

2016

Как удалить рекламу из приложений APK Инструкция для начинающих

Реклама перестала быть чем то полезным, перестала нести в себе информативность, перестала помогать пользователям в выборе полезных приложений. Реклама стала монстром, ей "пугают" детей и взрослых, с её помощью выманивают деньги, она "жрёт" трафик, ресурсы и т.д. и т.п. Однако хочу заметить что не все приложения могут существовать и развиваться без рекламы. Поэтому если вы видите что приложение качественное, несёт в себе нужный вам функционал, поддержите разработчиков купив приложение или пожертвовав ту или иную сумму.



Оглавление

Введение	4
Инструменты для PC	5
Инструменты компиляции и декомпиляции APK.	5
Batch ApkTool.....	5
Apk manager.....	6
Вспомогательные инструменты	7
Total Commander	7
AkelPad	7
WinMerge	8
BlueStacks App Player.....	9
Инструменты для Android	10
Инструменты компиляции и декомпиляции APK.	10
APK Editor	10
Apktool на Андроид	11
Вспомогательные инструменты	12
LuckyPatcher	12
Total Commander for Android.....	13
QuickEdit.....	13
Структура APK.....	15
Декомпиляция APK	17
Компиляция APK.....	18
Удаление рекламы из приложений APK	19
AndroidManifest.xml	19
Редактирование ресурсов в папке RES.	22

Отключение рекламы через изменение кода в SMALI	25
Изменяем размер рекламного баннера	25
Нарушаем работу ads.service	29
Удаляем рекламные ссылки	29
Пример удаление рекламы из приложения Дурак_2.4.4.apk	30
Пример удаление рекламы из приложения TouchVPN_2.9.2.apk.....	32
Пример удаление рекламы из приложения Длинные нарды_4.60.apk.....	34

Введение

Данная инструкция написана для новичков. Которые боятся или не знают с чего начать.

А начать надо с подбора инструментов (софта). Вам нужна программа компилирования и декомпилирования APK, хороший текстовый редактор, программа умеющая искать файлы по содержимому, программа для сравнения файлов и папок, ну и конечно желание.



*Крякнуть может любой, жаль что учиться
крякать хотят не все.*

Для начала лучше попробовать свои силы на пользовательских APK, и уже потом поднабравшись опыта редактировать системные APK. Если вы вообще в первый раз обратились к этой теме, попробуйте для начала повторить редактирование APK из примеров этой инструкции.

Чтобы не было вопросов:

Почему программа стала вылетать? что-то не работает или работает не так как надо? и Т.Д и Т.П.?

Читаем предупреждение.

ВНИМАНИЕ: изменение программного кода может привести к ошибкам в работе APK, поэтому если не уверены что делаете, лучше этого не делать, особенно это касается системных приложений.

Поэтому всегда делайте бэкап приложений, а лучше всего испытывайте модифицированные APK на каких-нибудь эмуляторах Android'a, на подобии **BlueStacks App Player**.

Инструменты для PC

Для того чтобы редактировать APK приложения прежде всего их нужно декомпилировать (разобрать), а после скомпилировать (собрать). Для этого нужны инструменты которые помогут вам в работе. Далее будут приведены инструменты для PC (персональных компьютеров), но вы можете использовать любые другие аналогичные программы.

Инструменты компиляции и декомпиляции APK.

Batch ApkTool

Краткое описание:

Утилита для правильной перекомпиляции APK-файлов.

Описание:

Утилита для работы с файлами APK. Несмотря на свой скромный интерфейс, утилита превосходит большинство аналогичных приложений по качеству работы и удобству использования, поскольку она разрабатывается в тесном сотрудничестве с профессионалами по модификации Android-приложений, а также с головой и руками.

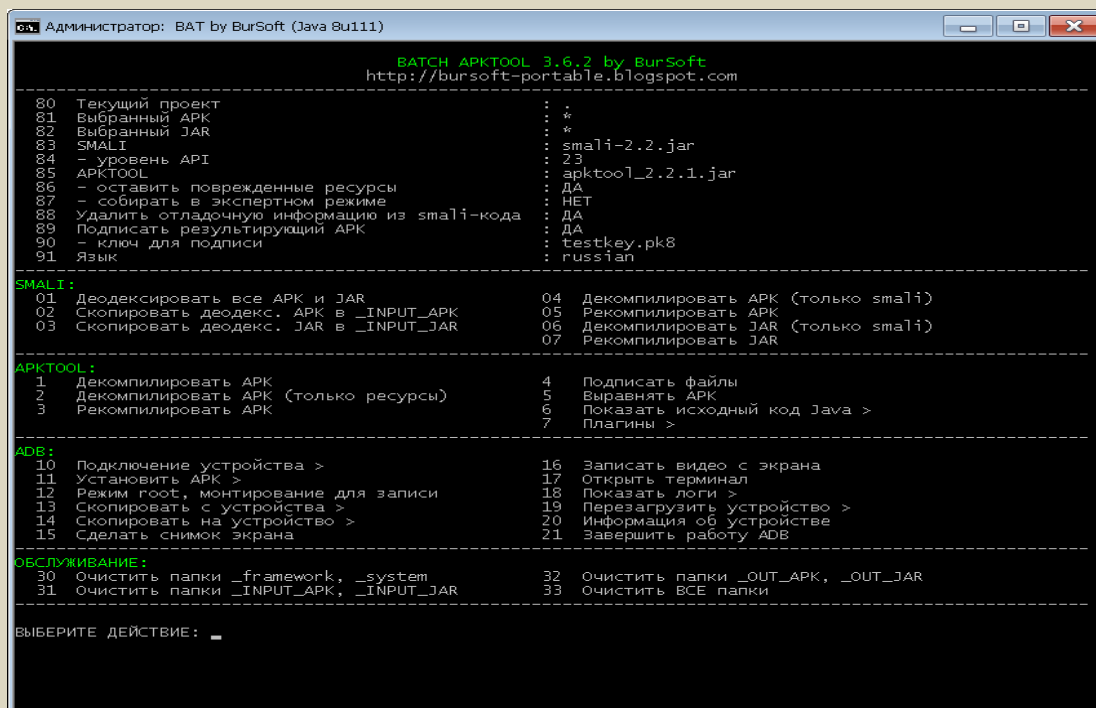


Рис. 1 - Интерфейс BatchApkTool

Русский интерфейс: Да

Разработчик: BurSoft

Домашняя страница: [BurSoft Portable - Batch ApkTool](http://bursoft-portable.blogspot.com)

Страница 4PDA: <http://4pda.ru/forum/index.php?showtopic=557858>

Apk manager

Краткое описание:

Программа для ПК, которая может распаковывать, компилировать, декомпилировать.

Описание:

Apk Manager может распаковывать, компилировать, декомпилировать и т.д. программы для Android. К сожалению, у программы нет как такового интерфейса, но это, по-моему, не является большим минусом.

```

C:\Windows\system32\cmd.exe

| Уровень сжатия: 9 | Размер хипа: 512Мб | Текущее приложение: 1.apk | API: 14
|-----|-----|
Простые действия                                     Полезные действия
|-----|-----|
0   Adb pull                                           32. Перекомпилировать приложение
1   Разархивировать приложение                         33. Переподписать приложение
2   Оптимизировать картинки                           41. Автом. путь odex в deodex (apk/jar)
3   Заархивировать приложение                         42. Свой путь odex в deodex (apk/jar)
4   Подписать приложение (Кроме системных)           55. Уровень API(odex): 14
5   Собрать/подписать приложение                     66. FrameworK.apk (Ресурсы)
6   Установить приложение                             88. Командная строка (Выход)
7   Архивировать/Подписать/Установить приложение
8   Adb push (Для системных)
|-----|-----|
Сложные действия (типа изменения кода)
|-----|-----|
9   Декомпилировать приложение                       70. Выбрать jar smali/baksmali
10  Декомпилировать приложение (С зависимостями)    (jar: Отсутствует)
11  Компилировать приложение                         71. Декомпилировать jar
12  Подписать приложение                             77. Компилировать jar
13  Установить приложение
14  Компилировать/Подписать/Установить приложение (Кроме системных)
99  Компилировать/Подписать приложение
|-----|-----|
Остальные действия
|-----|-----|
15  Оптимизировать приложение (в папке place-apk-here-to-batch-optimize)
16  Подписать приложение (в папке place-apk-here-for-signing)
17  Оптимизировать ogg файлы (в папке place-ogg-here)
18  Очистить файлы/папки
19  Выбрать уровень сжатия для приложений
20  Выбрать максимальный уровень памяти
21  Читать лог
22  Выбрать приложение
23  О приложении/Советы/Раздел отладки
24  Выход
|-----|-----|
Выберите действие: _

Apk Manager v 6.2
APKTOOL U 2.0.1
  
```

Рис. 2 - Интерфейс Apk manager

Русский интерфейс: Да

Разработчик: Неизвестен

Страница 4PDA: <http://4pda.ru/forum/index.php?showtopic=471803>

Вспомогательные инструменты

Total Commander

Описание: Total Commander (бывший Windows Commander) - это самый мощный и стабильный файловый менеджер для Windows с удобным пользовательским интерфейсом. Total Commander продолжает добрую традицию двухпанельных файловых менеджеров, но отличается повышенным удобством и функциональностью, поддерживает плагины, имеет множество встроенных полезных инструментов.

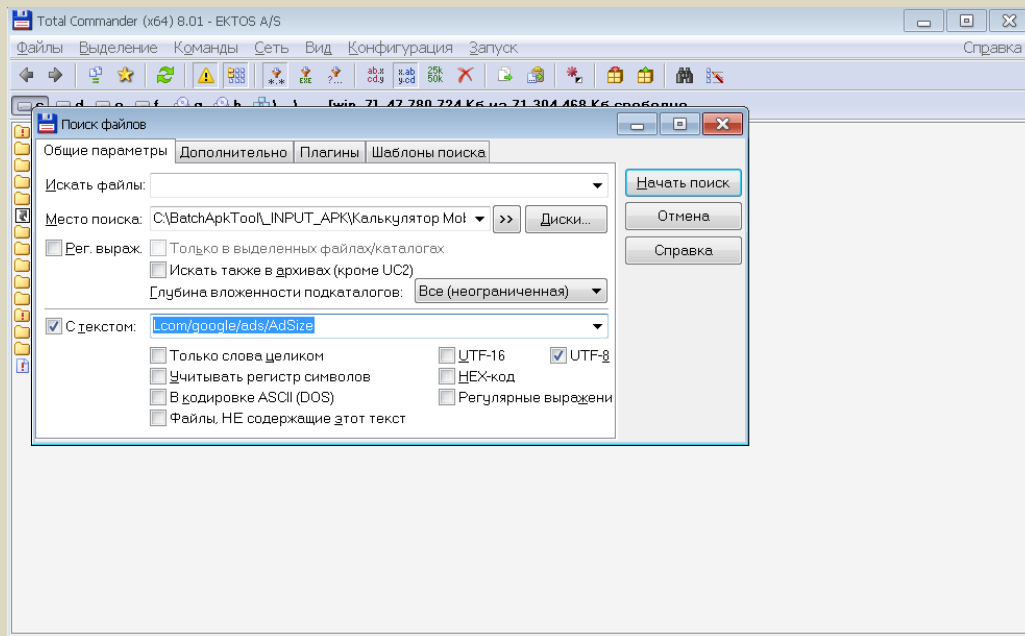


Рис. 3 - Поиск файла по содержимому в Total Commander

Нужный функционал: Есть возможность поиска файла по содержимому.

Скачать: <http://wincmd.ru/plugring/totalcmd.html>

AkelPad

AkelPad - текстовый редактор с открытым исходным кодом, созданный, чтобы быть маленьким и быстрым.

