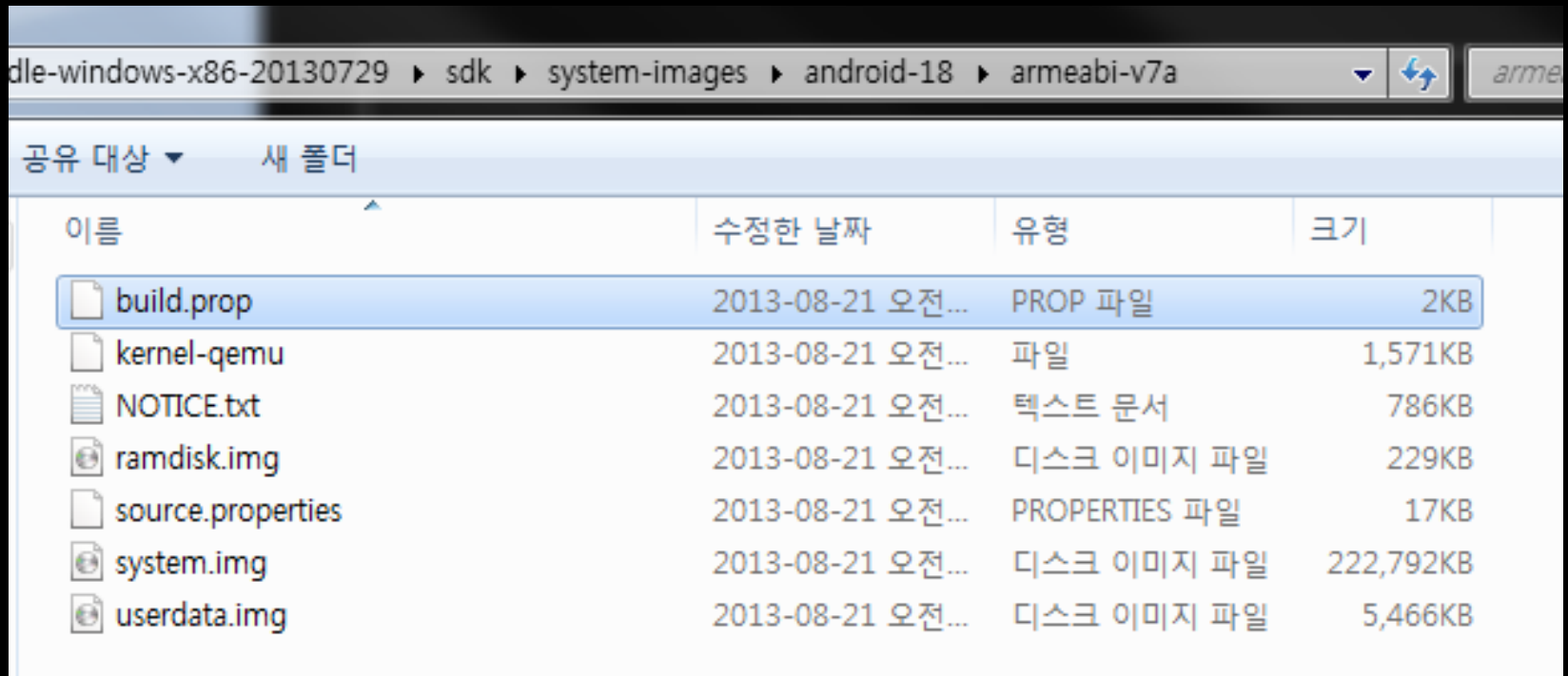
실습용 QEMU 이미지 추가하기

- 기본 설치 이미지
 - 안드로이드 4.3 (젤리빈)
- 실습에 필요한 이미지 추가 설치
 - 안드로이드 2.1 (Éclair)
 - 다시 SDK Manager로 돌아가서..
 - Android 2.1 체크 후 우측 하단의 "Install"



