Роман @memkill
Пользователь

6 апреля 2012 в 10:19

Взлом одного Android приложения

Разработка под Android*

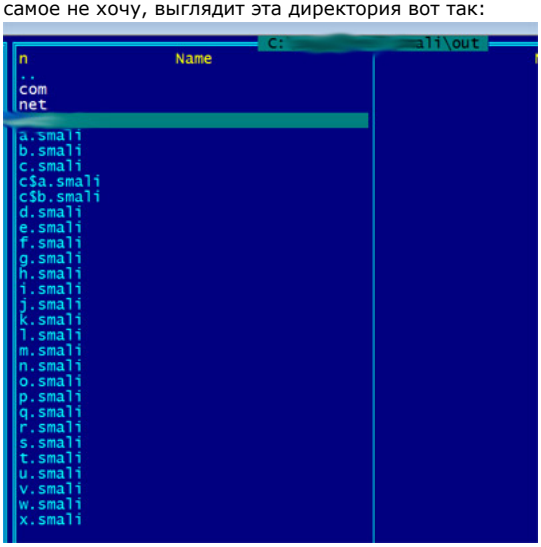
Недавно я усиленно разрабатывал свое приложение под Android, и в процессе защиты платной версии понял, что довольно сложно обезопас приложение от взлома. Ради спортивного интереса решил попробовать убрать рекламу из одного бесплатного приложения, в котором баннер предлагается скрыть, если заплатить денежку через In-App Purchase.

В этой статье я опишу, как мне удалось убрать рекламу бесплатно и в конце — несколько слов о том, как усложнить задачу взломщикам.

Шаг 1. Получаем «читаемый» код приложения.
~~Чтобы добыть APK приложения из телефона, нужны root права. Вытягиваем приложение из телефона с помощью adb (пусть, для конспирации нас будет приложение greatapp.apk):~~
~~`adb pull /data/app/greatapp.apk`~~
~~Хабраюзер @overmove подсказал мне, что root необязателен, можно с помощью Astro сделать бэкап любого приложения, и оно будет скопир в /mnt/sdcard.~~
Хабраюзер @MegaDiablo подсказал мне, что и Astro необязателен. Список установленных приложений и их файлы apk можно узнать через ут `pm` в шелле, а когда уже известно имя файла, его можно стянуть через `adb pull /data/app/app.filename.apk`.

APK — это ZIP архив, достаем оттуда интересующий нас файл `classes.dex` со скомпилированным кодом.
Будем использовать ассемблер/дизассемблер smali/baksmali для наших грязных дел.
`java -jar baksmali-1.3.2.jar classes.dex`

На выходе получаем директорию out с кучей файлов `*.smali`. Каждый из них соответствует файлу `.class`. Естественно, все обфусцированно самое не хочу, выглядит эта директория вот так:



Попытаемся понять, где в этой обфусцированной куче «говорится» о рекламе. Сначала я просто сделал поиск с текстом "AdView" (View, отображающий рекламу из AdMob SDK) по всем файлам. Нашелся сам `AdView.smali`, `R$id.smali` и некий `d.smali`. `AdView.smali` смотреть не интересно, `R.$id` я как-то сначала проигнорировал, и пошел сразу в таинственный `d.smali`.

Шаг 2. Пойти по неверному пути.
Вот и метод `a()` в файле `d.smali` с первым упоминанием `AdView` (я решил, скриншотом лучше, а то без форматирования это очень уныло чит

```
# virtual methods
.method public final declared-synchronized a()V
    .registers 5

    return-void  # return inserted by me.

    monitor-enter p0

    :try_start_1
    invoke-virtual {p0}, Ld;->e()Landroid/app/Activity;

    move-result-object v0

    if-nez v0, :cond_e

    const-string v0, "activity was null while trying to create an AdWebView."

    invoke-static {v0}, Lcom/google/ads/util/a;->a(Ljava/lang/String;)V
    :try_end_c
    .catchall {:try_start_1 .. :try_end_c} :catchall_3b
```

Метод ничего не возвращает, поэтому я, недолго думая, решил просто вставить поближе к началу `return-void`. Когда я все собрал и запусти
приложение радостно крэшнулось. Лог из `adb logcat`:

```
E/AndroidRuntime(14262): java.lang.RuntimeException: Unable to start activity ComponentInfo{com.greatapp/com.greatapp.GreatA
android.view.InflateException: Binary XML file line #22: Error inflating class com.google.ads.AdView
```

Понятно, что наш `AdView` в результате манипуляций должным образом не создался. Забудем пока про `d.smali`.