Возможности

- Однооконный (SDI), многооконный (MDI) и псевдо-многооконный режимы (PMDI);
- Полная поддержка Unicode строк на Unicode системах (NT/2000/XP/2003/Vista/Seven);
- Работа с кодировками Unicode (UTF-8, UTF-16LE, UTF-16BE, UTF-32LE, UTF-32BE);
- Работа с любой кодовой страницей, установленной в системе;
- Работа с DOS/Windows, Unix и Mac форматами перевода строки;
- Предпросмотр открытия файлов;
- Корректное отображение псевдографики;
- Блочное выделение текста;
- Многоуровневый откат действий;
- Быстрые поиск/замена строк текста;

- Запоминание кодировки и позиции каретки в файле;
- Печать и предпросмотр;
- Поддержка языковых модулей;
- Поддержка плагинов (подсветка синтаксиса, сворачивание блоков, авто-дополнение, запуск скриптов, клавиатурные макросы и много больше).

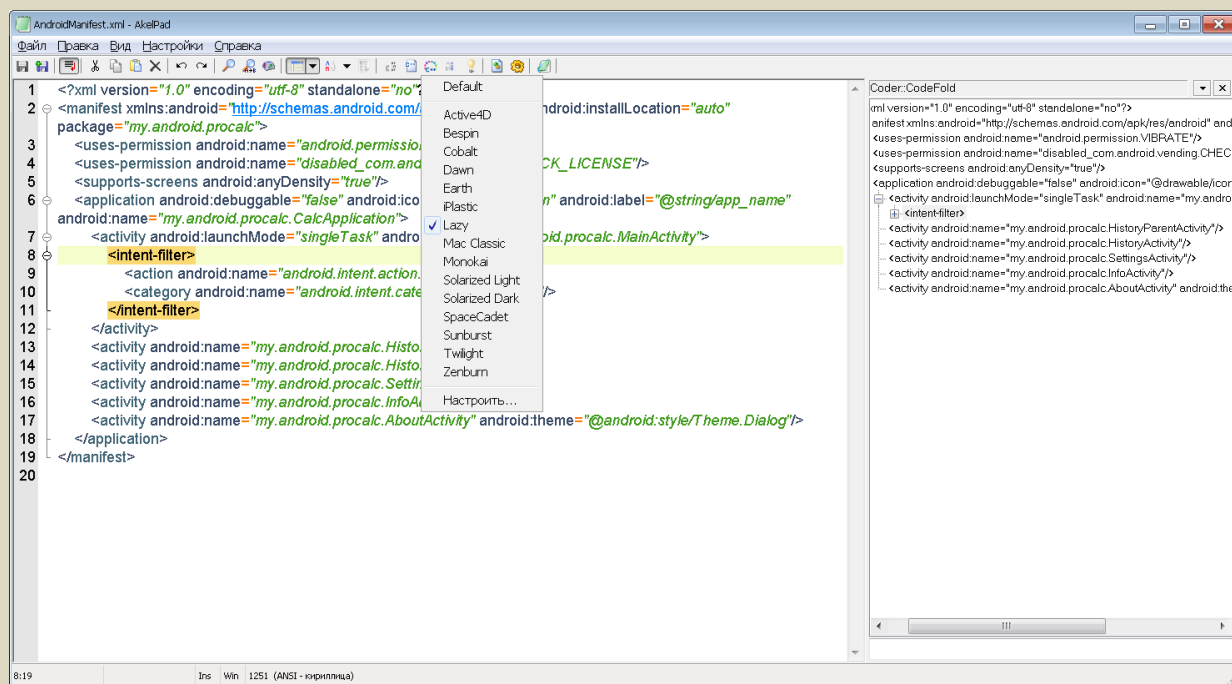


Рис. 4 - Интерфейс AkelPad

Русский интерфейс: Да

Разработчик: <http://akelpad.sourceforge.net/ru/index.php>

WinMerge

WinMerge является [Open Source](#) инструментом сравнения и слияния для Windows. WinMerge может сравнивать как файлы, так и папки, отображая различия в визуальной текстовой форме, которые легко понять и обработать.

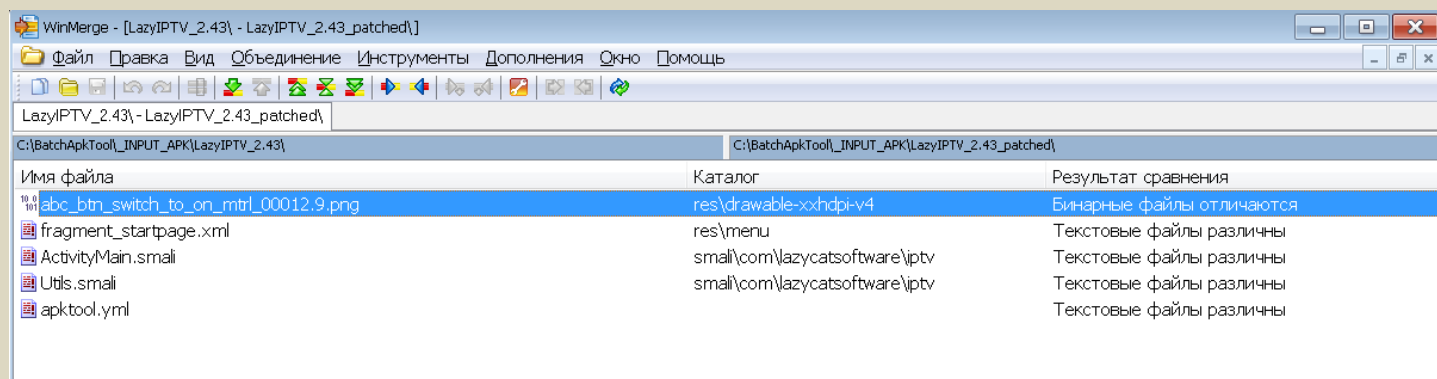


Рис. 5 - Поиск различий в папках

WinMerge является весьма полезной для определения мест, которые изменились между версиями проекта, а затем она позволяет объединять изменения между версиями. WinMerge можно использовать в качестве внешнего инструмента определения разностей/слияния, или как автономное приложение.

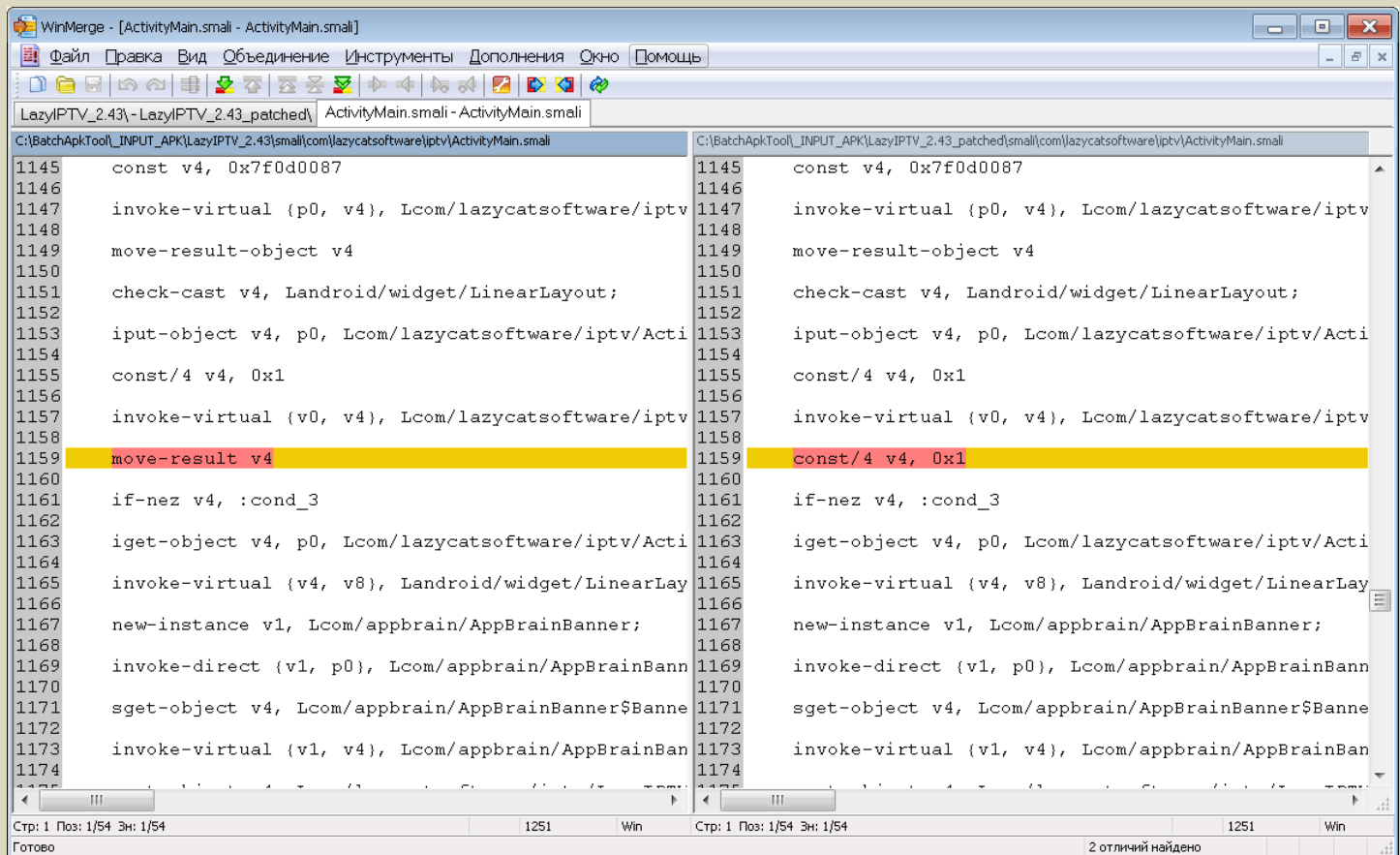


Рис. 6 - Поиск различий в файлах

Русский интерфейс: Да

Разработчик: <http://winmerge.org/about/?lang=ru>

BlueStacks App Player

Краткое описание: Полноценный эмулятор Android'a (почти). Если точнее, плеер приложений Android для PC

Описание: *BlueStacks App Player* - это специальная программа, использующая технологию под названием LayerCake, которая обеспечивает корректную среду для запуска ARM-приложений для ОС Android на компьютере с операционной системой Windows XP и выше.

Масштабы программы не так малы, как казалось бы с первого взгляда. После установки BlueStacks на ваш ПК с Windows **вы можете прямо из программы загрузить необходимое Android-приложение, без необходимости использовать Android-смартфон**. Выглядит BlueStacks довольно аккуратно и естественно вписывается в интерфейс Windows. Программа позволяет запустить Android-приложения в оконном или полноэкранном режиме, а разобраться с ним смогут почти все - оно переведено на 12 языков, в числе которых есть и русский.

