铪？

发布于 2016年7月12日 3 天前

作者：1024 链接：http://zhuanlan.zhihu.com/p/21558357 (http://zhuanlan.zhihu.com/p/21558357) 来源：知乎 著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。

这几年安卓系统的普及速度可谓迅猛，一时间各式各样的设备都承载着安卓系统，手机、平板、机顶盒等都忠实地成为了安卓系统的用户。由于安卓系统在移动设备上的使用率最高，而移动设备上存储的数据往往涉及到个人隐私，如手机通讯录、短信内容、拍摄照片、阅读书目、保存文档等，有时更会涉及到经济利益，这诱惑着一些利益集团开始制作基于安卓系统的远程控制程序，即安卓系统木马。

首个安卓系统木马应属2010年出现的“Trojan-SMS.AndroidOS.FakePlayer.a”，这是一个以扣取用户手机话费为目的的盈利性安卓系统木马。随着需要的发展，单纯的盈利性木马已经不是重点，用户的隐私数据才是核心，尤其是具有用户行为监视性的木马最受关注。所谓“用户行为监视性的木马”就是指该类安卓木马能够监视用户的所在、所说、所做。

所在”即用户所处位置在哪里；“所说”即通话内容，聊天内容；“所做”即在操作什么程序，在干什么活动。

这一类的木马由于涉及到用户核心利益，往往经济价值较大，多用于私人侦探、商业窃密等领域，平时很难见到，更不要说了解其核心代码、实现机制。为此，本文将逐步向读者揭秘这些高级安卓木马的核心实现技术，帮助大家更好地了解这些木马实现技术，从而做好对个人隐私的保护，防范该类木马的入侵。本文旨在讨论技术，凡利用本文技术进行违法活动的作者与杂志概不负责。

如何实现定位

如果你利用百度搜索安卓定位原理，会发现百度给出的解释不外乎是利用GPS或者手机基站定位，甚至结合Wi-Fi信号。原理不错，但这只是原理，要想具体实现定位可是有一定难度的。以手机基站定位为例，现在传统的实现方式是利用AndroidSDK中的API（TelephonyManager）获得MCC、MNC、LAC、CID等信息，然后通过Google的API获得所在位置的经纬度，最后再通过GoogleMap的API获得实际的地理位置。这其中：

MCC即MobileCountryCode，移动国家代码（中国的为460）；

MNC，MobileNetworkCode，移动网络号码（中国移动为00，中国联通为01）；

LAC，LocationAreaCode，位置区域码；

CID，CellIdentity，基站编号，是个16位的数据（范围是0到65535）。

由于谷歌存储了MCC、MNC、LAC、CID等信息，一旦我们能够获取当前移动设备所在基站的这些数据，就可以通过向谷歌的“http://www.google.com/loc/json (http://www.google.com/loc/json)”网址发送查询数据获取基站所在经纬度。

得到经纬度后，我们将其转换为实际地址，这需要向谷歌的“http://maps.google.cn/maps/geo?key=abcdefg&q= (http://maps.google.cn/maps/geo?key=abcdefg&q=)”发送经纬度数据，最终获得移动设备所在实际地址。这样的实现代码在网上很多，你会发现它们都不好使了，为什么呢？因为“http://www.weixianmanbu.com/ (http://www.weixianmanbu.com/)”这个网址现在已经不能访问了。这个可悲的消息使得我们意识到必须采用一种相对稳妥的方法来实现移动设备定位。在对某个安卓木马程序做逆向分析时，发现一种新的基于手机基站定位实现技术。当然在这之前，细心的读者会发现为什么我们一直在详细讲解基于手机基站的定位实现，而不采用最为常用的GPS。

因为手机这样的移动设备一旦进入到房屋内等封闭场所，GPS信号就衰减为0，不足以实现定位，而手机信号多半都是存在的，所以基于手机基站的定位方式更为稳妥，这就是为什么很多高级安卓木马会采用该方式实现定位的原因。言归正传，我们发现的这个安卓木马采用了基于百度提供的定位SDK。根据百度官方的解释：百度Android定位SDK支持Android1.5以及以上设备，提供定位功能，通过GPS、网络定位（WIFI、基站）混合定位模式，返回当前所处的位置信息。

反地理编码功能：

解析当前所处的位置坐标，获得详细的地址描述信息。如此丰富的技术支持，难怪该安卓木马会采用这个SDK。

百度Android定位SDK的使用非常简单，首先在百度的官网下载最新的库文件，将liblocSDK.so文件拷贝到libs/armeabi目录下，将locSDK.jar文件拷贝到工程根目录下，并在工程属性->JavaBuildPath->Libraries中选择AddJARs，选定locSDK.jar，确定后返回，就可以在程序中使用百度Android定位SDK了。在代码实现时，首先需要初始化LocationClient类，其代码如下：

publicLocationClientmLocationClient=null; publicBDLocationListenermyListener=newMyLocationListener(); publicvoidonCreate(){ mLocationClient=newLocationClient(this) //声明LocationClient类 //注册监听函数 mLocationClient.registerLocationListener(myListener); } 接着实现BDLocationListener接口。BDLocationListener接口有一个方法，作用是接收异步返回的定位结果，参数是BDLocation类型参数。其代码如下：

publicclassMyLocationListennerimplementsBDLocationListener{@OverridepublicvoidonReceiveLocation(BDLo {if(location null)return;
StringBuffersb=newStringBuffer(256);sb.append("time:");sb.append(location.getTime());sb.append("\nerrorcod if(location.getLocType() BDLocation.TypeGpsLocation){
sb.append("\nspeed:");sb.append(location.getSpeed());sb.append("\nsatellite:");sb.append(location.getSatellit { sb.append("\naddr:");sb.append(location.getAddrStr());}logMsg(sb.toString()); }} 接着设置参数。设置定位参数包括定位模式（单次定位，定时定位），返回坐标类型，是否打开GPS等。实现代码如下：

LocationClientOptionoption=newLocationClientOption();option.setOpenGps(true);option.setAddrType("detail"); 最后，发起定位请求。请求过程是异步的，定位结果在上面的监听函数中获取，代码如下：

if(mLocClient!=null&&mLocClient.isStarted())mLocClient.requestLocation();elseLog.d("LocSDK_2.0_Demo1","l 实际测试效果如图1所示。从图中可以看出，演示程序准确定位到了我此刻手机所在的位置，定位精度在百米内。木马程序一旦使用了这样的技术，完全可以实现对用户所在的监视，你此刻是不是有一种毛骨悚然的感觉呢？