Шаг 3. Откатываем назад все изменения и смотрим на пропущенный ранее `R$id`. Вот и строчка с `AdView`:

```
# static fields
.field public static final adView:I = 0x7f080006
```

Похоже, это идентификатор View с рекламой. Поищем, где он используется, сделав поиск по значению `0x7f080006`. Получаем всего два резу
тот же `R$id` и `GreatApp.smali`. В GreatApp.smali текст уже гораздо интереснее (комментарии мои):

```
    .line 102
    invoke-virtual {p0}, Lcom.greatapp.GreatApp;->getApplicationContext()Landroid/content/Context;

    move-result-object v7

    const-string v8, "ad_free"

    invoke-static {v7, v8}, Lnet/robotmedia/billing/BillingController;->isPurchased(Landroid/conten

    move-result v7

    invoke-static {v7}, Ljava/lang/Boolean;->valueOf(Z)Ljava/lang/Boolean;

    move-result-object v6

    .line 104
    .local v6, purchased:Ljava/lang/Boolean;
    invoke-virtual {v6}, Ljava/lang/Boolean;->booleanValue()Z

    move-result v7

    if-eqz v7, :cond_32

    .line 105
    const v7, 0x7f080005

    invoke-virtual {p0, v7}, Lcom.greatapp.GreatApp;->findViewById(I)Landroid/view/View;

    move-result-object v0

    check-cast v0, Landroid/widget/LinearLayout;

    # Find AdView by id.
    .line 106
    .local v0, adContainer:Landroid/widget/LinearLayout;
    const v7, 0x7f080006

    invoke-virtual {p0, v7}, Lcom.greatapp.GreatApp;->findViewById(I)Landroid/view/View;

    move-result-object v1

    # Remove it from layout.
    .line 107
    .local v1, admobAds:Landroid/view/View;
    invoke-virtual {v0, v1}, Landroid/widget/LinearLayout;->removeView(Landroid/view/View;)V
```

Видно, что этот идентификатор используется для поиска `View` (строка 588) и буквально сразу же AdView удаляется с экрана (строка 595). Ви
удаляется, если пользователь заплатил за отсутствие рекламы? Если посмотреть немного выше, то взгляд цепляется за строчку 558 с «ключе
словами»:

```
invoke-static {v7, v8}, Lnet/robotmedia/billing/BillingController;->isPurchased(Landroid/content/Context;Ljava/lang/String;)
```

robotmedia — сторонняя (open source) библиотека, призванная упростить работу с in-app billing-ом в андроиде. Почему же она не была полно
обфусцирована? Ну да ладно, повезло.
Видно, что метод `isPurchased()` возвращает строку, которая с помощью `Boolean.valueOf()` преобразуется в объект Boolean и, наконец, в об
`boolean` через `booleanValue()`.
И тут самое интересное, в строке 572 мы переходим в некий `:cond_32`, если значение результата `== false`. А иначе начинается уже
просмотренный код поиска и удаления `AdView`.

Шаг 4. Минимальное изменение, собрать и запустить.

Что ж, дело за малым — удаляем эту ключевую строку, собираем приложение и сразу инсталлируем на телефон:

```
java -jar ..\smali\smali-1.3.2.jar ..\smali\out -o classes.dex
apkbuilder C:\devel\greatapp\greatapp_cracked.apk -u -z C:\devel\greatapp\greatapp_noclasses.apk -f C:\devel\greatapp\classe
jarsigner -verbose -keystore my-release-key.keystore -storepass testtest -keypass testtest greatapp_cracked.apk alias_name
adb install greatapp_cracked.apk
```

(`greatapp_noclasses.apk` — это оригинальный APK приложения, из которого удален classes.dex, сертификаты создаются с помощью Android
И ура, запускаем приложение, никакой рекламы!