Поддерживаемые ОС: Windows 10, Windows 8.1, Windows 8, Windows 7, Windows Vista, Windows XP, MacOSX

Русский интерфейс: Есть

Разработчик: BlueStacks

Домашняя страница: <http://bluestacks.com/>

Страница 4PDA: <http://4pda.ru/forum/index.php?showtopic=273806&st=320>

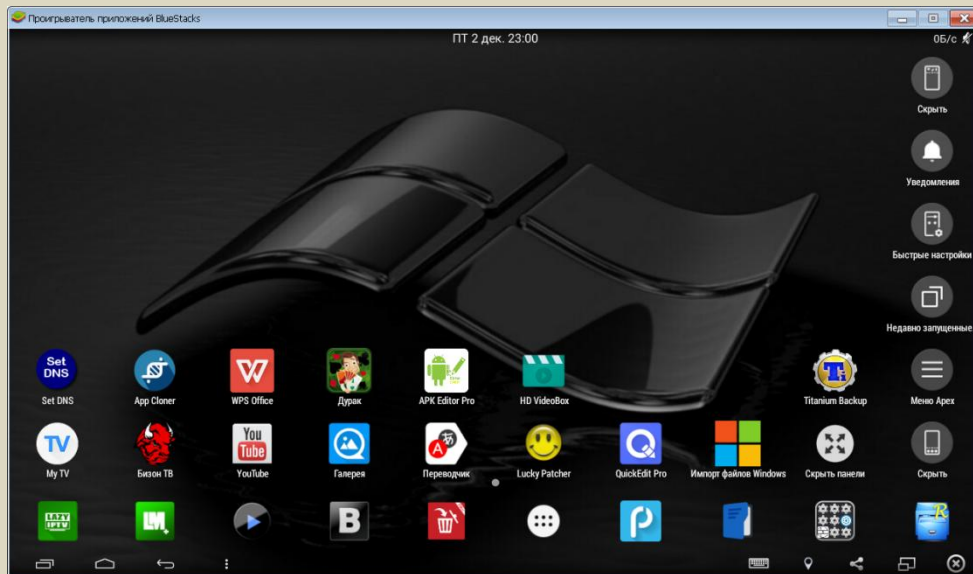


Рис. 7 - [BlueStacks App Player Rooted mod](#) от ajrys (4PDA)

Инструменты для Android

Для того чтобы редактировать APK приложения прежде всего их нужно декомпилировать (разобрать), а после скомпилировать (собрать). Для этого нужны инструменты которые помогут вам в работе. Далее будут приведены инструменты для Android, но вы можете использовать любые другие аналогичные программы.

Инструменты компиляции и декомпиляции APK.

APK Editor



Рис. 8 - APK Editor

Краткое описание:

Просмотр и редактирование арк-файлов

Описание:

APK Editor является мощным инструментом, который может редактировать/взламывать APK файлы, и поможет сделать с apk много интересного.

Это приложение поможет в таких операциях, как изменение строк локализации, замены фона изображений, изменения макета архитектуры, убрать рекламу, изменить разрешения и т.д. То, что оно может сделать, зависит от того, как вы используете это приложение.

В любом случае, что бы, что то сделать с APK файлом, нам необходимы некоторые знания и навыки.

Не бойтесь, на странице помощи приведены некоторые примеры.

Это про версия, по сравнению с бесплатной версией, есть некоторые отличия:

- (1) Нет никаких ограничений функционала
- (2) Отсутствие рекламы
- (3) Улучшенная поддержка

Требуется Android: 3.0 +

Русский интерфейс: Да

Разработчик: SteelWorks

Google Play: <https://play.google.co...l.heagoo.apkeditor.pro>

Страница 4PDA: <http://4pda.ru/forum/index.php?showtopic=575450>

Apktool на Андроид

Краткое описание:

Декомпиляция, рекомпиляция, подпись приложений.

Описание:

Программа выполняет различные операции с файлами .apk, .odex, .dex, .jar.

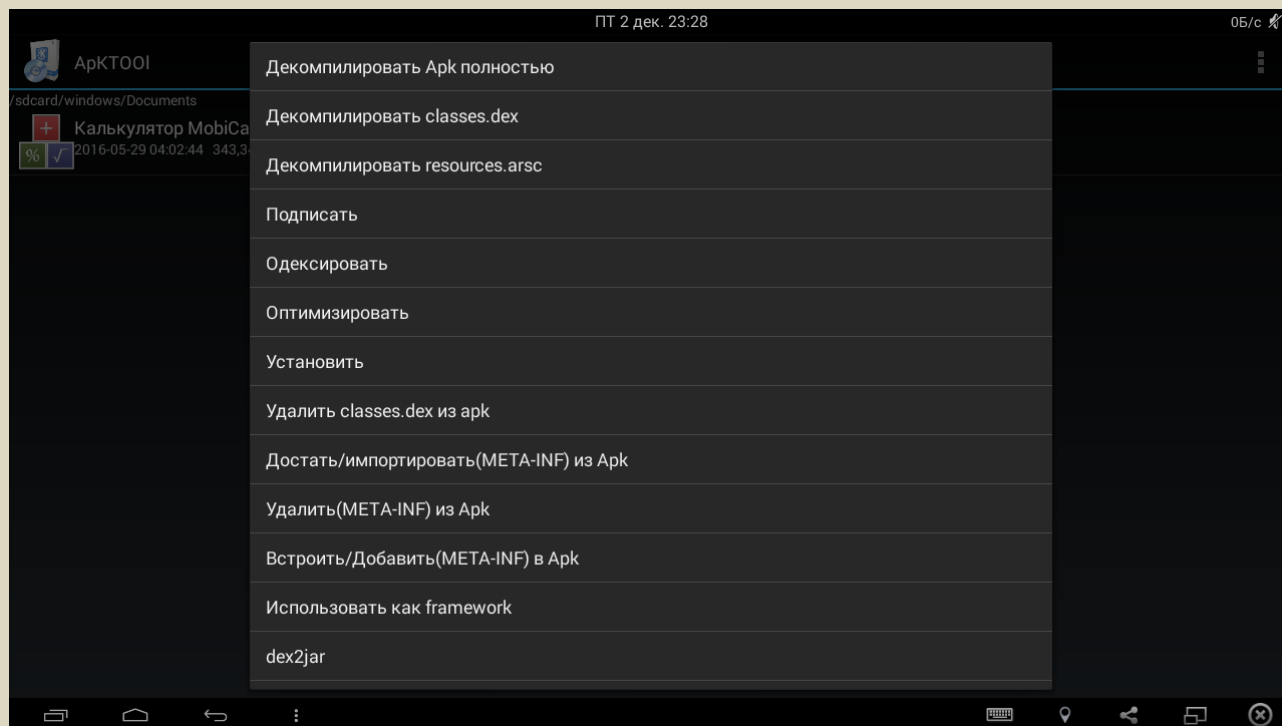


Рис. 9 - [Apktool 5.1.0 RC4 Rus Paint mod](#) от Belii81 (4PDA)

Для полноценного функционирования программы необходимы права root пользователя

Русский интерфейс: Нет (есть только в модах)

Страница 4PDA: <http://4pda.ru/forum/index.php?showtopic=482809>

Вспомогательные инструменты

LuckyPatcher

Программа прекрасно работает как с правами root пользователя так и без них, но с разным набором функционала.

Краткое описание:

Патчер к большинству программ и игр.

Описание:

Некоторые игры и программы имеют стандартный код взаимодействия с Google Market, но при этом очень сильные инструменты защиты программы от изменения.

Эта программа сканирует ваше устройство на все установленные программы, сортирует их на возможность и невозможность патча и размещает приложения в которых патч может быть полезен в начале списка. Вы можете применить к ней патч, и если он сработает, тогда вы будете иметь полностью зарегистрированную программу.

В крайних версиях добавлена возможность блокирования назойливой рекламы в приложениях, но не везде и всегда срабатывает. Перед применением патча рекомендуется сделать бэкап приложения!

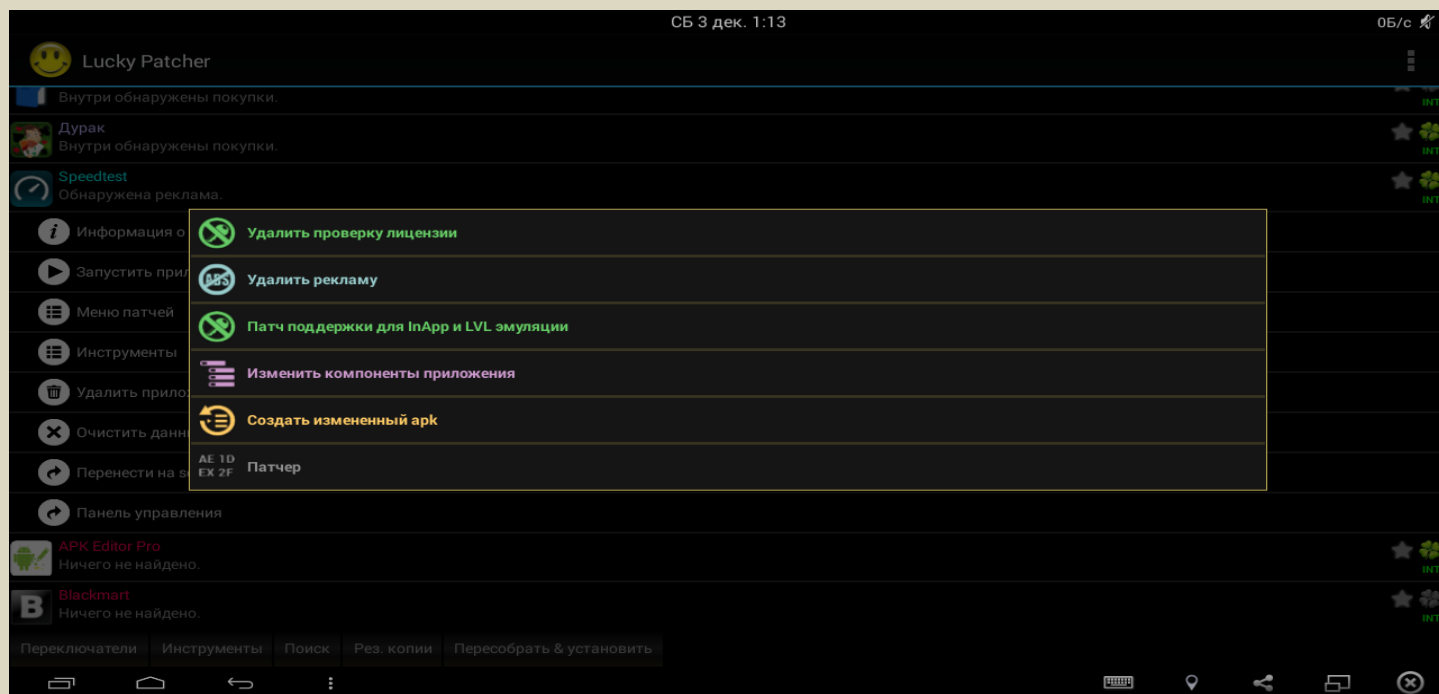


Рис. 10 - Меню патчей в LuckyPatcher

Русский интерфейс: Да

Требуется Андроид версии: 1.6+

Разработчик: [Chelpa](http://chelpa.ru)

Домашняя страница: http://dimonvideo.ru/forum/topic_1728149047/

Страница 4PDA: <http://4pda.ru/forum/index.php?showtopic=298302>

Total Commander for Android

Нужный функционал: Есть возможность поиска файла по содержимому.

Описание:

Двухпанельный файловый менеджер для Android с русским интерфейсом.

