Beijing Forest Studio
北京理工大学信息系统及安全对抗实验中心

Android Hook 技术分析

硕士研究生 柯懂湘

2017年10月09日

内容提要

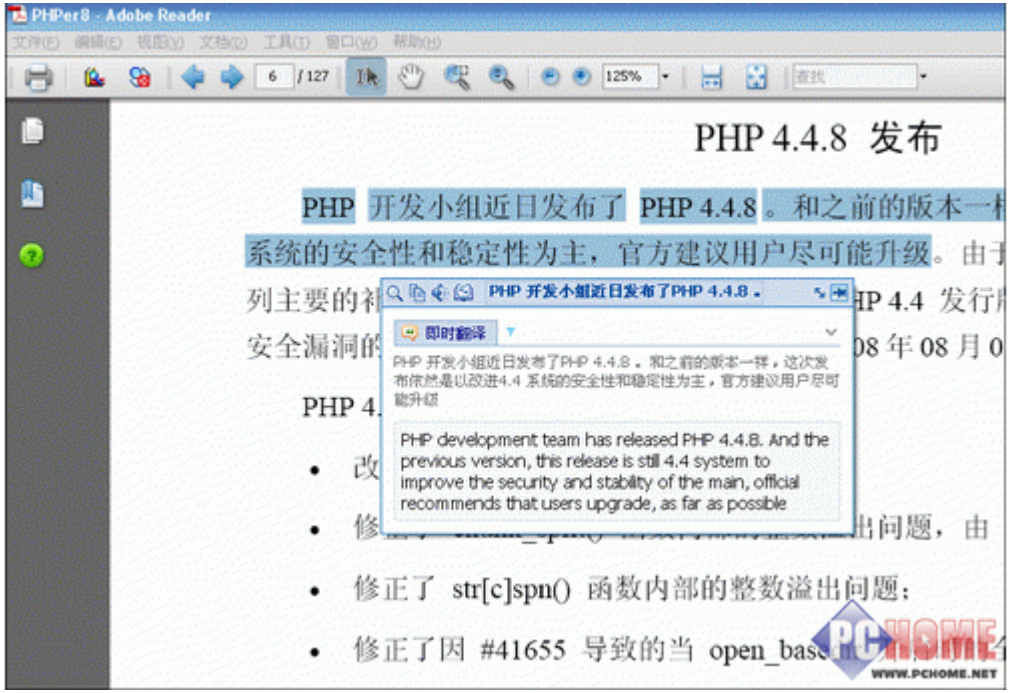
- Hook技术简介
  - Hook技术起源
  - 什么是Hook技术？
- Hook技术的原理
  - Hook技术分类
  - Hook技术实例
  - 实例的原理分析
- Hook技术用途及防范
  - Hook技术用途
  - Hook检测与修复

Hook技术起源

Hook技术起源

- 先有问题，后有技术，Hook技术解决了什么问题？

系统提供的接口无法满足我们的需求，这时就需要修改系统接口，使之可以更好的服务于程序。



Hook技术起源

- 翻译软件设计的正常思路：

划线取词的事件->系统捕获到这个事件->系统通知给翻译软件->翻译软件将翻译的结果输出

- 系统本身的处理流程：

划线取词的事件->系统捕获到这个事件->调用系统函数ExTextOut函数

Hook技术起源

- 解决方案

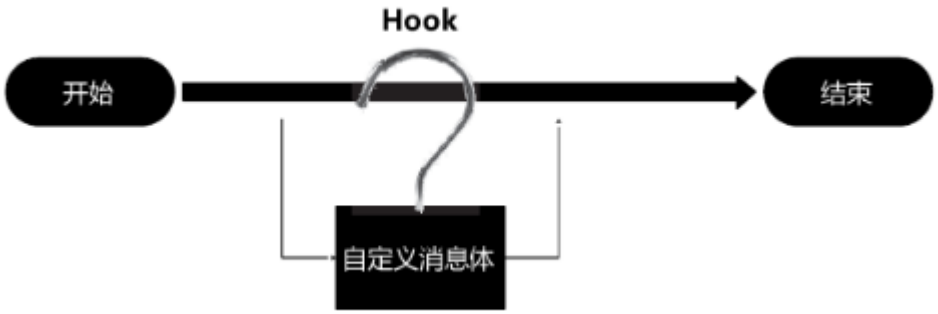
  将系统函数ExTextOut函数更改为翻译软件的处理函数

- 处理流程：

  划线取词的事件->系统捕获到这个事件->调用翻译软件的处理函数->翻译，并将结果输出

什么是Hook技术？

Hook是钩子的意思，而“钩子”的意思，就是在事件传送到终点前截获并监控事件的传输，像个钩子钩上事件一样，并且能够在钩上事件时，处理一些自己特定的事件。Hook的本质就是函数的劫持。



