防止tweak依附，App有高招；破解App保护，tweak留一手

snakeninny Oct '14

相信大多数AppStore开发者在阅读完《iOS应用逆向工程》之后，对iOS开发的理解会深入不少。一入侯门深似海，tweak的出现让各种裸奔App的作者瞬间跪了，自己的App被玩弄于股掌之间，内牛满面😭
但是，一些防护性优秀App的出现，给安全性堪忧的AppStore注入了一针强心剂。他们用到的防护方式处于业界领先地位，值得我们学习👏。如北京时间9月30日更新的美团iOS客户端4.8.1版，就能够强力阻止各种dylib的注入，使得一切tweak均为狗比。下面，小弟就带大家看看美团是怎么做到的，而我们又可以通过什么方式，破解这种防护。

一、下载美团，把玩把玩

下载完成，启动App之后，ssh到iOS，运行

```
ps -e
```

得到输出

```
  1374 ??         0:03.53 /var/mobile/Applications/DCDC3F9D-227A-414F-B796-54AA9DB
```

好的，拿到了美团的目录路径和进程名。下面我们分别尝试用Cycript注入dylib，和用debugserver来动态调试：

> 此处应有图，挂掉了，也找不回来了 😅

嗯……事情变得棘手了：注入dylib和动态调试都失败了。动态调试失败，很大的概率是采用了ptrace方法，这里就不赘述了，网上随便搜一搜就有很多例子；而dylib注入失败是大家比较少碰到的情况，我们就从它下手，看看到底发生了什么。

二、anti-DYLD_INSERT_LIBRARIES

dylib的注入一般是通过DYLD_INSERT_LIBRARIES这个环境变量来实现的，现在dylib连注入都失败，即其constructor根本未得到执行，说明此行为不是由美团的代码完成的，而应该发生在代码执行前。既然这样的话，此行为多半是因MachO头部的某个标注，导致dyld有意为之的。那么……
我们直接去看看dyld的源代码好了啊！源代码总共也没几行，我们着重看看pruneEnvironmentVariables这个函数，它的注释写道：
同时注意到这个switch case：

```
switch (sRestrictedReason) {
                        case restrictedNot:
                                break;
                        case restrictedBySetGUid:
                                dyld::log("main executable (%s) is setuid or setg
                                break;
                        case restrictedBySegment:
                                dyld::log("main executable (%s) has __RESTRICT/__
                                break;
                        case restrictedByEntitlements:
```

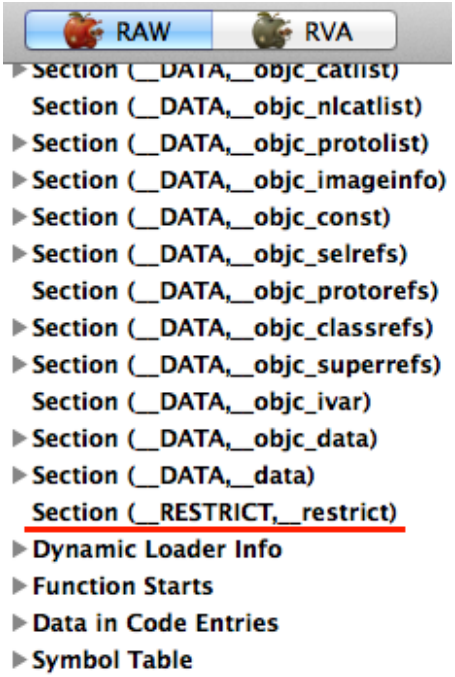
```
                                dyld::log("main executable (%s) is code signed wi
                                break;
                }
```