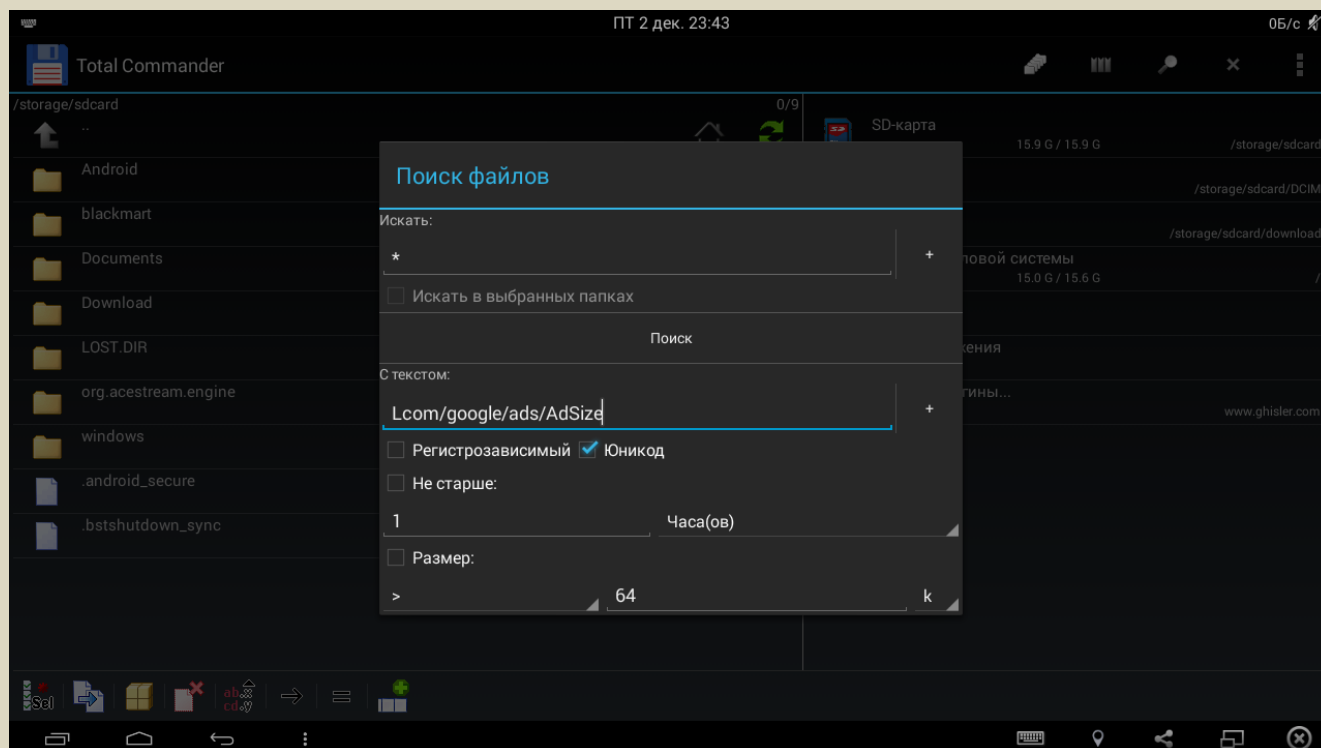


Рис. 11 - Поиск файлов в Total Commander

Русский интерфейс: Да

Поддерживаемая версия Android: 1.5+

Разработчик: Christian Ghisler

Домашняя страница: <http://ghisler.com>

Google Play: <https://play.google.com/store/apps/details?....TotalCommander>

Страница 4PDA: <http://4pda.ru/forum/index.php?showtopic=224555>

QuickEdit

Краткое описание:

Быстрый, стабильный и полнофункциональный текстовый редактор.

Описание:

- * Расширенное приложение блокнот с многочисленными улучшениями.
- * Редактор кода и подсветка синтаксиса для более чем 40 различных языков программирования.
- * Высокая производительность и отклик в реальном времени для больших текстовых файлов (более 10 000 строк).
- * Показывает и скрывает номера линий.
- * Отмена действия и повтор изменений без ограничений.

- * Показ, увеличение и уменьшение отступов строк.
- * Быстрый выбор и редактирование.
- * Плавная прокрутка в вертикальном и горизонтальном направлении.
- * Непосредственно переходите на указанный номер строки.
- * Быстро находите и заменяйте.
- * Легко вводить цвет в значениях hex.
- * Автоматически определять кодировку символов.
- * Меняйте кодировку текста.
- * Открывайте файл из списка последних открытых и добавленных файлов.
- * Поддержка светлой и темной тем.
- * Оптимизирован для телефонов и планшетов.



Рис. 12 - Редактирование файлов в QuickEdit

Русский интерфейс: Да

Разработчик: Rhythm Software

Домашняя страница: <http://forum.xda-developers.com/android/apps-games/app-quickedit-text-editor-t2899385>

Google Play:

Free: <https://play.google.com/store/apps/details?id=com.rhmsoft.edit>

Pro: <https://play.google.com/store/apps/details?id=com.rhmsoft.edit.pro>

Страница 4PDA: <http://4pda.ru/forum/index.php?showtopic=625901>

Структура APK

APK (Android Package) — формат архивных исполняемых файлов-приложений для Андроид. Каждое приложение Android скомпилировано и упаковано в один файл, который включает в себя весь код приложения (.DEX файлы), ресурсы, активы и файл .manifest. Файл приложения может иметь любое имя, но расширение должно быть .APK.

Файлы с данным расширением хранятся в магазине Google Play, и загружаются с его помощью в телефон или планшет для их использования, либо устанавливаются пользователем вручную на устройстве.

Каждый **.APK** файл — это сжатый архив для исполнения в DalvikVM (виртуальная машина), который может быть установлен не только на операционной системе Android.

Приложения которые вы устанавливаете на Android можно открыть обыкновенным архиватором, для мелких и несложных редакций приложений, а также декомпиляция для сложных и полных изменений приложений apk (в дальнейшем apk).

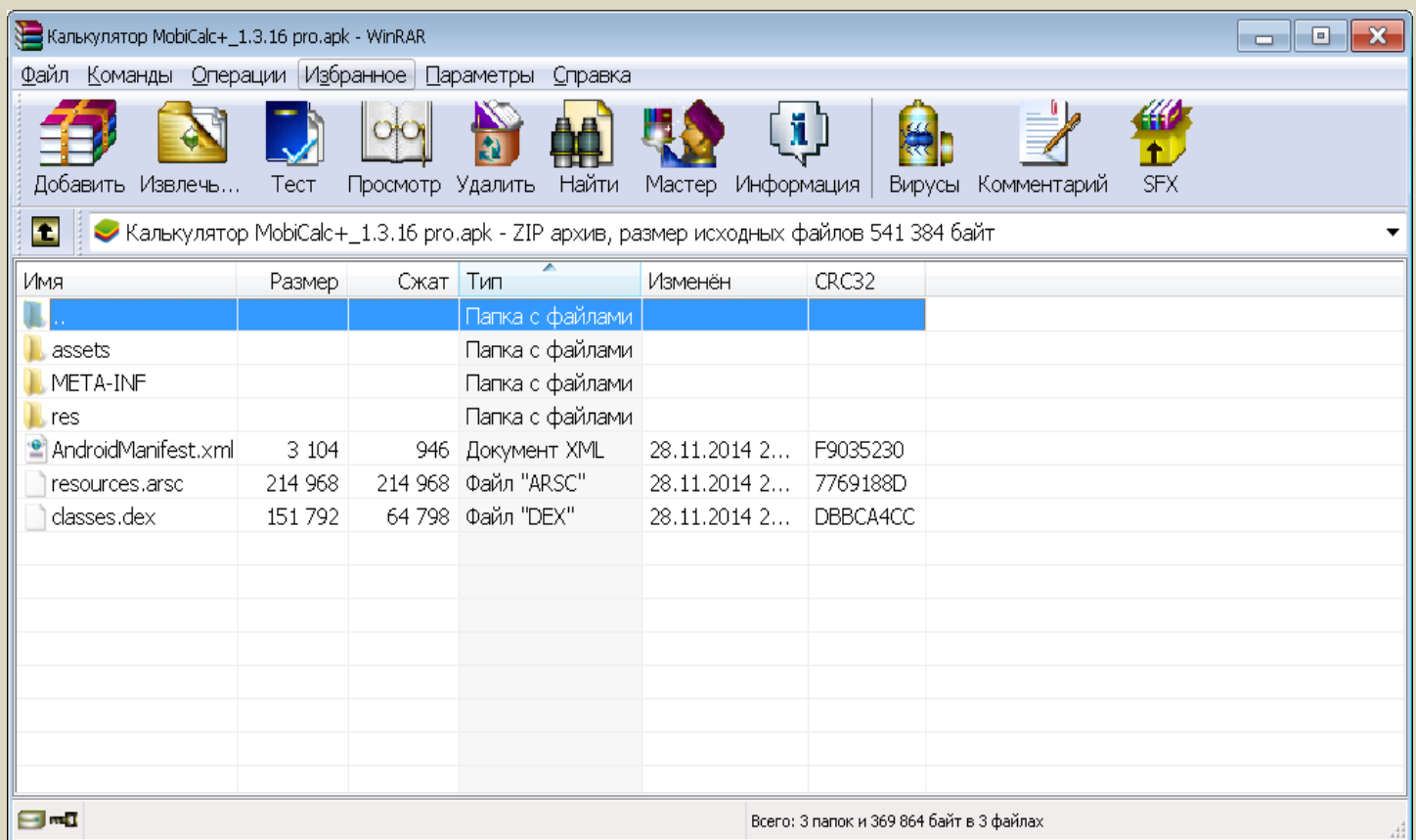


Рис. 13 - Структура компилированного APK (WinRAR)

Папка Assets — активы в данной папки находятся файлы которые задают «возможности» приложения.

Папка LIB — в данной папке находятся дополнительные библиотеки, которые обслуживают работу приложения

Папка META-INF — в данной папке находятся файлы подписи. У каждой компании, которая выкладывает приложения, есть своя подпись, которая свидетельствует, что это его собственность, так как имеет свой ключ шифрования.

Каталог META-INF содержит:

CERT.RSA — сертификат приложения

CERT.SF — контрольные суммы файлов ресурсов (картинок, звуков и т.д.)

MANIFEST.MF — служебная информация, описывающая сам apk-файл

Папка RES — в данной папке находится видимая, слышимая часть приложения. В данной папке содержатся файлы, которые позволяют строить интерфейс, картинки и мелодии.

Файл AndroidManifest.xml — паспорт apk приложения. Данный файл содержит общие сведения о приложении.

Файл classes.dex — скомпилированная «рабочая часть» приложения (**Папка декомпилированного APK - SMALI**)

Файл resources.arsc — скомпилированная «видимая часть» приложения (**Папка декомпилированного APK - RES**)