Hook技术原理

Andorid Hook技术分类

- Android平台Hook技术分类
  - Java层 API hook
  - Inline Hook
  - GOT Hook

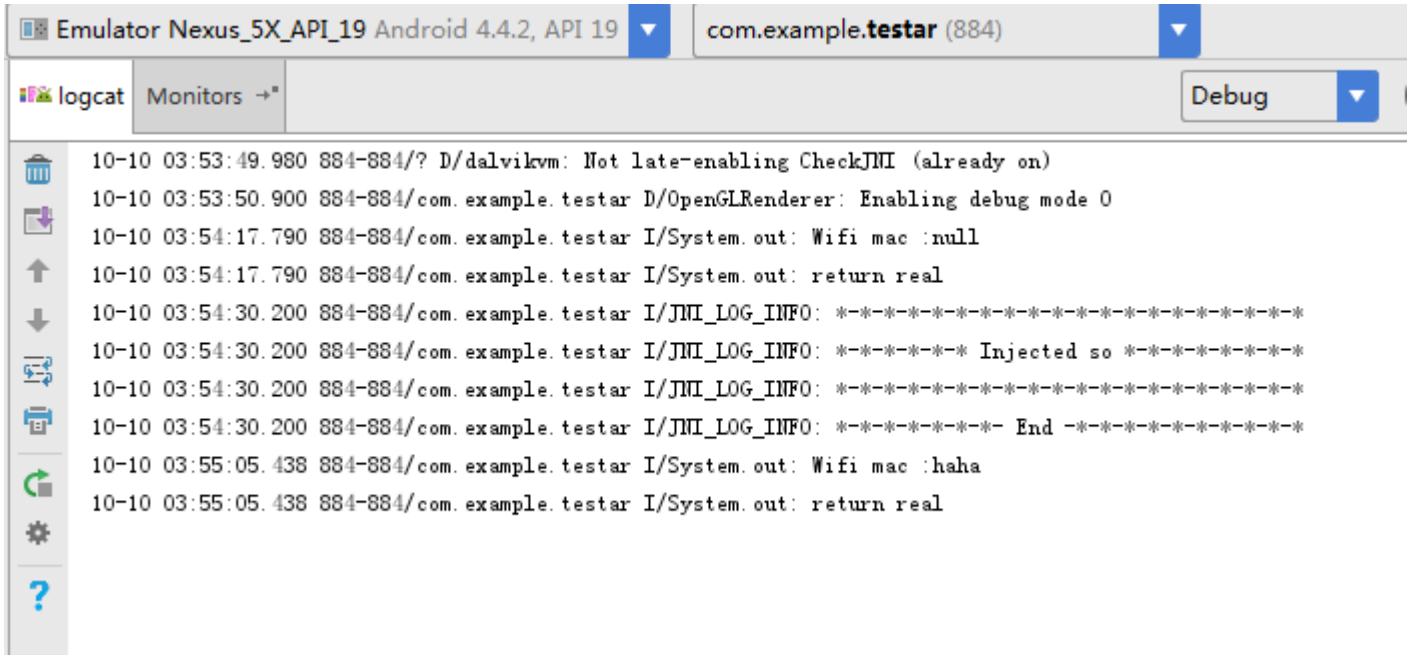
- 实例
  - So注入Hook Java层API

Hook技术实例


```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    Button btn = (Button) findViewById(R.id.button1);
    btn.setOnClickListener((v) -> {
            // TODO Auto-generated method stub
            WifiManager wifi = (WifiManager) getSystemService(Context.WIFI_SERVICE);
            WifiInfo info = wifi.getConnectionInfo();
            System.out.println("Wifi mac :" + info.getMacAddress());
            System.out.println("return " + test());
            //InjectApplication ia = (InjectApplication) getApplication();
            //System.out.println(ia.test());
    });
}
```