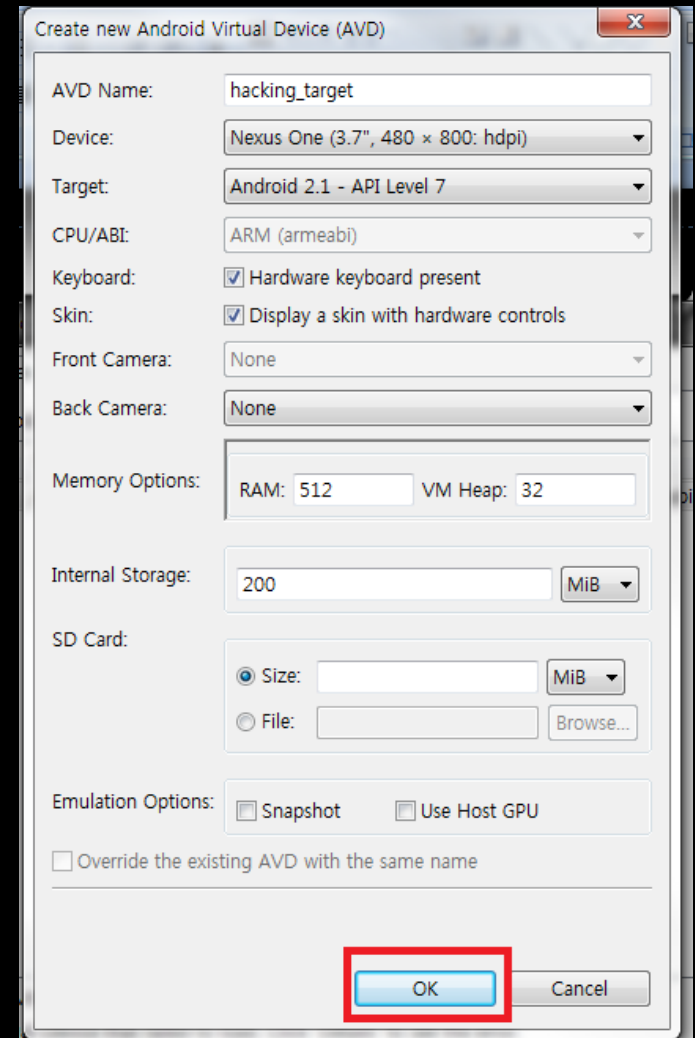
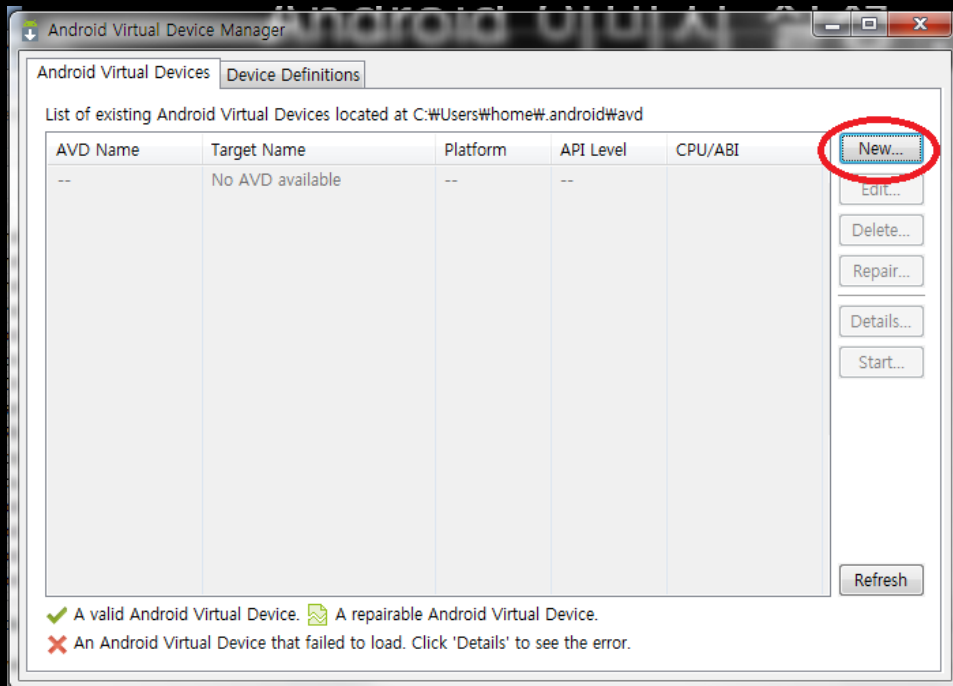
QEMU 이미지 파일의 구성

- SDK\system-images\ 폴더

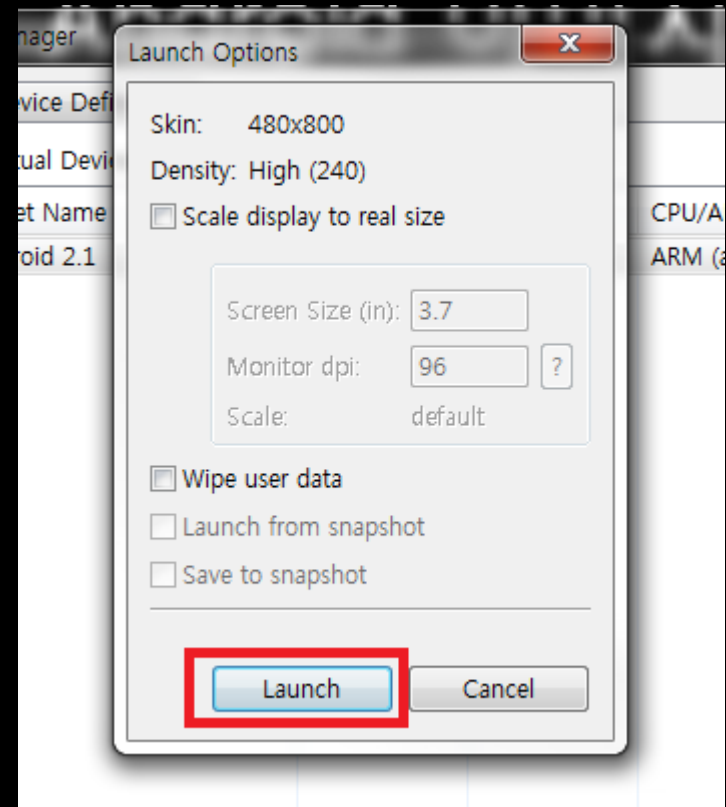
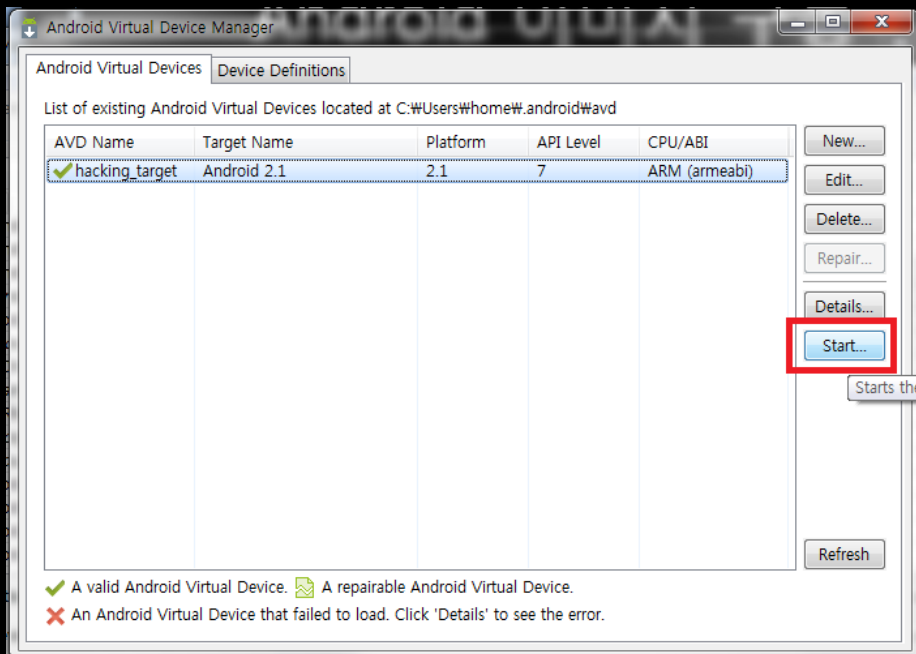