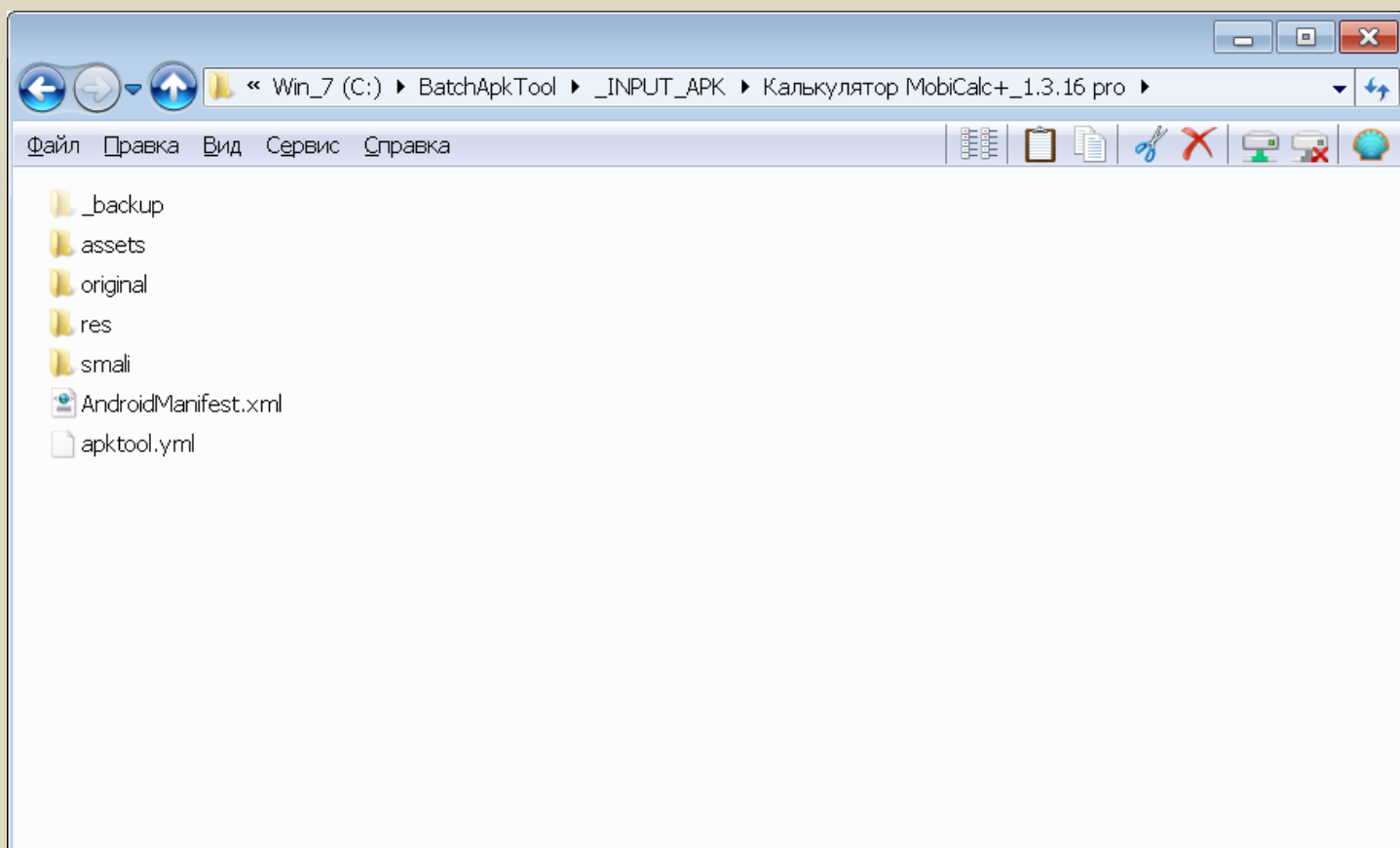


Рис. 14 - Структура декомпилированного APK (BatchApkTool)

Декомпиляция APK

Для того чтобы редактировать APK приложения прежде всего их нужно декомпилировать (разобрать). Далее будет приведен пример декомпиляции APK с помощью программы BatchApk Tool.

Скачиваем (ссылка см. Инструменты для PC) и распаковываем программу BatchApk Tool на компьютер.

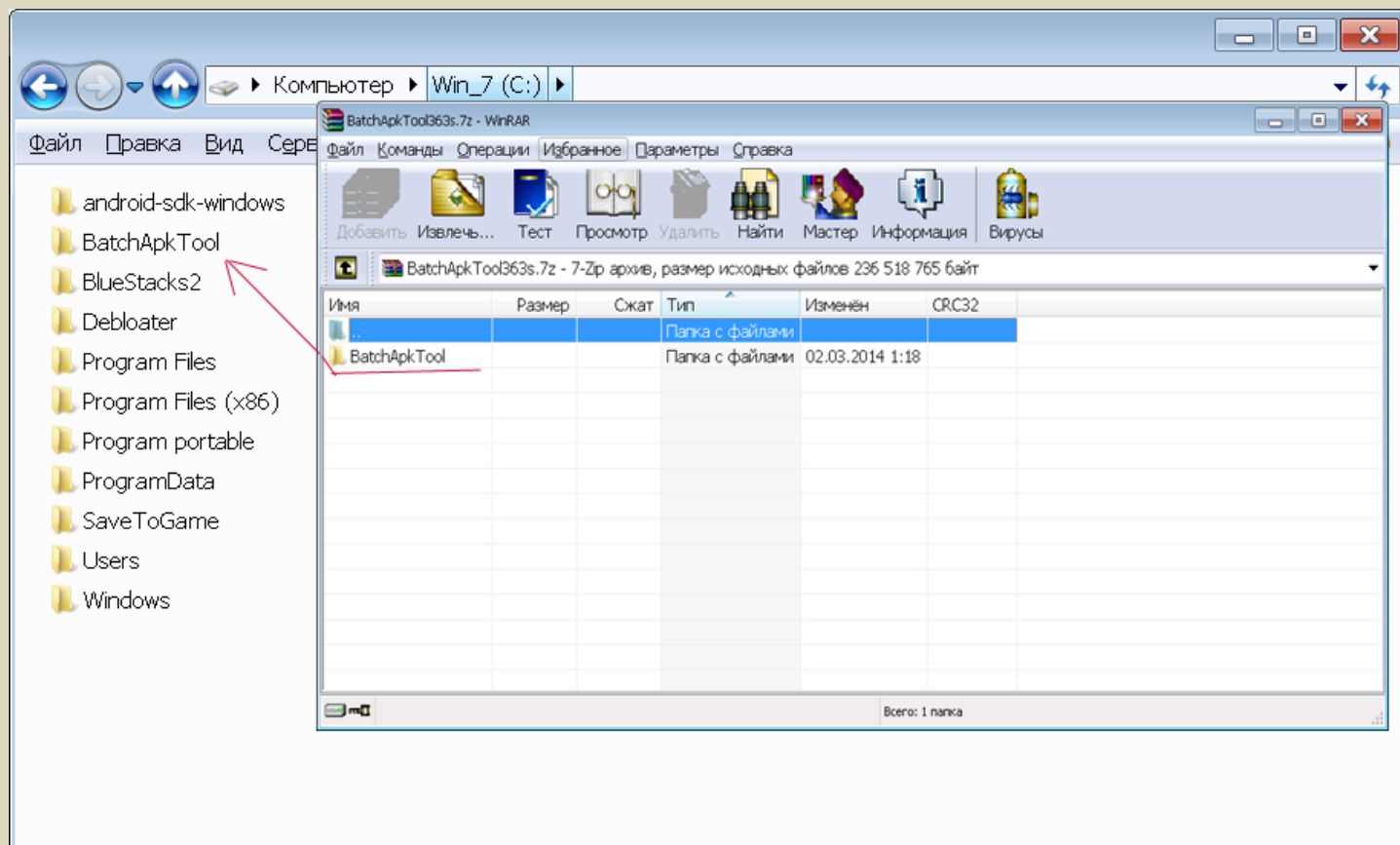


Рис. 15 - Установка BatchApk Tool на компьютер

Запускаем BatchApkTool.exe и выбираем нужный язык

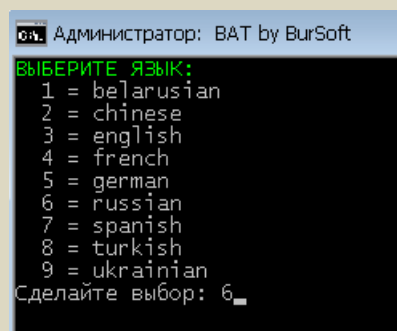


Рис. 16 - Выбор языка в BatchApk Tool

После этого открывается окно интерфейса программы. (см. Рис 1)

Далее нам нужно скопировать APK который мы хотим декомпилировать в папку _INPUT_APK. Путь по умолчанию: C:\BatchApkTool_INPUT_APK

Внимание: для избегания ошибок компиляции и декомпиляции "Путь по умолчанию" не должен состоять из кириллических названий. Пример: C:\Мои программы\BatchApkTool_INPUT_APK. APK который мы хотим декомпилировать лучше переименовать покороче: например Дурак_com.gobrainfitness.durak_2.4.4_19.apk переименуйте в 1.apk

Далее возвращаемся в интерфейс программы и в поле **ВЫБЕРИТЕ ДЕЙСТВИЕ: 1** вводим 1 и нажимаем Enter.

После этого запускается процесс декомпиляции **ВЫБЕРИТЕ ДЕЙСТВИЕ: 1**
[*] Декомпиляция 1.apk...

По завершении процесс вы увидите **ГОТОВО**
Нажмите ENTER для продолжения или введите 1, чтобы открыть лог:

Всё, на этом процесс декомпиляции закончен. В папке _INPUT_APK появиться новая папка с декомпилированными файлами, имя папки будет такое же как у APK который мы декомпилировали, в данном примере это 1.

BatchApkTool можно закрыть. Из папки _INPUT_APK можно удалить 1.apk он больше не нужен. Папку с декомпилированными файлами лучше заархивировать, на тот случай если нужно восстановить оригинальные файлы. Структура декомпилированных файлов будет примерно такой как показано на Рис. 14.

После декомпиляции можно приступить к редактированию файлов. После редактирования надо собрать (компилировать APK).

Компиляция APK

Запускаем BatchApkTool.exe (если вы его закрыли). В поле **ВЫБЕРИТЕ ДЕЙСТВИЕ: 3** вводим 3 и нажимаем Enter.

Убедитесь чтобы была включена настройка подписи результирующего APK.

89 Подписать результирующий APK : ДА

В этом случае APK будет собран и подписан одновременно. Если настройка отключена то APK будет скомпилирован без подписи. Подписать APK можно будет позже, выбрав 4.

По завершении процесс вы увидите

ВЫБЕРИТЕ ДЕЙСТВИЕ: 3
[*] Рекомпиляция 1...
ГОТОВО
Нажмите ENTER для продолжения или введите 1, чтобы открыть лог: _

Скомпилированный APK будет находится в папке _OUT_APK
Путь по умолчанию: C:\BatchApkTool_OUT_APK

Удаление рекламы из приложений APK

Удалить рекламу из APK приложений можно тремя способами:

1. Отредактировать AndroidManifest.xml - запретить, поломать, удалить activity с рекламой (это самый простой способ)
2. Сделать рекламный баннер равным нулю, для этого надо отредактировать файлы отвечающие за вывод и размер рекламного баннера в интерфейс программы (папка с ресурсами - RES)
3. Изменение программного кода «рабочей части» приложения, папка SMALI (это самый сложный способ)

AndroidManifest.xml

Файл манифеста AndroidManifest.xml предоставляет основную информацию о программе системе. Каждое приложение должно иметь свой файл AndroidManifest.xml. Редактировать файл манифеста можно вручную, изменяя XML-код через любой тестовый редактор, который позволяет осуществлять визуальное и текстовое редактирование файла.

Элемент **<activity>** объявляет активность. Если активность не объявлена в манифесте, она не будет видна системе и не будет запущена при выполнении приложения или будет выводиться сообщение об ошибке. Этим мы и воспользуемся для отключения рекламы.

1. Ищем activity с рекламой.

Первым признаком рекламы является присутствие в activity ключевых слов: **ads, adview, AdActivity, Banner**.

Пример:

```
<activity android:name="com.heyzap.sdk.ads.HeyzapProxyActivity"/>
<activity android:name="com.applovin.advview.AppLovinInterstitialActivity"/>
<activity android:name="com.smaato.soma.video.VAST AdActivity"/>
<activity android:name="com.smaato.soma.ExpandedBannerActivity"/>
```

*Может встретиться вот такая строка с **Banner***. Обычно эти строки идут сразу после объявления разрешений (**<uses-permission android:name="Имя_разрешения"/>**)

```
<application android:banner="@drawable/banner" android:icon="@drawable/ic_launcher"
android:label="@string/app_name".....>
или
<activity android:banner="@drawable/banner" android:hardwareAccelerated="false"
android:label="@string/app_name".....>
```

Её удалять нельзя, можно только удалить из этой строки **android:banner="@drawable/banner"** остальное оставить как есть.

Элемент **<application>** один из основных элементов манифеста, содержащий описание компонентов приложения, доступных в пакете: стили, значок и др. В манифесте может быть только один элемент **<application>**

И так мы определили откуда исходит реклама это **com.heyzap**, **com.applovin**, **com.smaato**. Существует большая вероятность того, что все строки с этими адресами являются поставщиками рекламы, хоть они и не содержат ключевых слов: **ads**, **adview**, **AdActivity**, **Banner**. Значит их тоже нужно отключить или удалить.

