自由Android安全研究员陈愉鑫：移动App灰色产业案例分析与防范

原创

csdn业界要闻 于 2017-12-01 16:02:22 发布 814 收藏 2
文章标签： 安全 陈愉鑫 移动App 看雪安全开发者峰会

11月18号，2017看雪安全开发者峰会在北京悠唐皇冠假日酒店举行。来自全国各地的开发人员、网络安全爱好者及相应领域顶尖专家，在2017看雪安全开发者峰会汇聚一堂，只为这场“安全与开发”的技术盛宴。

在移动互联网时代，互联网业务飞速发展，在这样的大背景下滋生了一条以刷单、倒卖、刷榜、引流、推广为主的灰色产业链。山寨APP遍地开花，给原APP的开发者带来了很大的冲击，这些人往往被称为“打包党”。他们以低成本换取了高额的利润，给互联网企业以及用户都带来了巨大的损失。虽然加固技术、风险控制、设备指纹、验证码等技术也都在飞速发展，但实际效果并不能让人满意。本次峰会上，自由Android安全研究员、看雪会员陈愉鑫揭露了多个真实案例的技术细节、开发流程、运营流程，描述了一条以刷单、倒卖、刷榜、引流、推广为主的灰色产业链，并提出了一些防护建议，协议安全需要从体系上进行加强。

自由Android安全研究员、看雪会员 陈愉鑫

> 陈愉鑫，自由Android安全研究员、看雪会员。热爱APP通信协议分析，曾经分析过各商城、直播软件协议流程并还原加密。对灰色产业链流程略有研究。

以下为演讲：

陈愉鑫：大家好，我叫陈愉鑫，网名无名侠。今天很高兴能够在这里和大家相聚，这一次看雪的峰会办的特别好，我是2012年成为看雪的会员，这几年我在看雪也学了很多的知识，非常感谢看雪能够给我们提供这样的一个平台，我今天演讲的题目是移动APP灰色产业案例分析与防范。

大家玩过薅羊毛吗？我首先讲的是薅羊毛的工业革命。之所以叫它为什么工业革命，是因为我想讲薅羊毛怎么样从手工到脚本自动化完成。今天我会以小米商城、抢购软件做一个简单的例子，针对这些东西我会提出防范的建议。

羊毛党的工业革命。手动薅然后转换成自动化脚本，这一群人可以做什么？首先就是情报的搜集，要提前知道有那些活动，比如说今天发什么券他们都会提前知道，或者某一个什么APP有奖可以抽。羊毛党提前知道活动之后再找合伙人准备小号，准备小号就会批量注册，批量注册就涉及到注册协议的分析，现在这些APP里面都做了加密算法，这些加密算法有没有用，未知。也有很多加固，这些加固是否有效果，我们也未知。注册小号之后就会准备VPS，VPS越多请求量就越大。比如抽奖，用很多机器抽，肯定会比单人快。找技术人员，比如说分宜写法，解决算法脱掉破解，这些薅羊毛相当于做一个管理，管理下面的技术人员。活动上线之后，这个协议以及算法都生效了可以开始分析了，活动上线之后就要在第一时间内把这个东西做出来，所以上线之后就会找技术脱壳，找技术分析协议，直到出一个完整的软件出来。

安卓App容易被反编译以及调试，大部分的算法都是用Java写的，反编译非常容易，还有一些非常核心的算法会放在SO里面，但是效果也不是很好。做这一块的技术人很多，会的人也很多，做起来特别方便。

这是小米的抢购软件，在小米商城上有活动，比如说小米商城有一个板块叫做真心想要，这个板块里面就定期有非常低价商品出售的活动，这些活动一旦出来了，有一些人就会写脚本，买很多的服务器，同时开抢，抢到之后以稍微高的一点的价格转手出去。很多人从这个里面捞了很多钱。

首先我们来分析一下小米抢购软件的解决方案，要怎么样做一个抢购软件？我们要抢购东西首先要有一个API接口来源，小米应该有三个端，Web端，app以及盒子端。App又有Android和iOS两个端 ，iOS分析成本比较高，安卓比较容易。

小米盒子的商城里面时候也有一些便宜的东西，盒子看着挺难分析的，实际上可以很容易把里面的算法、协议提取出来。小米盒子抓包可能有一些困难，因为小米盒子没有代理配置功能，但是小米也自己犯了一个坑，发包函数里面会把发的数据通过log输出，我们通过查看logcat日志，可以知道他发送了什么。