Теперь о том, как усложнить задачу любителям халявы (это лишь то, что я запомнил из видео про пиратство с Google IO 2011, ссылка ниже)

- Не осуществлять проверку оплаты или лицензирования в классах Activity и особенно методах onCreate() и ему подобных. Эти «точки вх запускаются всегда в известное время и не обфусцируются, их всегда можно посмотреть и понять, что происходит с различными элеме UI
- Лучше всего проводить проверку не в основном потоке и в случайные моменты времени
- Проверять CRC файла `classes.dex`, причем хранить его зашифрованным
- Хранить код проверки лицензии или покупки скомпилированным и зашифрованным как ресурс приложения, динамически его загружать запускать через reflection

Надеюсь, было интересно. В заключение, несколько полезных ссылок по теме:

- Отличное видео с Google IO 2011 о том, как защитить приложение от пиратов.
- Небольшая статья с блога Android Developers с краткой подборкой техник защиты приложения от взлома, много повторяет предыдущее
- Статья на Хабре о реверс-инжиниринге будильника
- Dalvik VM bytecodes
- http://androidcracking.blogspot.com/, отличный блог, посвященный взлому приложений

smali, hacking, android, dex

Нужны ли нам нейронные сети?

+14 1,2k 10 2

Композиция или наследование: как выбрать?

+18 4,8k 64 12

0day в банковском ПО для геолокации

+43 4,2k 47 13

Процессоры Intel станут троичными

+86 37,9k 42 113

Обработка ошибок в C

+7 3,1k 61 11

Комментарии (42)

отслеживать новые: в почте в трекере

Umnik_ADS 6 апреля 2012 в 10:32

play.google.com/store/apps/details?id=com.bigtincan.android.adfree

ответить

Niemand 6 апреля 2012 в 11:26

Если приложение сделано грамотно, то оно не будет работать с AdFree, например как SSHDroid. Оно проверяет показ рекламы и при запуске предлагает ку полную версию, либо восстановить файл hosts.

ответить

Umnik_ADS 6 апреля 2012 в 11:27

Приложение, которые восстанавливает мой hosts без моего разрешения должно быть удалено как троянское.

ответить

Niemand 6 апреля 2012 в 11:59

вообще-то я написал, что оно предлагает это сделать (с целью включения адс и соответственно работы программы)

ответить

НЛО прилетело и опубликовало эту надпись здесь

Colobock 6 апреля 2012 в 21:51

Да не требует она ничего, а просто не работает с модифицированным hosts, прямо намекая: «не поправишь — не поеду».

ответить

Hoorsh 6 апреля 2012 в 12:01

Скоро его сольют в Google Play. Уже столько единиц из-за этого новшества наставили ему

ответить

Davidov 6 апреля 2012 в 12:26

Грустно, конечно. То есть хорошему приложению ставят низкий бал не за то, что оно плохо работает, а за то, что не получилось его «взломать» (хотя я предполагаю, что очень просто найти взломанную версию, если задаться целью).

Я понимаю, если бы оно стоило каких-то заоблачных денег; но 46 рублей, меньше полу процента от стоимости среднего android-телефона.

Что характерно, те, кто ставят 1 ещё что-то там про жадность пишут.

ответить

noxxx 6 апреля 2012 в 12:35

Не согласен. Это не грустно. Это ужасно, это катастрофа. Я бы еще понял если б приложение стоило от 500 рублей — это уже ощутимая сумма дл людей. Но 46 рублей!!! Если ты пользуешься каждый день приложением за 46 рублей и жмешь эту сумму — ты моральный урод. Я так считаю.
Если приложение которым я пользуюсь стоит меньше 100 рублей — покупаю сразу.
Но перед этим, конечно, или ставлю лайт версию, или ставлю пиратку :) Но тут уж не денешься ни куда.

ответить

diamant 16 апреля 2012 в 08:54

Это Андроид, здесь не принято платить.