Пример:

```
<activity android:name="com.smaato.soma.interstitial.InterstitialActivity"/>
```

Кроме activity можно также отключить и receiver содержащий **ads**, **adview**, **AdActivity**. Элемент **<receiver>** объявляет приёмник широковещательных намерений как один из компонентов приложения. Приёмники широковещательных намерений дают возможность приложениям получить намерения, которые переданы системой или другими приложениями, даже когда другие компоненты приложения не работают.

Пример:

```
<receiver android:name="com.heyzap.sdk.ads.PackageAddedReceiver">
```

Существуют **activity** на рекламу в которых нет ключевых слов: **ads**, **adview**, **AdActivity**, **Banner**. В этом случае только опыт может подсказать что это реклама. Вот несколько примеров таких activity:

```
<activity android:name="com.startapp.android.publish.list3d.List3DActivity"
android:theme="@android:style/Theme"/>
```

```
<activity android:configChanges="keyboardHidden|orientation|screenSize"
android:excludeFromRecents="true" android:hardwareAccelerated="true"
android:name="com.chartboost.sdk.CBImpressionActivity"
android:theme="@android:style/Theme.Translucent.NoTitleBar.Fullscreen"/>
```

```
<activity android:name="com.colortv.android.ui.ContentRecommendationActivity"
android:screenOrientation="landscape" android:theme="@android:style/Theme.Translucent.NoTitleBar"/>
```

```
<activity android:configChanges="keyboardHidden|orientation|screenSize"
android:name="com.mopub.mobileads.MoPubActivity"/>
```

Также реклама может быть с facebook. Если после отключений описанных выше, реклама все еще присутствует, отключите activity для facebook. В своей основе это не реклама, а выскакивающие окна связанные с facebook.

```
<activity android:configChanges="keyboard|keyboardHidden|orientation|screenLayout|screenSize"
android:label="@string/app_name" android:name="com.facebook.FacebookActivity"
android:theme="@android:style/Theme.Translucent.NoTitleBar"/>
```

2. Способы отключение permission, activity и receiver.

После того как вы определились с поставщиками рекламы их нужно отключить.

Первый способ это просто полностью **удалить строку** с **permission**, **activity** или **receiver**.

Второй способ это добавление ключа **disabled_**.

Третий способ запрет на запуск с помощью **android:enabled="false"**

Примеры:

1. Самый простой способ отключить рекламу это запретить приложению выход в INTERNET. Для этого надо удалить соответствующее разрешение **permission**.

Удаляем строку:

```
<uses-permission android:name="android.permission.INTERNET"/>
```

Тоже самое относится к **activity**. Удаляем строки:

```
<activity android:name="com.heyzap.sdk.ads.HeyzapProxyActivity"/>
<activity android:name="com.applovin.adview.AppLovinInterstitialActivity"/>
<activity android:name="com.smaato.soma.video.VASTAdActivity"/>
<activity android:name="com.smaato.soma.ExpandedBannerActivity"/>
<activity
android:configChanges="locale|fontScale|keyboard|keyboardHidden|mcc|mnc|navigation|orientation|screenLayout|screenSize|smallestScreenSize|touchscreen|uiMode" android:hardwareAccelerated="true"
android:name="com.unity3d.ads.adunit.AdUnitActivity"
android:theme="@android:style/Theme.NoTitleBar.Fullscreen"/>
```

2. Второй способ это добавление ключа **disabled_**

Изменяем строку:

```
<uses-permission android:name="disabled_android.permission.INTERNET"/>
```

Тоже самое относится к **activity**. Изменяем строки:

```
<activity android:name="disabled_com.heyzap.sdk.ads.HeyzapProxyActivity"/>
<activity android:name="disabled_com.applovin.adview.AppLovinInterstitialActivity"/>
<activity android:name="disabled_com.smaato.soma.video.VASTAdActivity"/>
<activity android:name="disabled_com.smaato.soma.ExpandedBannerActivity"/>
<activity
android:configChanges="locale|fontScale|keyboard|keyboardHidden|mcc|mnc|navigation|orientation|screenLayout|screenSize|smallestScreenSize|touchscreen|uiMode" android:hardwareAccelerated="true"
android:name="disabled_com.unity3d.ads.adunit.AdUnitActivity"
android:theme="@android:style/Theme.NoTitleBar.Fullscreen"/>
```

3. Третий способ запрет на запуск с помощью **android:enabled="false"**

Изменяем строки:

```
<activity android:enabled="false" android:name="com.heyzap.sdk.ads.HeyzapProxyActivity"/>
<activity android:enabled="false" android:name="com.applovin.adview.AppLovinInterstitialActivity"/>
<activity android:enabled="false" android:name="com.smaato.soma.video.VASTAdActivity"/>
<activity android:enabled="false" android:name="com.smaato.soma.ExpandedBannerActivity"/>
```

```
<activity
android:configChanges="locale|fontScale|keyboard|keyboardHidden|mcc|mnc|navigation|orientation|screenLayout|screenSize|smallestScreenSize|touchscreen|uiMode" android:enabled="false"
android:hardwareAccelerated="true" android:name="com.unity3d.ads.adunit.AdUnitActivity"
android:theme="@android:style/Theme.NoTitleBar.Fullscreen"/>
```

Тоже самое относится к **receiver**. Изменяем строку

```
<receiver android:enabled="false" android:name="com.heyzap.sdk.ads.PackageAddedReceiver">
```

Редактирование ресурсов в папке RES.

Основные ресурсы Android-приложения:

тип ресурса	Размещение	Описание
Цвета	res/values/имя_файла	Идентификатор цвета, указывающий на цветовой код. ID таких ресурсов выражаются в R.java как R.color.* . XML-узел: /resources/color
Строки	res/values/имя_файла	Строковые ресурсы. В их число также входят строки в формате java и html. ID таких ресурсов выражаются в R.java как R.string.* . XML-узел: resources/string. Можно использовать дополнительное форматирование при помощи стандартных html-тегов , <i> и <u>. Методы, которые будут обрабатывать строковые ресурсы с HTML-форматированием, должны уметь обрабатывать эти теги.
Меню	/res/menu/имя_файла	Меню в приложении можно задать как XML-ресурсы.
Параметры	/res/values/имя_файла	Представляет собой параметры или размеры различных элементов. Поддерживает пиксели, дюймы, миллиметры, не зависящие от плотности экрана пиксели (dip) и пиксели, не зависящие от масштаба. ID таких ресурсов выражаются в R.java как R.dimen.* . XML-узел: resources/dimen
Изображения	/res/drawable/ваши_файлы	Ресурсы-изображения. Поддерживает форматы JPG, GIF, PNG (самый предпочтительный) и др. Каждое изображение является отдельным файлом и получает собственный идентификатор, который формируется по имени файла без расширения. Такие ID ресурсов представлены в файле R.java как R.drawable.*. Система также поддерживает так называемые растягиваемые изображения (stretchable image), в которых можно менять масштаб отдельных элементов, а другие элементы оставлять без изменений.
Отрисовываемые цвета	/res/values/ваш_файл или /res/drawable/ваши_файлы	Представляет цветные прямоугольники, которые используются в качестве фона основных отрисовываемых объектов, например точечных рисунков. Поддержка такой функции обеспечивается тегом значения drawable, находящимся в подкаталоге значений. Такие id ресурсов выражаются в файле R.java как R.drawable.*. XML-узел для такого файла: /resources/drawable. В Android при помощи специальных XML-файлов, расположенных в

		/res/drawable, также поддерживаются скругленные и градиентные прямоугольники. Корневым XML-тегом для drawable является <shape>. Идентификаторы таких ресурсов выражаются в файле R.java как R.drawable.*. В таком случае, каждое имя файла преобразуется в уникальный id отрисовываемого объекта
Анимация	/res/anim/ваш_файл	Android может выполнить простую анимацию на графике или на серии графических изображений. Анимация включает вращения, постепенное изменение, перемещение и протяжение.
Произвольные XML-файлы	/res/xml/*.xml	В Android в качестве ресурсов могут использоваться произвольные XML-файлы. Они компилируются в aapt. Идентификаторы таких ресурсов также выражаются в файле R.java как R.xml.*
Произвольные необработанные ресурсы	/res/raw/*.*	Любые некомпиллированные двоичные или текстовые файлы, например, видео. Каждый файл получает уникальный id ресурса. Идентификаторы таких ресурсов выражаются в файле R.java как R.raw.*
Произвольные необработанные активы	/assets/*.*/.*.*	Можно использовать произвольные файлы в произвольно названных каталогах, которые находятся в подкаталоге /assets. Это не ресурсы, а просто необработанные файлы. В этом каталоге, в отличие от /res, подкаталоги могут располагаться на любой глубине. Для таких файлов не создаются идентификаторы ресурсов. При работе с ними нужно использовать относительное имя пути, начиная с /assets, но не указывая этого каталога в имени пути

Почти любой рекламный баннер выведенный в интерфейсе программы имеет свой размер и место где он должен быть показан. Следующим способом удаления рекламы будет нахождение и изменения размера рекламного баннера.

Точно так же как и в AndroidManifest.xml, первым признаком является присутствие ключевых слов: **ads**, **advview**, **Banner**. Т.к в AndroidManifest.xml мы определили поставщиков рекламы то можно искать и по ним: **com.heyzap**, **com.applovin**, **com.smaato**, **com.startapp**, **com.chartboost**, **com.colortv**, **com.mopub** и т.д.

В основном вывод рекламы встречается в папках **...\res\layout** и **...\res\values**.

Суть метода заключается в том чтобы найти строки вывода баннера на экран и сделать его размер равным нулю **"0.0dip"**. Или запретить вывод на экран при помощи **android:visible="false"**

android:layout_width="24.0dip" android:layout_height="140.0dip"

меняем на

android:layout_width="0.0dip" android:layout_height="0.0dip"

или

```
android:layout_width="fill_parent" android:layout_height="wrap_content"
```

меняем на

```
android:layout_width="0.0dip" android:layout_height="0.0dip"
```

Где:

- android:layout_width - ширина элемента.
- android:layout_height - высота элемента

Атрибут может принимать следующие значения:

- fill_parent — элемент имеет размер такой же как и родительский элемент
- wrap_content — элемент имеет размер по своему содержимому .

Точное значение размера указываться в следующих единицах:

- dp - (Density-independent Pixels) - абстрактная единица, основанная на плотности экрана.

Пример:

```
<item android:icon="?iconABPromo" android:id="@id/im_promo" android:title="@string/bar_new"
app:showAsAction="always" />
```

добавляем **android:visible="false"**

```
<item android:icon="?iconABPromo" android:id="@id/im_promo" android:visible="false"
android:title="@string/bar_new" app:showAsAction="always" />
```

или

```
<com.mopub.mobileads.MoPubView android:id="@id/adView" android:layout_width="fill_parent"
android:layout_height="wrap_content" />
```

меняем "fill_parent" и "wrap_content" на "0.0dip", получается так:

```
<com.mopub.mobileads.MoPubView android:id="@id/adView" android:layout_width="0.0dip"
android:layout_height="0.0dip" />
```

или

```
<dimen name="bannerHeight">43.0dip</dimen>
```

меняем на "0.0dip", получается так:

```
<dimen name="bannerHeight">0.0dip</dimen>
```

Но не всегда можно изменить размер рекламного баннера в папке с ресурсами (RES)

В таком случае придется искать и отключать рекламу через изменение программного кода «рабочей части» приложения, папка SMALI

Отключение рекламы через изменение кода в SMALI

При декомпиляции файла **classes.dex** получаем папку **smali** с кучей файлов ***.smali**

Файлы ***.smali** являются **классами** (что то типа шаблона для создания объекта). В данном случае объектом является реклама, а вот ее размер и другие параметры будут являться классами. Обычно авторы приложений создают специальные классы для вывода рекламы и вызывают методы этих классов во время запуска приложения.