이름	수정한 날짜	유형	크기
build.prop	2013-08-21 오전...	PROP 파일	2KB
kernel-qemu	2013-08-21 오전...	파일	1,571KB
NOTICE.txt	2013-08-21 오전...	텍스트 문서	786KB
ramdisk.img	2013-08-21 오전...	디스크 이미지 파일	229KB
source.properties	2013-08-21 오전...	PROPERTIES 파일	17KB
system.img	2013-08-21 오전...	디스크 이미지 파일	222,792KB
userdata.img	2013-08-21 오전...	디스크 이미지 파일	5,466KB

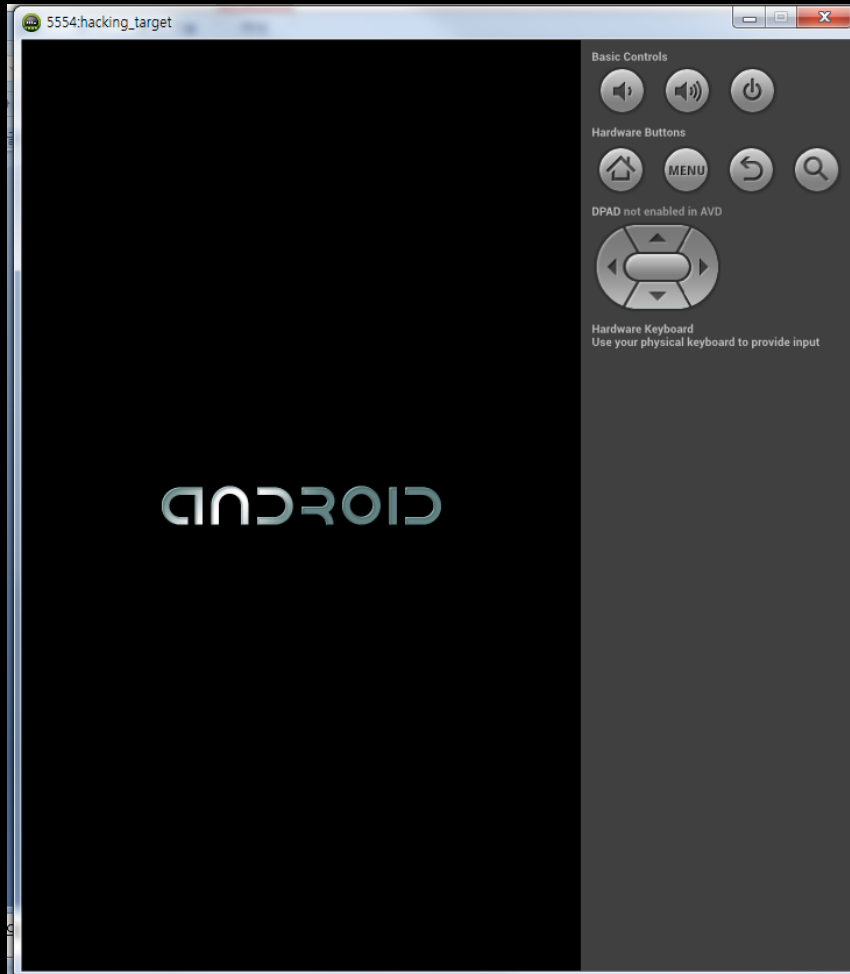
Android 이미지 설정



Android 이미지 구동



야호 공짜 스마트폰이다



Webkit 취약점

- CVE-2010-1759
 - webkit normalize 취약점
 - 유형 : use-after-free
 - 공격 대상
 - Android 2.1 (éclair)
 - Android 2.2 (froyo)
 - Android 2.3 (gingerbread)
 - Drive-by-download 형태
 - "http://xxx.com/xxx.html <- "클릭해보셈"

취약점 테스트

- Browser 실행
- <http://hackerschool.org/e/> 접속
- 링크 클릭
- 수십초 기다림 (heap spray)
- Browser 비정상 종료 확인