Hook技术实例


Emulator Nexus_5X_API_19 Android 4.4.2, API 19
com.example.testar (884)
logcat
Monitors
Debug
10-10 03:53:49.980 884-884/? D/dalvikvm: Not late-enabling CheckJNI (already on)
10-10 03:53:50.900 884-884/com.example.testar D/OpenGLRenderer: Enabling debug mode 0
10-10 03:54:17.790 884-884/com.example.testar I/System.out: Wifi mac :null
10-10 03:54:17.790 884-884/com.example.testar I/System.out: return real
10-10 03:54:30.200 884-884/com.example.testar I/JNI_LOG_INFO: *-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
10-10 03:54:30.200 884-884/com.example.testar I/JNI_LOG_INFO: *-*-*-*-*-* Injected so *-*-*-*-*-*-*-*
10-10 03:54:30.200 884-884/com.example.testar I/JNI_LOG_INFO: *-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
10-10 03:54:30.200 884-884/com.example.testar I/JNI_LOG_INFO: *-*-*-*-*-*-*- End -*-*-*-*-*-*-*-*-*-*
10-10 03:55:05.438 884-884/com.example.testar I/System.out: Wifi mac :haha
10-10 03:55:05.438 884-884/com.example.testar I/System.out: return real

预备知识

- Android平台有两个世界
  - Java世界
  - Native世界，Native指的是C/C++
  - Java世界和native世界可以通过jni来交互
  - Java调用C/C++代码，需要有个绑定的过程

- Java程序是由Dalvik虚拟机执行的
  - Java程序中的每个函数在Dalvik虚拟机中都会有一个Method结构来描述，包含函数签名(函数名，函数参数)，函数类型(Native/Java)。

预备知识

- Dalvik虚拟机是如何执行函数调用的?
  - Dalvik虚拟机首先找到该函数的Method对象，判断函数类型
  - 如果该函数java函数，调用函数dvmInterpret来执行它的代码
  - 如果该函数是native函数，则直接跳转到该nativeFunc执行

- 动态链接库
  - 可以在程序运行时，有系统动态加载到内存
  - dlopen函数：打开指定的动态链接库
  - dlsym函数：取函数执行地址
  - dlclose函数：关闭动态链接库