ответить

Hoorsh 6 апреля 2012 в 14:03

Очень не люблю читать отзывы о моем бесплатном приложении. Обычно там самые гадкие комментарии. Одному не нравится то, что нравится дру
Делаешь нечто компромиссное — ставят колы в итоге оба. Гораздо приятнее развивать платную версию приложения. Обычно приобретают прилож
адекватные люди. Даже новые функции добавить просят, а не требуют. Парадокс.

ответить

de1337ed 6 апреля 2012 в 15:30

Да, тоже замечал, такое ощущение, что это два разных маркета (сам не разработчик, просто наблюдение пользователя)

ответить

Mairon 6 апреля 2012 в 16:18

Я просто эти комментарии минусую. Задолбали быдло-неадекваты, пишущие отзывы с сотней орфографических и пунктуационных ошибок, мог
миня нирапотайет автор казел.

ответить

iago 6 апреля 2012 в 17:22

(сам айфонщик) а что, на Google Play можно рейтинг выставлять комментам как на хабре?

ответить

Davidov 6 апреля 2012 в 17:51

Да, а ещё помечать как спам.

ответить

Hoorsh 6 апреля 2012 в 18:22

С недавних пор там можно комменты сортировать по рейтингу, фильтровать по устройствам. То есть хочешь видеть отзывы только по сво
аппарату — пожалуйста, хочешь, чтобы бесполезный спам не попадался на глаза — сортируй по рейтингу.

ответить

Paul 6 апреля 2012 в 18:39

Сам рейтинг в цифрах не виден, но можно отметить коммент как Helpful или Unhelpful и по этому Helpfulness сортировать.

ответить

and7ey 6 апреля 2012 в 20:30

А как это делается? Как проверяется, что реклама показывается? Что если подключения к интернету нет?

ответить

Hoorsh 8 апреля 2012 в 16:08

Видимо проверяется содержимое /etc/hosts и если там over9000 записей, прога решает, что установлен AdFree

ответить

twistedfall 6 апреля 2012 в 11:12

Для анализа кода андроидных приложений неплохо подходит связка dex2jar + jd-gui. Правда, собрать приложение после этого назад, скорее всего, не получитс

ответить

hellman 6 апреля 2012 в 19:08

Да, удобно смотреть код в jd-gui (после dex2jar) — можно локализовать нужное место, а патчить с backsmali/smali

ответить

bachin 6 апреля 2012 в 11:43

> Проверять CRC файла classes.dex, причем хранить его зашифрованным

Это возможно? (Я дилетант, но подозреваю, что приложение может не иметь возможности доступа к своему же байт-коду)

ответить

memkill 6 апреля 2012 в 12:05

Да, сам CRC естественно нужно хранить отдельно, в ресурсах или удаленно. Вот статейка на эту тему: http://androidcracking.blogspot.de/2011/06/anti-tamperir
crc-check.html

ответить

Vass 6 апреля 2012 в 11:53

1. Качаем приложение с баннерной рекламой (например Angry Birds)
2. Выключаем Wi-Fi и 3G
3. Запускаем приложение
4.…
5. PROFIT

Да для приложений которым нужна сеть (например ssh клиент этот метод не сработает)

ответить

baskos 6 апреля 2012 в 11:56

можно просто заблокировать приложение в Droidwall

ответить

silvansky 6 апреля 2012 в 11:58

Я так и делаю. Правда, в iOS.

ответить

memkill 6 апреля 2012 в 12:02

Да, с Angry Birds я тоже так делал :)
Но:
1) Несколько напрягает включать-выключать интернет
2) Приложению может быть нужен интернет для нормальной работы
3) Место для рекламы все равно будет съедаться, хотя зависит от приложения, в Angry Birds это не так.