Android 디버깅

- logcat

```
C:\Windows\system32\cmd.exe

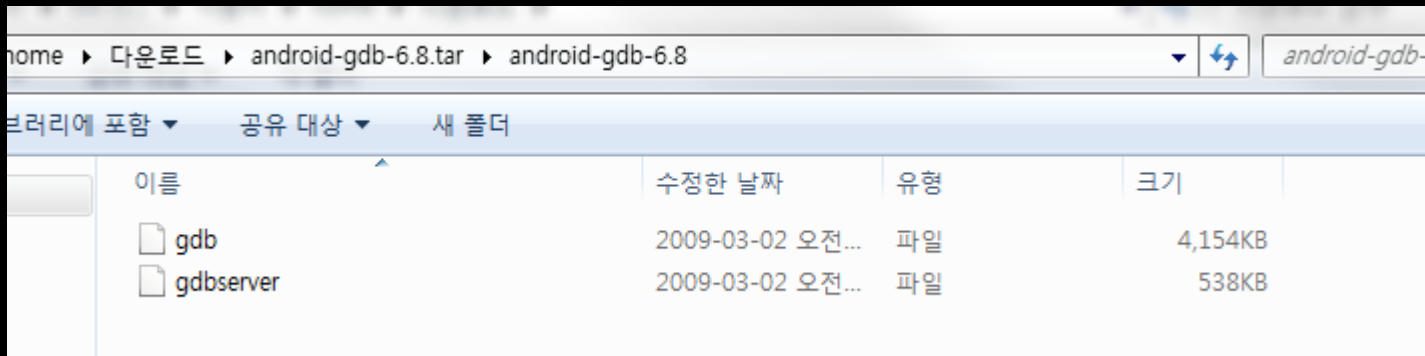
C:\Users\home\Downloads\adt-bundle-windows-x86-20130729\adt-bundle-windows-x86-20130729\sdk\platform-tools>adb logcat_
```

```
C:\Windows\system32\cmd.exe - adb logcat

school.org/e/exploit2.html:25
I/DEBUG < 27>: *** **
I/DEBUG < 27>: Build fingerprint: 'generic/sdk/generic/:2.1-update1/ECLAIR/35983:eng/test-keys'
I/DEBUG < 27>: pid: 219, tid: 232 >>> com.android.browser <<<
I/DEBUG < 27>: signal 11 (SIGSEGV), fault addr 00c400ca
I/DEBUG < 27>: r0 00620065 r1 00520052 r2 4858dbb0 r3 00620065
I/DEBUG < 27>: r4 00500035 r5 00000001 r6 00452e70 r7 4858dbbc
I/DEBUG < 27>: r8 4858ed80 r9 4374eee0 i0 4374eec8 fp 00379058
I/DEBUG < 27>: ip aa413808 sp 4858db98 lr aa067ca5 pc 00500034 cpsr 20000030
W/browser < 219>: Console: JavaScript execution exceeded timeout. http://hacker
school.org/e/exploit2.html:25
I/DEBUG < 27>: #00 pc 00500034 [heap]
I/DEBUG < 27>: #01 pc 00067ca2 /system/lib/libwebcore.so
I/DEBUG < 27>: #02 pc 0005d648 /system/lib/libwebcore.so
I/DEBUG < 27>: #03 pc 00067b90 /system/lib/libwebcore.so
I/DEBUG < 27>: #04 pc 001ad39c /system/lib/libwebcore.so
I/DEBUG < 27>: #05 pc 001d3ae0 /system/lib/libwebcore.so
I/DEBUG < 27>: #06 pc 001d5dde /system/lib/libwebcore.so
I/DEBUG < 27>: #07 pc 001e926e /system/lib/libwebcore.so
I/DEBUG < 27>: #08 pc 0004b83e /system/lib/libwebcore.so
I/DEBUG < 27>: #09 pc 000d4930 /system/lib/libwebcore.so
I/DEBUG < 27>: #10 pc 000bfd6 /system/lib/libwebcore.so
```

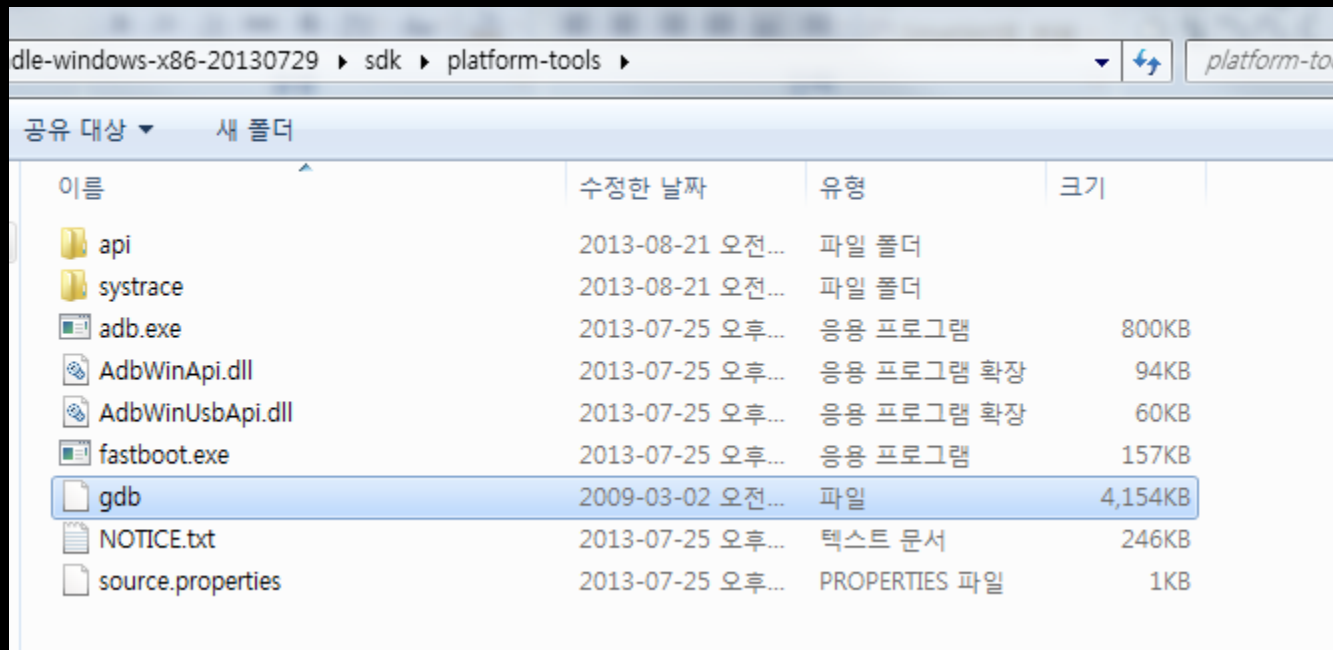
Android 디버깅

- gdb
 - 안드로이드용 gdb
 - <https://sites.google.com/site/ortegaalfredo/android>
- 알집등으로 압축 해제