三种情况下，DYLD_环境变量会被dyld无视，分别是：
1. 可执行文件被setuid或setgid了；
2. 可执行文件含有__RESTRICT/__restrict这个section；
3. 可执行文件被签了某个entitlements。
其中，因为Apple的审核机制，1和3不能由用户指定，因此不大可能出现在AppStore App中。为了确保万无一失，我们简单验证一下就好了：

```
FunMaker-5s:~ root# ls -l /var/mobile/Applications/DCDC3F9D-227A-414F-B796-54AA9DB0
-rwxr-xr-x 1 mobile mobile 25340624 Sep 30 10:23 /var/mobile/Applications/DCDC3F9D-
FunMaker-5s:~ root# ldid -e /var/mobile/Applications/DCDC3F9D-227A-414F-B796-54AA9D
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/Pro
<plist version="1.0">
        <dict>
                <key>keychain-access-groups</key>
                <array>
                        <string>FSS9ANCQ68.com.meituan.access</string>
                        <string>FSS9ANCQ68.com.meituan.imeituan</string>
                </array>

                <key>com.apple.developer.pass-type-identifiers</key>
            <array>
                    <string>T3ZLXP4K5B.*</string>
            </array>

            <key>application-identifier</key>
            <string>FSS9ANCQ68.com.meituan.imeituan</string>

            <key>aps-environment</key>
            <string>production</string>

    </dict>
</plist><?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/Pro
<plist version="1.0">
        <dict>
                <key>keychain-access-groups</key>
                <array>
                        <string>FSS9ANCQ68.com.meituan.access</string>
                        <string>FSS9ANCQ68.com.meituan.imeituan</string>
                </array>

                <key>com.apple.developer.pass-type-identifiers</key>
            <array>
                    <string>T3ZLXP4K5B.*</string>
            </array>

            <key>application-identifier</key>
```

```
        <string>FSS9ANCQ68.com.meituan.imeituan</string>

        <key>aps-environment</key>
        <string>production</string>

    </dict>
</plist>
```

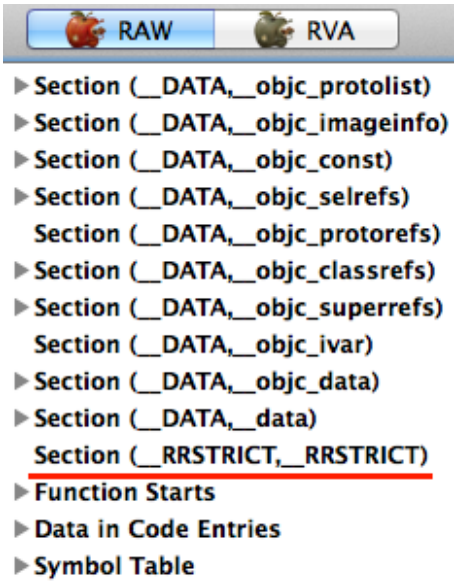
可以看到，imeituan这个可执行文件既没有setuid/setgid位，也没有特殊的entitlements，那么它含有__RESTRICT/__restrict这个section的可能性激增。至于如何验证我们的想法，用OSX逆向顶级大牛fG!的改造版MachOView看看就好了嘛~



好了，这应该就是美团4.8.1所采用的大杀器了，这种anti-DYLD_INSERT_LIBRARIES的方法，其实早在iOS 7完美越狱时，已经由GeoHot提到过了😱膜拜！！！

三、anti-anti-DYLD_INSERT_LIBRARIES

既然anti-DYLD_INSERT_LIBRARIES是由__RESTRICT/__restrict实现的，那么anti-anti-DYLD_INSERT_LIBRARIES自然就等同于anti-**RESTRICT/**restrict咯！而要anti-**RESTRICT/**restrict也很简单，把imeituan这个可执行文件用macvim等二进制编辑器打开，把所有的__RESTRICT/__restrict字符串给重命名一下就好了，比如：



这样dyld就找不到__RESTRICT/__restrict，也就不会忽略DYLD_INSERT_LIBRARIES了嘛！
把改过以后的imeituan拷贝回iOS，

```
snakeninnys-MacBook:~ snakeninny$ scp /Users/snakeninny/imeituan root@192.168.3.3
```