ответить

webkumo 6 апреля 2012 в 12:58

1. Нисколько не напрягает. Не отключённый интернет скушивает батарею слишком быстро/грустно. Постоянные нотификации нисколько не радуют. Хотя , нужно постоянно быть в курсе почты/чего-либо ещё, то не вариант
2. У меня такое подозрение что действительно нужна сеть скорее рабочим приложениям, которые не грех и купить
3. Не видел таких

ответить

egormerkushev 6 апреля 2012 в 13:28

По третьему пункту — я как-то в одном бесплатном приложении предусмотрел такой вариант: если баннер не получается загрузить из сети, тогда вме просто показывается черный прямоугольник с надписью Advertisment (точнее, прямоугольник — это вьюха, в которую помещается баннер).
Гадкий я)

ответить

webkumo 6 апреля 2012 в 13:42

лучше бы написали «программирую за еду», а лучше какую-нибудь шутку/цитату… глядишь кто-нибудь прочувствовался бы и заплатил ;)

ответить

egormerkushev 6 апреля 2012 в 14:00

А неплохая мысль, кстати, цитаты показывать там…

ответить

Tweak 6 апреля 2012 в 12:01

etc/hosts

ответить

Dm4k 6 апреля 2012 в 12:11

Если у вас есть рут, то многие кастомные прошивки (либо установка соответствующей программы) позволяют редактировать разрешения для приложений — и можно просто запретить вылезать в интернет.
Или обрезать доступ в инет каким-либо фаерволом ;)
p.s. Естественно это применимо для приложений которым не нужен интернет для работы

ответить

НЛО прилетело и опубликовало эту надпись здесь

dlancer 6 апреля 2012 в 14:28

Если у вас есть рут и вы не разработчик, то в перспективе вы клиент антивирусной компании. Они вас ждут :)

ответить

Yakhnev 6 апреля 2012 в 12:14

Вставлю свои 5 копеек про упрощение получения исходного кода. AirDroid позволяет получить dex-файл без наличия root, не для всех приложений, но для большинства.

ответить

MegaDiablo 7 апреля 2012 в 01:10

Не хочу ни кого обидеть, но RTFM. Ребята у вас есть под рукой такая простая и замечательная утилита и имеет она название adb. По личному опыту могу с нет такого приложения которое нельзя было бы скачать с приложения и для этого не надо иметь root доступа.

ответить

Murevich 6 апреля 2012 в 12:14

Статья понравилась.
Любителям халявы:
если автор приложения сам не раздаёт его бесплатно и без рекламы, значит это ему (автору) зачем-то надо.

И, если пользователи будут игнорировать условия распространения приложения, определенные автором, скорее всего автор потеряет интерес к приложению и какое-то время забросит его. Или будет уделять меньше времени и усилий на разработку/обновления.

Вам это надо?

Нет денег, не хотите платить, не любите рекламу — не используйте такое приложение или (вместо того чтобы ломать защиту) напишите своё и раздайте други Поверьте, создавать что-то своё гораздо интереснее, чем копаться в чужом коде.

ответить

pravic 8 апреля 2012 в 06:13

Не согласен с последним утверждением. Иногда интереснее покопаться где-то (ессно, не в CD-ejector'е).

ответить

egormerkushev 6 апреля 2012 в 13:45

Столько стараний за 99¢ (обычно)!
Но для разработчика полезные знания. Спасибо.

ответить

silentnuke 6 апреля 2012 в 15:41

не проще было использовать dex2jar и Java Decompiler?)

ответить

maloii 6 апреля 2012 в 17:41

А мне кажется бороться с пиратами бесполезно. Тут все от политики монетизации зависит, если все правильно сделать то крякеры лишь помогают разработчик

ответить

Вы не можете комментировать эту публикацию

Можно комментировать публикации, которые не старше 10 дней, а также те, которые вы уже комментировали ранее.

ИНТЕРЕСНЫЕ ПУБЛИКАЦИИ

Как устроена инфраструктура интернета GT
665 10 1

Bash-скрипты: начало
1,5k 58 4

Нужны ли нам нейронные сети?
1,2k 10 2

Tesla Amazing — geek-стикеры от авторов из России, которые можно клеить куда угодно GT
957 1 3

Аутентификация OAuth2 в приложении посредством Google Sign-In. Непрерывный доступ к API Google
614 15 2

Arvifox
Профиль
Трекер
Настройки

Разделы
Публикации
Хабы
Компании
Пользователи
Песочница

Информация
О сайте
Правила
Помощь
Соглашение
Помощь стартапам

Услуги
Реклама
Тарифы
Контент
Семинары

Приложения
Загрузите в App Store
ДОСТУПНО В Google

Служба поддержки Мобильная версия