Android 디버깅

- Adb 있는 곳으로 복사



Android 디버깅

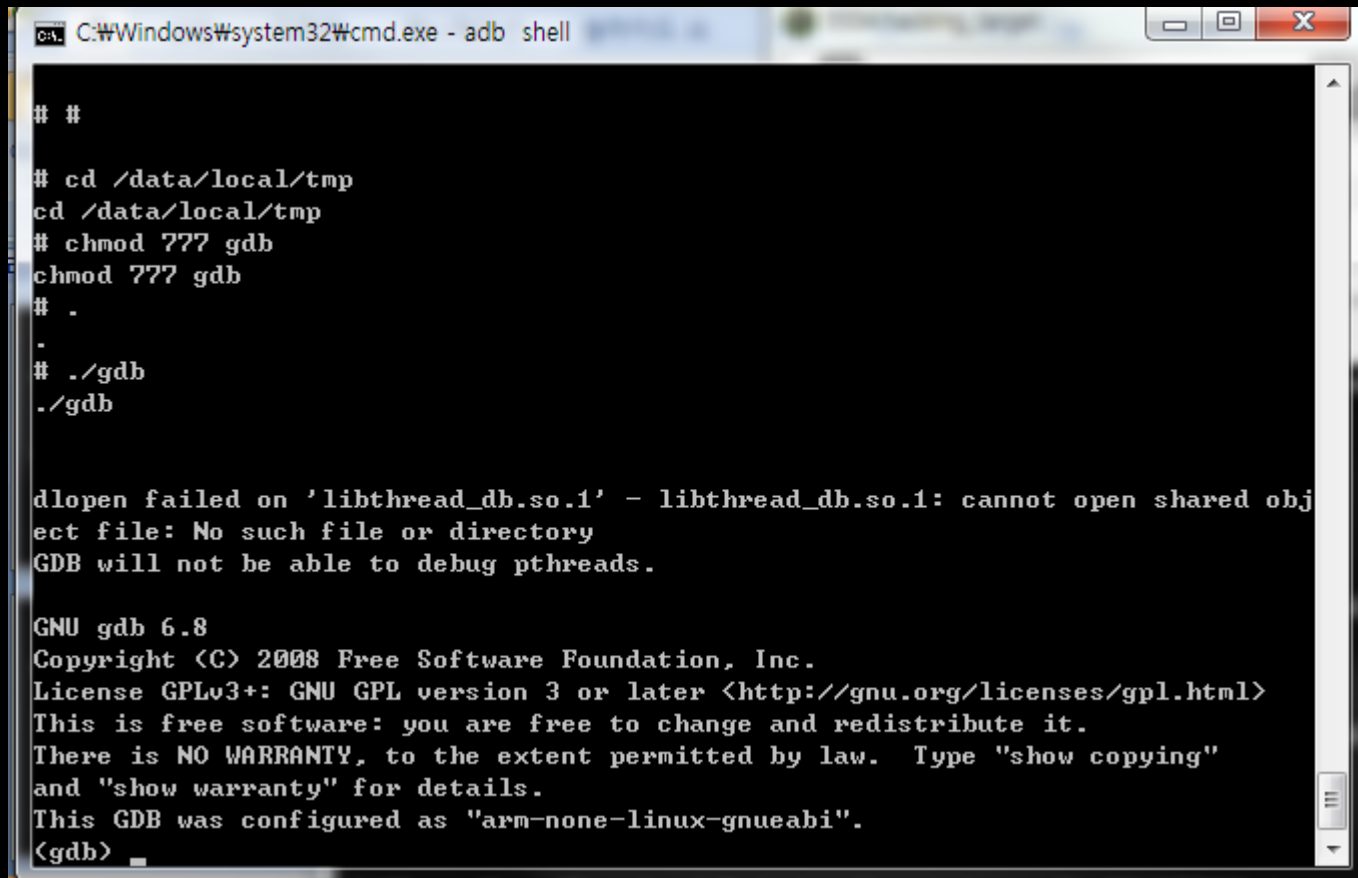
- adb push를 이용하여 폰 안으로 복사
 - 시간 좀 걸림

```
C:\Users\home\Downloads\adt-bundle-windows-x86-20130729\adt-bundle-windows-x86-20130729\sdk\platform-tools>adb push gdb /data/local/tmp/gdb
70 KB/s (4252764 bytes in 58.613s)

C:\Users\home\Downloads\adt-bundle-windows-x86-20130729\adt-bundle-windows-x86-20130729\sdk\platform-tools>
```

Android 디버깅

- 실행 잘 되는 것 확인 후 종료(quit)



```
C:\Windows\system32\cmd.exe - adb shell

# #

# cd /data/local/tmp
cd /data/local/tmp
# chmod 777 gdb
chmod 777 gdb
# .
.
# ./gdb
./gdb

dlopen failed on 'libthread_db.so.1' - libthread_db.so.1: cannot open shared object file: No such file or directory
GDB will not be able to debug pthreads.

GNU gdb 6.8
Copyright (C) 2008 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "arm-none-linux-gnueabi".
(gdb) _
```