Изменяем размер рекламного баннера

Идем в каталог **...\smali\com\google\ads**. Внутри находим кучу файлов с расширением **smali**. Это классы, и наиболее примечателен из них класс **AdSize.smali**, по названию нетрудно догадаться, что именно в нем задается размер рекламы.

Открываем **AdSize.smali** любым текстовым редактором и ищем **# direct methods**

Код:

```
# direct methods
.method static constructor <clinit>()V
    .locals 4

    new-instance v0, Lcom/google/ads/AdSize;

    const/4 v1, -0x1 меняем на 0x0

    const/4 v2, -0x2 меняем на 0x0

    const-string v3, "mb"

    invoke-direct {v0, v1, v2, v3}, Lcom/google/ads/AdSize;-><init>(IILjava/lang/String;)V

    sput-object v0, Lcom/google/ads/AdSize;:>SMART_BANNER:Lcom/google/ads/AdSize;

    new-instance v0, Lcom/google/ads/AdSize;

    const/16 v1, 0x140 меняем на 0x0

    const/16 v2, 0x32 меняем на 0x0

    const-string v3, "mb"

    invoke-direct {v0, v1, v2, v3}, Lcom/google/ads/AdSize;-><init>(IILjava/lang/String;)V

    sput-object v0, Lcom/google/ads/AdSize;:>BANNER:Lcom/google/ads/AdSize;

    new-instance v0, Lcom/google/ads/AdSize;
```

```
const/16 v1, 0x12c меняем на 0x0
const/16 v2, 0xfa меняем на 0x0
const-string v3, "as"
invoke-direct {v0, v1, v2, v3}, Lcom/google/ads/AdSize;-><init>(ILjava/lang/String;)V
sput-object v0, Lcom/google/ads/AdSize;->IAB_MRECT:Lcom/google/ads/AdSize;
new-instance v0, Lcom/google/ads/AdSize;
const/16 v1, 0x1d4 меняем на 0x0
const/16 v2, 0x3c меняем на 0x0
const-string v3, "as"
invoke-direct {v0, v1, v2, v3}, Lcom/google/ads/AdSize;-><init>(ILjava/lang/String;)V
sput-object v0, Lcom/google/ads/AdSize;->IAB_BANNER:Lcom/google/ads/AdSize;
new-instance v0, Lcom/google/ads/AdSize;
const/16 v1, 0x2d8 меняем на 0x0
const/16 v2, 0x5a меняем на 0x0
const-string v3, "as"
invoke-direct {v0, v1, v2, v3}, Lcom/google/ads/AdSize;-><init>(ILjava/lang/String;)V
sput-object v0, Lcom/google/ads/AdSize;->IAB_LEADERBOARD:Lcom/google/ads/AdSize;
new-instance v0, Lcom/google/ads/AdSize;
const/16 v1, 0xa0 меняем на 0x0
const/16 v2, 0x258 меняем на 0x0
const-string v3, "as"
invoke-direct {v0, v1, v2, v3}, Lcom/google/ads/AdSize;-><init>(ILjava/lang/String;)V
sput-object v0, Lcom/google/ads/AdSize;->IAB_WIDE_SKYSCRAPER:Lcom/google/ads/AdSize;

return-void
.end method
```

Далее в папке **smali** ищем, есть ли ещё файлы **AdSize.smali**.

Находим ещё файл **AdSize.smali** по пути...\\smali\\com\\google\\android\\gms\\ads

Открываем AdSize.smali любы текстовым редактором и ищем # direct methods

Код:

```
# direct methods
.method static constructor <clinit>()V
    .locals 6

    const/16 v5, 0x140 меняем на 0x0

    const/4 v4, -0x3 меняем на 0x0

    new-instance v0, Lcom/google/android/gms/ads/AdSize;

    const/16 v1, 0x32 меняем на 0x0

    const-string v2, "320x50_mb" меняем на 0x0

    invoke-direct {v0, v5, v1, v2}, Lcom/google/android/gms/ads/AdSize;-><init>(IILjava/lang/String;)V

    sput-object v0, Lcom/google/android/gms/ads/AdSize;->BANNER:Lcom/google/android/gms/ads/AdSize;

    new-instance v0, Lcom/google/android/gms/ads/AdSize;

    const/16 v1, 0x1d4 меняем на 0x0

    const/16 v2, 0x3c меняем на 0x0

    const-string v3, "468x60_as"

    invoke-direct {v0, v1, v2, v3}, Lcom/google/android/gms/ads/AdSize;-><init>(IILjava/lang/String;)V

    sput-object v0, Lcom/google/android/gms/ads/AdSize;->FULL_BANNER:Lcom/google/android/gms/ads/AdSize;

    new-instance v0, Lcom/google/android/gms/ads/AdSize;

    const/16 v1, 0x64 меняем на 0x0

    const-string v2, "320x100_as"

    invoke-direct {v0, v5, v1, v2}, Lcom/google/android/gms/ads/AdSize;-><init>(IILjava/lang/String;)V

    sput-object v0, Lcom/google/android/gms/ads/AdSize;->LARGE_BANNER:Lcom/google/android/gms/ads/AdSize;

    new-instance v0, Lcom/google/android/gms/ads/AdSize;

    const/16 v1, 0x2d8 меняем на 0x0

    const/16 v2, 0x5a меняем на 0x0
```

```
const-string v3, "728x90_as"

invoke-direct {v0, v1, v2, v3}, Lcom/google/android/gms/ads/AdSize;-><init>(IILjava/lang/String;)V

sput-object v0, Lcom/google/android/gms/ads/AdSize;-
>LEADERBOARD:Lcom/google/android/gms/ads/AdSize;

new-instance v0, Lcom/google/android/gms/ads/AdSize;

const/16 v1, 0x12c меняем на 0x0

const/16 v2, 0xfa меняем на 0x0

const-string v3, "300x250_as"

invoke-direct {v0, v1, v2, v3}, Lcom/google/android/gms/ads/AdSize;-><init>(IILjava/lang/String;)V

sput-object v0, Lcom/google/android/gms/ads/AdSize;-
>MEDIUM_RECTANGLE:Lcom/google/android/gms/ads/AdSize;

new-instance v0, Lcom/google/android/gms/ads/AdSize;

const/16 v1, 0xa0 меняем на 0x0

const/16 v2, 0x258 меняем на 0x0

const-string v3, "160x600_as"

invoke-direct {v0, v1, v2, v3}, Lcom/google/android/gms/ads/AdSize;-><init>(IILjava/lang/String;)V

sput-object v0, Lcom/google/android/gms/ads/AdSize;-
>WIDE_SKYSCRAPER:Lcom/google/android/gms/ads/AdSize;

new-instance v0, Lcom/google/android/gms/ads/AdSize;

const/4 v1, -0x1 меняем на 0x0

const/4 v2, -0x2 меняем на 0x0

const-string v3, "smart_banner"

invoke-direct {v0, v1, v2, v3}, Lcom/google/android/gms/ads/AdSize;-><init>(IILjava/lang/String;)V

sput-object v0, Lcom/google/android/gms/ads/AdSize;-
>SMART_BANNER:Lcom/google/android/gms/ads/AdSize;

new-instance v0, Lcom/google/android/gms/ads/AdSize;

const/4 v1, -0x4 меняем на 0x0

const-string v2, "fluid"
```

```
invoke-direct { v0, v4, v1, v2 }, Lcom/google/android/gms/ads/AdSize;-><init>(IILjava/lang/String;)V
sput-object v0, Lcom/google/android/gms/ads/AdSize;->FLUID:Lcom/google/android/gms/ads/AdSize;
new-instance v0, Lcom/google/android/gms/ads/AdSize;
const/4 v1, 0x0
const-string v2, "search_v2"
invoke-direct { v0, v4, v1, v2 }, Lcom/google/android/gms/ads/AdSize;-><init>(IILjava/lang/String;)V
sput-object v0, Lcom/google/android/gms/ads/AdSize;->SEARCH:Lcom/google/android/gms/ads/AdSize;
return-void
.end method
```

Нарушаем работу ads.service

Ищем в папке smali все файлы содержащие:

```
"com.google.android.gms.ads.identifier.service.START"
меняем на:
"com.google.android.gms.ads.identifier.service.STAPT"
```

Ищем в папке smali все файлы содержащие:

```
"com.google.android.gms.ads.service.START"
меняем на:
"com.google.android.gms.ads.service.STAPT"
```

Удаляем рекламные ссылки

Еще одним способом отключения показа рекламы является удаление рекламных ссылок. Сначала ищем рекламные ссылки в папке ...\\smali\\com. Если после удаления реклама осталась то нужно просмотреть всю папку ...\\smali

Тут можно пойти двумя путями:

1. Искать все файлы содержащие "<https://>" и "<http://>"
2. Если определили откуда исходит реклама в AndroidManifest.xml, в данном случае мы определили что реклама исходит из: com.heyzap, com.applovin, com.smaato. Только "com" надо перенести в конец и искать: heyzap.com, applovin.com, smaato.com

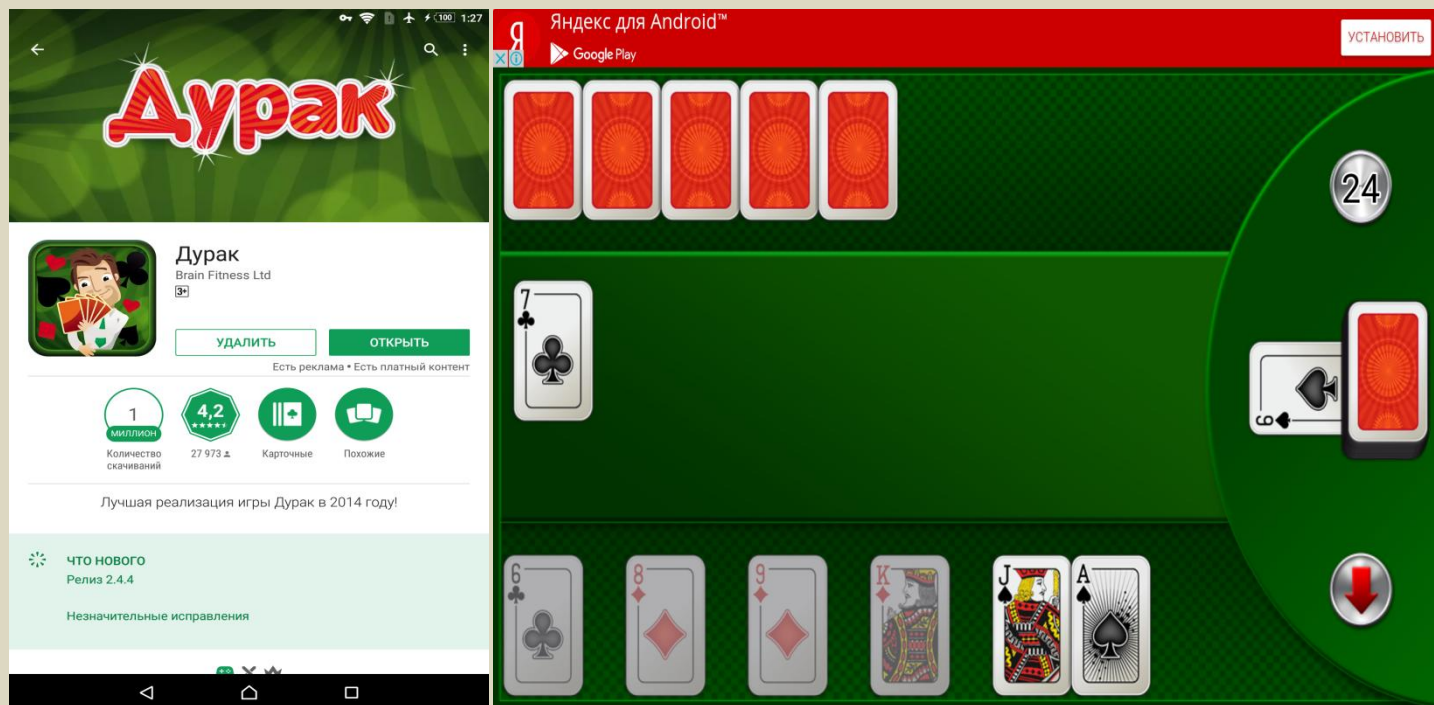
Пример удаления ссылок:

```
"http://ads.heyzap.com/in\_game\_api/ads"
"http://schemas.applovin.com/android/1.0"
"http://www.smaato.com"
```

Для удаления можно заменить все ссылки на: "="

Пример удаление рекламы из приложения

Дурак_2.4.4.apk



Краткое описание:

Реализация игры Дурак от создателей преферанса Марьяж. Мы сделали все возможное, чтобы играть было интересно и удобно!