因为我们对App的可执行文件进行了静态patch，其md5值已经改变，所以Apple签名失效，正常情况下美团App无法启动。要解决这个问题，很方便，也很惭愧，因为要用到盗版App的利器——AppSync……
在Cydia中搜索AppSync，安装并respring后即可禁用iOS的签名校验。一切就绪后，打开美团，用Cycript重新测测看：

```
FunMaker-5s:~ root# cycript -p imeituan
cy# [UIApp displayIdentifier]
@"com.meituan.imeituan"
```

打完收工~

四、总结
根据我们的分析结果，来一个马后炮，我们可搜到Sam的一篇新博文（关键词“DYLD_INSERT_LIBRARIES __RESTRICT”），讲到的正是这个帖子所提到的内容。除了更专业更详细外，他还给出了怎么用这种机制给自己的App加上防护的方法，完爆我们这种只破坏不保护的猥琐行径:huffy:，值得崇拜~!

参考：
1. http://www.samdmarshall.com/blog/blocking_code_injection_on_ios_and_os_x.html
2. http://geohot.com/e7writeup.html
3. http://www.opensource.apple.com/source/dyld/dyld-210.2.3/src/dyld.cpp
4. https://theiphonewiki.com/wiki/Launchd.conf_untether

---

admin **Oct '14**

:grin: 非常好的文章

---

ithinco **Oct '14**

狗神威武！

liangweidarth **Oct '14**

与美团相比，更迷恋的是楼主“dylib的注入一般是通过DYLD_INSERT_LIBRARIES这个环境变量来实现的，现在dylib连注入都失败，即其constructor根本未得到执行，说明此行为不是由美团的代码完成的，而应该发生在代码执行前。既然这样的话，此行为多半是因MachO头部的某个标注，导致dyld有意为之的。”的这种分析思路，与楼主的分析思路相比，更迷恋的是楼主的头像:tongue:

11162 **Oct '14**

狗神威武阿，这片文章带出的知识点，和分析思路够我慢慢学习消化的了。能有如此的中文逆向论坛实在是件幸福的事情:biggrin:

11147 **Oct '14**

添加__RESTRICT Section,只能在工程中配置flags吗?还有没有其他途径呢?
例如说发布SDK,供其他工程用,但是又担心其他人员不添加这个flags!
有没有一种SDK中的配置,影响被用的App的途径,不需要主工程做其他修改就可以达到这个效果?
我尝试在子工程添加flags,但是并没有对主工程生成的APP产生效果,需要在主工程中配置才行.

ehll **Oct '14**

因为是在link的时候添加，所以当然只能在主工程里添加才有效

Maka **Oct '14**

我表示差距比去火星还要远:cry:

Robby **Oct '14**

固件6.0 版本iPhone4s, AppSync已经安装。下载美团，通过Mav0view 检查__RESTRICT Section也在。Cycript直接注入，根本不用任何其他设置。
snakeninny分析分析，这是中大奖了吗？:smile:

snakeninny **Oct '14**

不是，因为6.0的dyld还没有引入__RESTRICT检查~
参考链接4里也提到了，

Robby **Oct '14**

晕，就第4个参考文档没有看。谢谢解惑呀。

niumeng **Oct '14**

好文!
另外360手机卫士和腾讯手机管家有个支付保护功能，可以禁止其它App被tweak依附，能分析下吗?

---

snakeninny **Oct '14**

这我还是头一次听说，你逆向了吗？看看大概原理是什么？

---

krishno **Oct '14**

收藏

---

sagexy **Dec '14**

那么问题来了，怎么在工程里面添加__RESTRICT/__restrict这个section呢?

---

snakeninny **Dec '14**

请看参考链接1

---

sagexy **Dec '14**

sorry for my stupid

---

sagexy **Dec '14**

sorry for my stupid

PS:狗神头像越来越骚包了

SnowNight01 **Jan 15**

哎，现在美团可以了cycript，可以给一个不能的例子麽
root# cycript -p imeituan
cy#

---

snakeninny **Jan 21**

你自己写一个不能的例子试试看嘛