Android 디버깅

- Com.android.browser 디버깅

```
C:\Windows\system32\cmd.exe - adb shell

inyin
radio      110    29    171740 24276 ffffffff afe0da04 $ com.android.phone
app_7      136    29    172788 26220 ffffffff afe0da04 $ android.process.acore
app_17     153    29    153604 19484 ffffffff afe0da04 $ com.android.alarmclock
app_3      166    29    154276 20160 ffffffff afe0da04 $ android.process.media
app_14     190    29    163248 20348 ffffffff afe0da04 $ com.android.mms
app_23     206    29    156740 20768 ffffffff afe0da04 $ com.android.email
app_1      244    29    160476 20560 ffffffff afe0da04 $ com.google.process.gapps
root       271    36     728    328 c003d444 afe0d6ac $ /system/bin/sh
app_0      276    29    184900 38036 ffffffff afe0da04 $ com.android.browser
root       276    29     868    332 00000000 afe0c7dc $ R ns

# ./gdb -p 276
./gdb -p 276

dlopen failed on 'libthread_db.so.1' - libthread_db.so.1: cannot open shared object file: No such file or directory
GDB will not be able to debug pthreads.

GNU gdb 6.8
Copyright (C) 2008 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
```

Android 디버깅

- 뭔가 예상치 않은 결과가 나온다면?
- Android용 gdb의 버그일 가능성이 높음
- gdb 대신 gdbserver를 이용하여 원격 디버깅 진행
- <http://techpedia.tistory.com/tag/gdbserver>

Use-After-Free란?

- 동일한 객체의 주소를 두 개의 포인터 A와 B가 가리키고 있다고 가정
- 이 중 A 포인터를 통해 객체를 free한다면?
- B라는 포인터는 free된 객체를 참조하게 됨
- 윈도우, 리눅스, 안드로이드 등 다양한 환경에서 발생할 수 있는 취약점 유형

We make references to the element in 2 different ways

```
var elem1 = doc.getElementsByTagName("textarea")
```

```
var elem2 = doc.getElementById("target")
```

```
<body>  
<textarea id="target" rows=20>blah</textarea>  
</body>
```

출처 : <http://www.slideshare.net/mjza/bsides>

0x00650000

0x00385100

0x0000a000

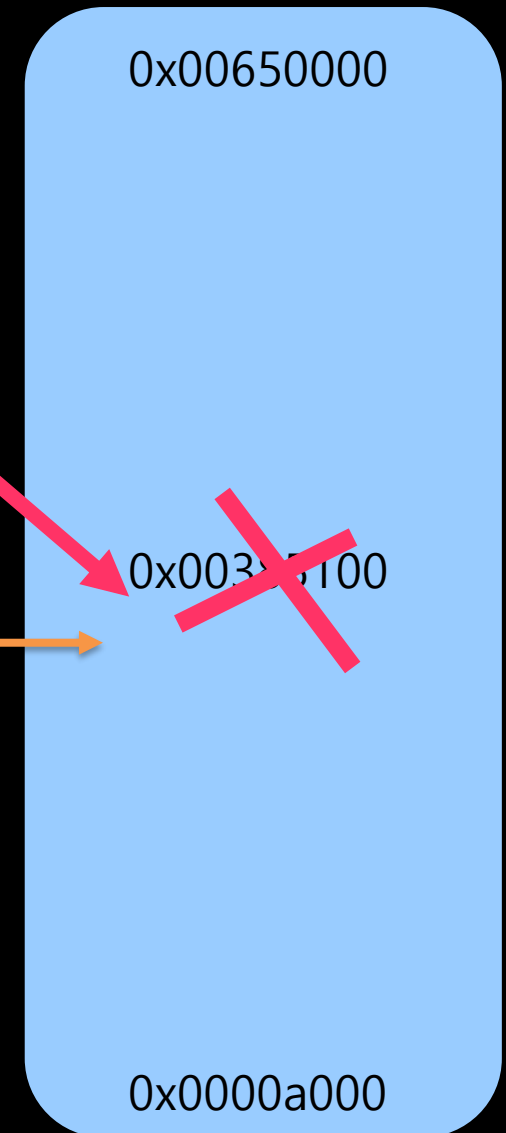
We remove the element using our second reference. This essentially unlocks the memory that both variables are referencing. The elem1 var retains its pointer to the deallocated spot in memory

```
var elem1 = doc.getElementsByTagName("textarea")
```

```
var elem2 = doc.getElementById("target")
```

```
elem2.parentNode.removeChild("target");
```

```
<body>  
<textarea id="target" rows=20>blah</textarea>  
</body>
```