做抢购就需要大量的小号，这些小号需要注册脚本，就涉及到一个验证码的识别，这个解码平台很多验证码就可以识别。今天上午看到一个笑话，验证码需要你手机上的尾码乘以数字是多少，这种解码平台就无能为力了，感觉特别好，这是一个特别好的思路。

小米的登陆算法也挺复杂的，会涉及到很多的步骤包括设备信息绑定等等。登陆之后，我们还会涉及到一个批量设置收货地址，批量下单。这些东西的原理还是非常简单，模拟一下小米数据包，像爬虫一样。抓包不能完全防止，因为抓包禁止不了，服务端也不可能完全对这个客户端鉴权，所以只能提高协议难度。

最简单的方法是最每一个数据包里添加一个sign 字段，仅仅通过抓包我们是无法得知该sign字段的计算方法的。我们知道一点的是，这个字段必须输入正确才可以成功登陆一个账号，一般这种解决思路就是通过逆向APP寻找注册密钥，还原这个过程，把真正密码以及提交后的加密密码对应生成关系，生成之后又可以让服务器认可，服务器认可之后就可以登陆，这是一个基本的思路。

现在许多应用开发商为了图方便，于是就直接调用Java的Crypto算法库。直接调用这样的算法库会存在一个潜在的安全隐患，密钥是固定的，算法也是固定的。既然是Java的算法库，那么就可以通过Hook的方法输出密钥、加密数据等重要信息。即使一个APP已经加固甚至做过许多混淆也没有太大的用处。在这种情况下，逆向就显得太慢了！

千里之提毁于蚁穴，有小部分APP选择使用加固产品，但是只使用了DEX加固功能，他们又将协议加密算法的代码放在SO里面，这样加固就没有起到太大的作用。

我们如何利用经过OLLVM编译的SO呢？ SO文件是已经编译的可执行文件，那么既然都叫可执行文件了，是否有手段可以让这些SO执行起来呢？ 有部分APP中的加密SO有x86版本，这就非常好办，直接load到内存，设置环境执行即可。如果只有ARM版本的SO又该如何处理呢？ 我选择使用Unicorn 库模拟执行。Unicorn是一款基于Qemu模拟器内核的库，提供了许多方便的API接口。利用该库，我可以通过编程，完完全全虚拟出一颗ARM的处理器以及内存。

模拟执行成本就低了，而且使用Unicorn模拟执行还是线程安全的，非常适合羊毛党的业务需求。

模拟执行对抗方法也很简单，在核心的算法内增加上下文依赖，增加系统API调用，尽量避免纯运算函数。

其它的安全建议还有许多，例如修改标准算法也是不错的选择。最简单的方法是修改Base64的映射表、替换字符等等。再复杂一点就是修改Hash函数的初始化常量、增删算法部分逻辑等等，比较典型的是jd的tea算法。

自己实现VM也是一种不错的方案，自己实现VM有一个好处，可以根据自身APP的业务需求来设计。Bytecode可以灵活更新，甚至可以根据设备信息生成唯一加密算法。

最后，我建议核心点的协议采用一些二进制的序列化协议，这样能增加分析的难度。

> 注：本文根据大会主办方提供的速记整理而成，不代表CSDN观点。

2017看雪安全开发者峰会更多精彩内容：

- 2017看雪安全开发者峰会在京召开 共商网络安全保障之策
- 中国信息安全测评中心总工程师王军：用技术实现国家的网络强国梦
- 兴华永恒公司CSO仙果：Flash之殇—漏洞之王Flash Player的末路
- 中国婚博会PHP高级工程师、安全顾问汤青松：浅析Web安全编程
- 威胁猎人产品总监彭巍：业务安全发展趋势及对安全研发的挑战
- 启明星辰ADLab西南团队负责人王东：智能化的安全——设备&应用&ICS
- 腾讯反病毒实验室安全研究员杨经宇：开启IoT设备的上帝模式
- 绿盟科技应急响应中心安全研究员邓永凯：那些年，你怎么写总会出现的漏洞
- 腾讯游戏安全高级工程师胡和君：定制化对抗——游戏反外挂的安全实践
- 绿盟科技网络安全攻防实验室安全研究员廖新喜：Java JSON 反序列化之殇
- 阿里安全IoT安全研究团队Leader谢君：如何黑掉无人机