Hook技术原理

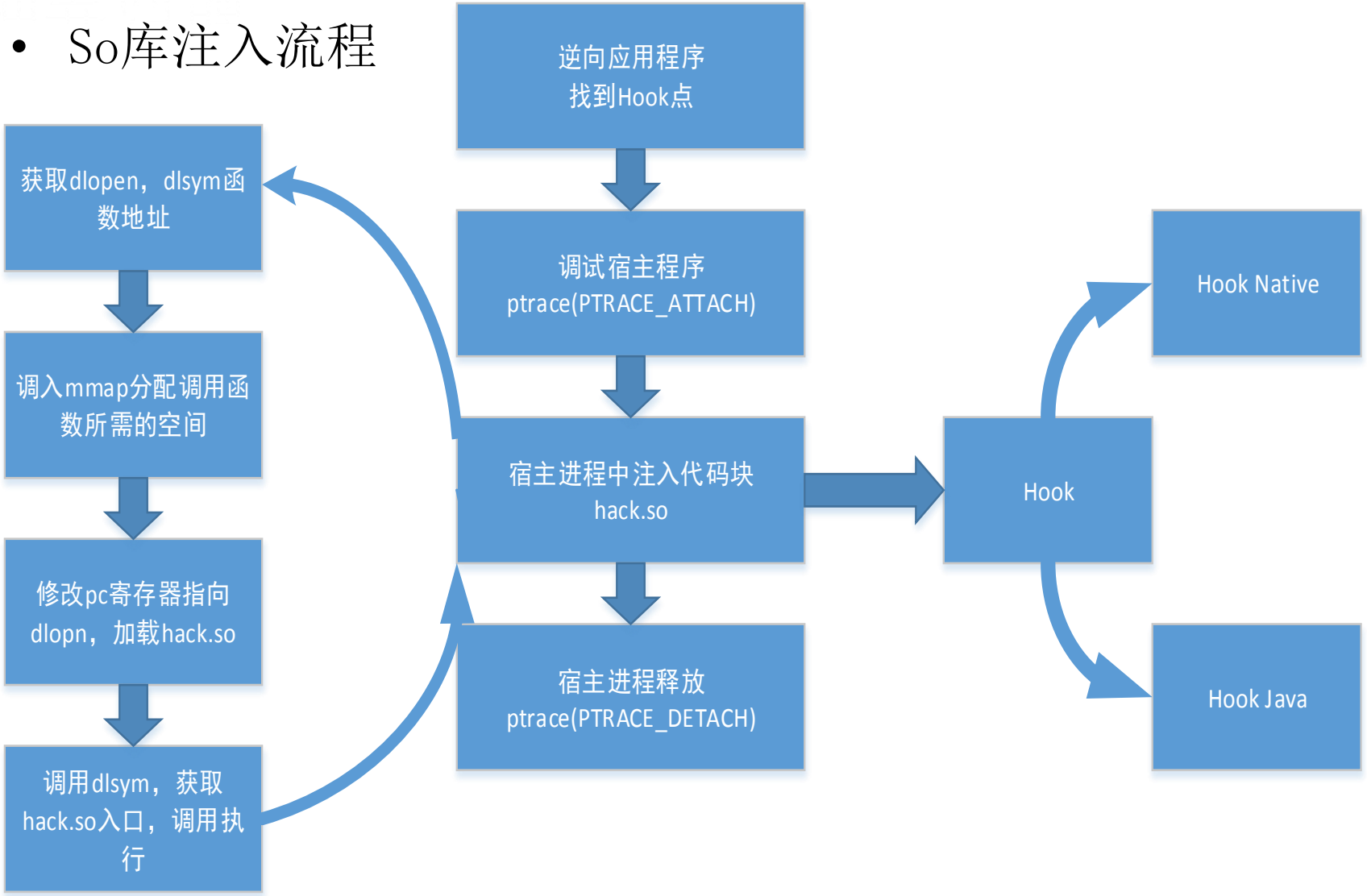
- Hook技术本质是函数调用的劫持
- Hook核心步骤
  - 将自定义的GetMacAddress函数注入到目标App中
  - 更改目标App的GetMacAddress函数的Method结构体，使目标函数由Java方法变成native方法；
  - 将我们GetMacAddress函数的实现与我们注入的函数绑定
  - 调用GetMacAddress函数

细节问题

- 如何将GetMacAddress函数注入目标app？
  - 利用ptrace函数可以直接更改另一个进程的内存与寄存器
  - 利用ptrace函数可以做到在另一个进程中调用指定函数
  - 详细参见《调试器工作原理》
- 进程so库注入步骤
  - 用ptrace函数附加到目标进程
  - 查找目标进程的dlopen函数地址
  - 调用目标进程的dlopen函数加载包含GetMacAddress函数的动态库
  - 用ptrace函数释放目标进程，恢复原进程的运行

细节问题

- So库注入流程



细节问题

- 为什么要先将GetMacAddress函数更改为Native然后再劫持，而不是直接在Java世界劫持GetMacAddress函数？
  - java世界运行在native世界中的，Dalvik虚拟机是用c/c++编写的
  - 我们可以利用ptrace来干涉其他进程的Native世界内存
  - 没有方法可以直接更改别的进程的Java世界

细节问题

- 如何将GetMacAddress的实现与我们注入的代码绑定?
  - 直接绑定
    - 将Method->nativeFunc设置为我们注入的函数地址
  - 延迟绑定
    - 先将注入的函数注册
    - 将Method->nativeFunc设置为dvmResolveNativeMethod
    - 第一次调用GetMacAddress时，先调用dvmResolveNative Method，该函数负责解析函数地址

Hook技术用途及防范

Hook技术的用途

- 应用开发领域
  - 翻译软件的划线取词翻译
- 安全领域
  - 手机卫士的安装应用前的app检测
- 软件测试领域
  - 在不更改产品代码的情况下，输出一些关键信息

Hook的检测与修复

- Hook的危害是巨大的
  - 注入广告
  - 窃取隐私

- 检测
  - 加入反调试机制
  - 检测自身是否加载了多余的动态库

- 修复
  - 使用dlclose函数卸载多余的动态库

谢谢！

大成若缺，其用不弊。大盈若冲，其用不穷。大直若屈。大巧若拙。大辩若讷。静胜躁，寒胜热。清静为天下正。