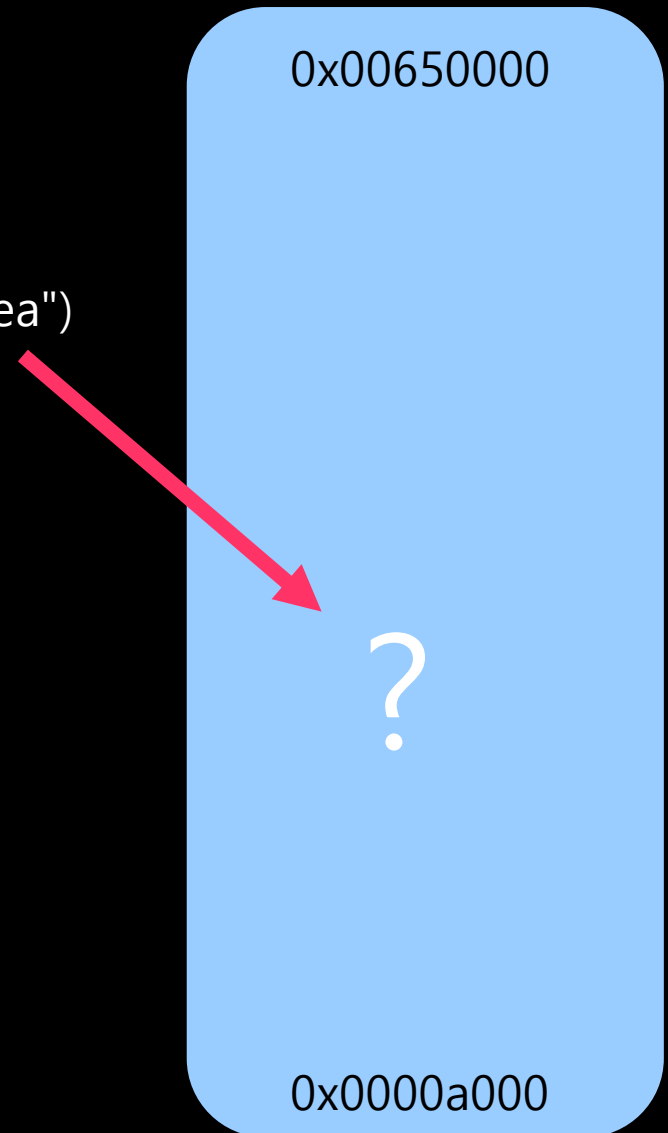


We are left with a pointer to memory that is deallocated. We can now reallocate this memory

```
var elem1 = doc.getElementsByTagName("textarea")
```

```
<body>
```

```
</body>
```



Using a for loop we can create the same small string over and over until we collect garbage and refill the memory with our new data

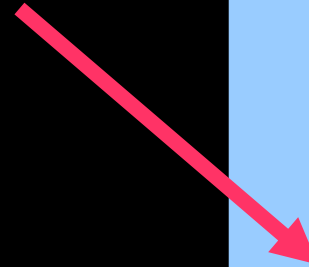
```
var elem1 = doc.getElementsByTagName("textarea")
```

```
for (var i = 0; i < 10000; i++) {  
  var s = new String("LALA");  
}
```

```
<body>
```

```
</body>
```

0x00650000



0x0000a000

We can now request data from our original variable.

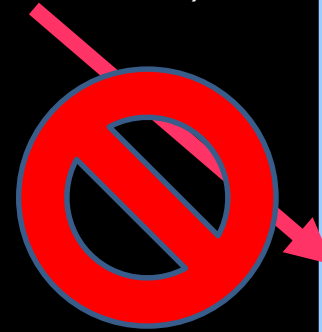
```
var elem1 = doc.getElementsByTagName("textarea")
```

```
for (var i = 0; i < 10000; i++) {  
  var s = new String("LALA");  
}
```

```
elem1.innerHTML;
```

```
<body>
```

```
</body>
```



0x00650000

LALA	LALA	LALA
LALA	LALA	LALA
LALA	LALA	LALA
LALA	LALA	LALA
LALA	LALA	LALA
LALA	LALA	LALA

0x0000a000

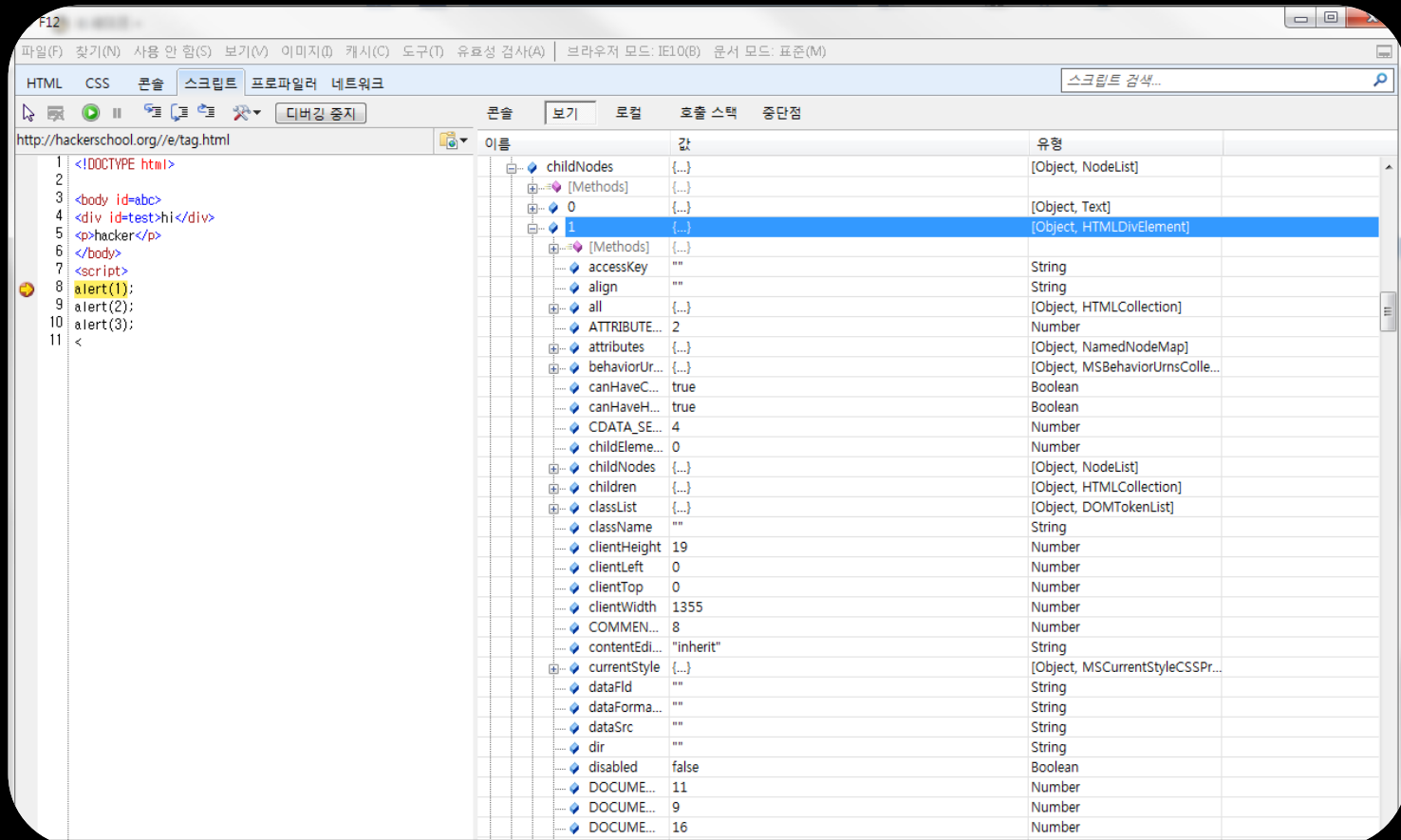
Exploit 코드 분석 시연

요약

- Normalize() 메서드에서 발생하는 use-after-free 취약점
- 중복 참조되는 객체
 - "b=a" attributes 객체
- 이를 참조하는 객체들
 - 1. elem1 객체
 - 2. Normalize() 내부 객체 (normalizing 시작 전 생성)
- 이를 삭제하는 코드
 - 이벤트 핸들러 내의 removeAttribute("b")
- 텍스트 노드가 합쳐질 때 DOMSubtreeModified 이벤트 발생
 - <div id="test1"></div> => "" 빈 텍스트 노드
 - createTextNode에 의해 생성된 "hi" 텍스트 노드
- 이론적으로는 elem1 하나만으로 공격 가능하지만 성공 확률을 높이기 위해 elem2와 elem3가 추가로 사용됨

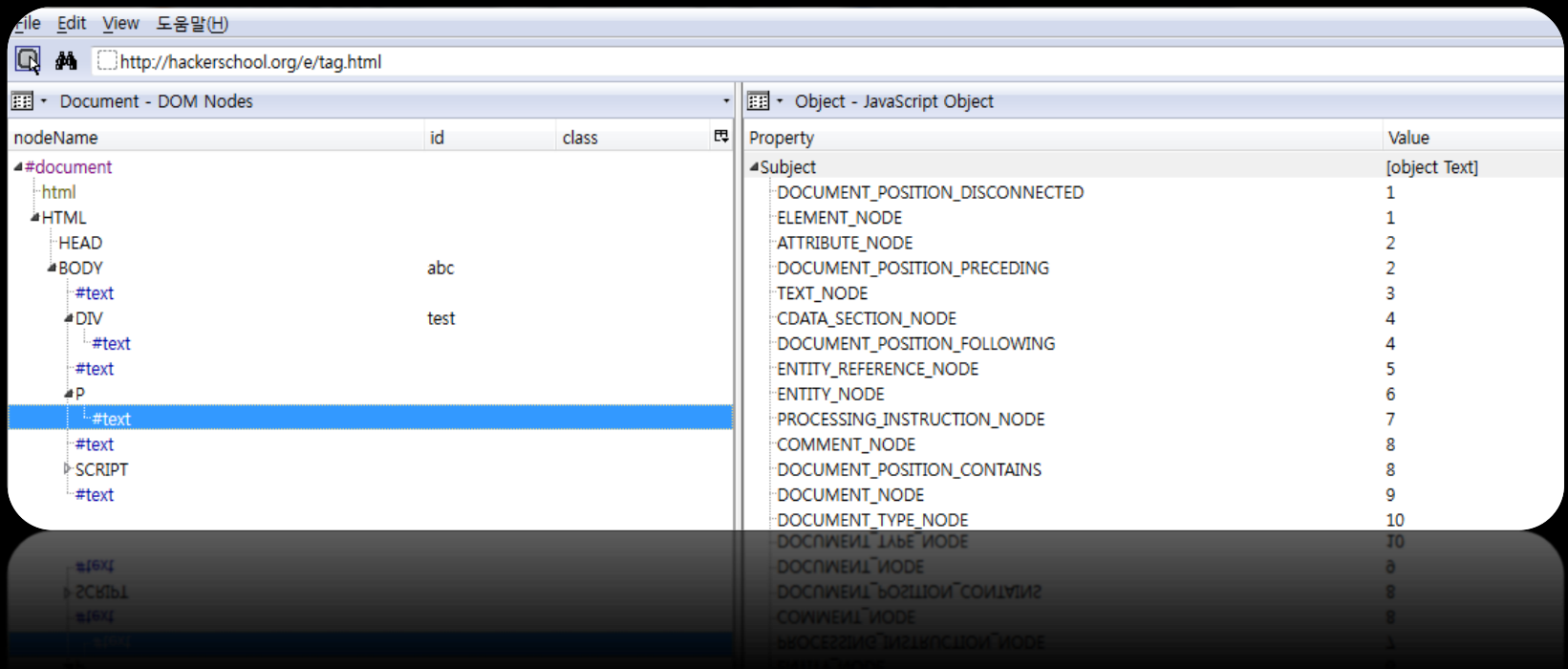
Object Debugging

- IE -> 디버깅모드 -> 보기 -> ID 추가



Object Debugging

- 설치
 - Firefox -> 메뉴 -> 부가기능 -> 검색 -> DOM inspector
- 실행
 - 메뉴 -> 개발자 기능 -> DOM inspector



과제

- 안드로이드 에뮬레이터 환경에서 CVE-2010-1759 취약점을 성공시키시오.
- 목표 : Remote Shell 획득
- 과제제출 : cybermong@grayhash.com
- 기한 : 9월 7일

Q/A

감사합니다.