Русский интерфейс: Да

Google Play: <https://play.google.com/store/apps/details?id=com.gobrainfitness.durak>

Страница 4PDA: <http://4pda.ru/forum/index.php?showtopic=576226>

1. Ищем и отключаем activity с рекламой в AndroidManifest.xml

```
<activity
android:configChanges="keyboard|keyboardHidden|orientation|screenLayout|screenSize|smallestScreenSize|ui
Mode" android:enabled="false" android:name="com.google.android.gms.ads.AdActivity"
android:theme="@android:style/Theme.Translucent"/>
...
<activity android:enabled="false"
android:name="com.google.android.gms.ads.purchase.InAppPurchaseActivity"
android:theme="@style/Theme.IAPTheme"/>
```

После отключения activity с рекламой на месте рекламы остается предупреждение об ошибке активации: **Missing AdActivity with android:configChanges in AndroidManifest.xml.**



2. Изменяем размер рекламного баннера

Для того чтобы убрать эту надпись (красным в баннере рекламы), нужно изменить размер рекламного баннера в файлах:

...\smali\com\google\ads\AdSize.smali

...\smali\com\google\android\gms\ads\AdSize.smali

Ищем

direct methods

.method static constructor <clinit>()V

...Далее "шерстим" весь метод в этих двух файлах и меняем размер на 0x0



Все рекламы больше нет.

Пример удаление рекламы из приложения

TouchVPN_2.9.2.apk

Краткое описание:

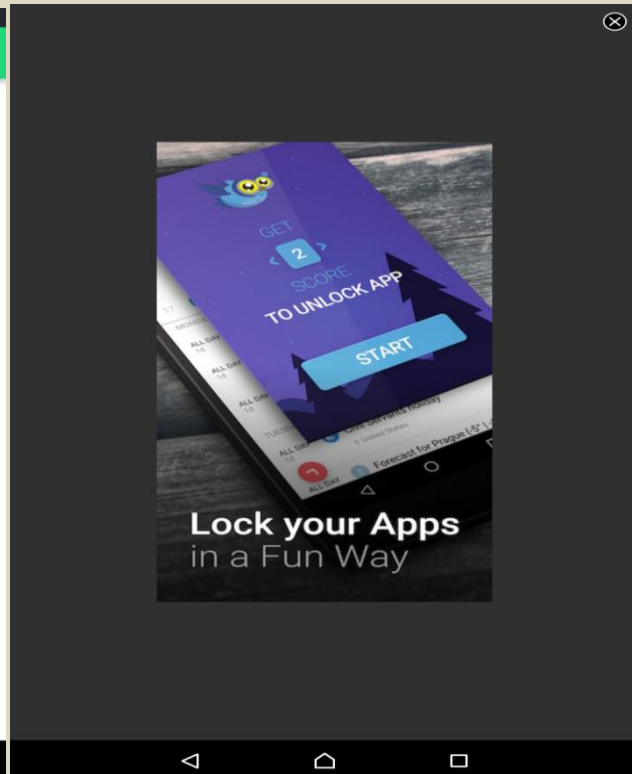
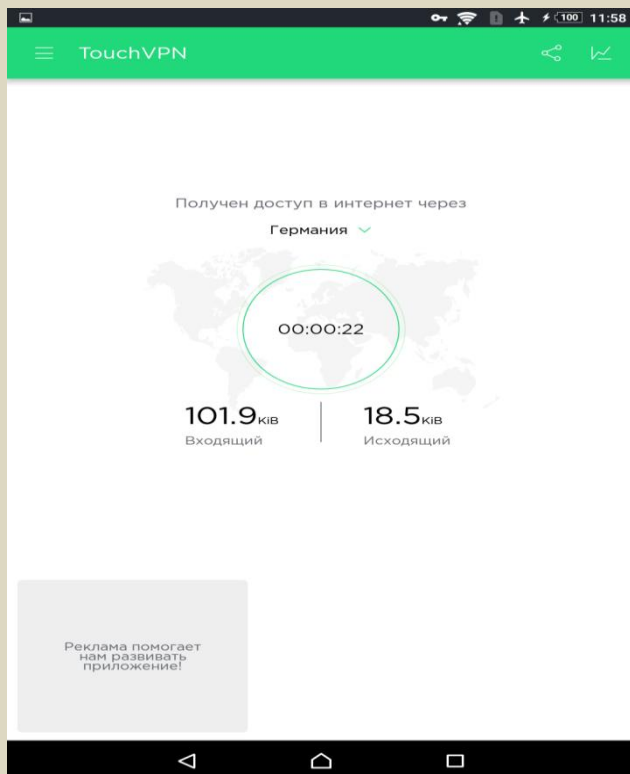
Бесплатный неограниченный сервис VPN.

Требуется Android: 4.0.3+

Русский интерфейс: Да

Google Play: <https://play.google.com/store/apps/details?id=com.northghost.touchvpn&hl=ru>

Страница 4PDA: <http://4pda.ru/forum/index.php?showtopic=749832>



1. Ищем и отключаем activity с рекламой в AndroidManifest.xml

Можно просто удалить эти строки.

```
<activity  
android:configChanges="keyboard|keyboardHidden|orientation|screenLayout|screenSize|smallestScreenSize|ui  
Mode" android:name="disabled_com.google.android.gms.ads.AdActivity"  
android:theme="@android:style/Theme.Translucent"/>
```

```
<activity android:configChanges="keyboard|keyboardHidden|orientation|screenLayout|screenSize"  
android:label="@string/app_name" android:name="disabled_com.facebook.FacebookActivity"  
android:theme="@android:style/Theme.Translucent.NoTitleBar"/>
```



```
<activity android:configChanges="keyboardHidden|orientation|screenSize"
android:launchMode="singleInstance" android:name="disabled_com.facebook.ads.InterstitialAdActivity"
android:noHistory="true"/>
```

```
<activity android:configChanges="keyboardHidden|orientation|screenSize"
android:launchMode="singleInstance" android:name="disabled_com.mopub.mobileads.MoPubActivity"
android:noHistory="true"/>
```

```
<activity android:configChanges="keyboardHidden|orientation|screenSize"
android:launchMode="singleInstance" android:name="disabled_com.mopub.mobileads.MraidActivity"
android:noHistory="true"/>
```

```
<activity android:configChanges="keyboardHidden|orientation|screenSize"
android:launchMode="singleInstance" android:name="disabled_com.mopub.common.MoPubBrowser"
android:noHistory="true"/>
```

```
<activity android:configChanges="keyboardHidden|orientation|screenSize"
android:launchMode="singleInstance"
android:name="disabled_com.mopub.mobileads.MraidVideoPlayerActivity" android:noHistory="true"/>
```

```
<activity android:name="disabled_com.google.android.gms.ads.purchase.InAppPurchaseActivity"
android:theme="@style/Theme.IAPTheme"/>
```

2. УДАЛЯЕМ МЕСТО ПОД БАННЕР (реклама помогает нам развивать приложение)

В ресурсах ищем фразу **реклама помогает нам развивать приложение**

Находим в ...res\values-ru\strings.xml

```
<string name="ad_helps_support">Реклама помогает нам развивать приложение!</string>
```

Далее в ресурсах ищем **ad_helps_support**, находим 2 файла:

...res\layout\carousel_layout_ghost.xml

...res\layout\carousel_layout_ghost_white.xml

В этих файлах, редактируем строки

```
<LinearLayout android:orientation="vertical" android:background="@drawable/carousel_bg"
android:padding="40.0dip" android:layout_width="fill_parent" android:layout_height="fill_parent"
android:layout_centerInParent="true">
```

```
<TextView android:textColor="@color/text_gray" android:gravity="center" android:id="@id/placehoder_text"
android:layout_width="140.0dip" android:layout_height="fill_parent"
android:text="@string/ad_helps_support" />
```

В этих строках меняем все :

"40.0dip" и "140.0dip" и "fill_parent" на "0.0dip"

3. УДАЛЯЕМ ВЫСКАКИВАЮЩЕЕ ОКНО LikeDialog

нравится приложение?

помогите нам стать лучше. отправьте нам свои пожелания или напишите положительный отзыв
не совсем да

```
...\smali\com\northghost\touchvpn\activity\MainActivity.smali
...\smali\com\northghost\touchvpn\activity\MainActivity$2.smali
```

В этих файлах, ищем строки содержащие showLikeDialog

```
invoke-direct {p0}, Lcom/northghost/touchvpn/activity/MainActivity; -> showLikeDialog()V
и
# invokes: Lcom/northghost/touchvpn/activity/Main nActivity; -> showLikeDialog()V
```

Удаляем их полностью. Все рекламы больше нет.

Пример удаление рекламы из приложения

Длинные нарды_4.60.apk

Русский интерфейс: Да

Google Play: https://play.google.co...e=apps_topselling_free

Страница 4PDA: <http://4pda.ru/forum/index.php?showtopic=295456>

Краткое описание:

одна из разновидностей игры в нарды.

1. Ищем и удаляем activity с рекламой в AndroidManifest.xml

Удаляем строки частично или полностью содержащие в себе (удалить только **activity**):

- com.amazon.device.ads
- com.applovin.adview
- com.appodeal.ads
- com.chartboost.sdk.CBImpressionActivity
- com.facebook.ads
- com.flurry.android.FlurryFullscreenTakeoverActivity
- com.google.android.gms.ads
- com.mopub
- com.startapp.android.publish
- com.startapp.android.publish.list3d.List3DActivity
- com.unity3d.ads.android.view.UnityAdsFullscreenActivity
- com.unity3d.ads.android2.view.UnityAdsFullscreenActivity
- com.vungle.publisher.FullScreenAdActivity
- com.yandex.mobile.ads
- om.jirbo.adcolony.AdColony
- org.nexage.sourcekit.mraid.MRAIDBrowser
- org.nexage.sourcekit.vast.activity.VASTActivity

- org.nexage.sourcekit.vast.activity.VPAIDActivity
- ru.mail.android.mytarget.ads

Если нужен полный экран ищем в...\res\values\dimens.xml

```
<dimen name="ad_height">50.0dip</dimen>
<dimen name="ad_width">320.0dip</dimen>
```

Меняем на:

```
<dimen name="ad_height">0.0dip</dimen>
<dimen name="ad_width">0.0dip</dimen>
```

ЧТОБЫ НЕ ВЕРЕЩАЛА ОБ ОШИБКАХ: Missing permissions и Missing activities

В файле...\smali\com\appodeal\ads\g\a.smali ищем:

```
if-nez v0, :cond_2
```

```
const-string v0, "Missing permissions:"
```

Меняем на:

```
if-eqz v0, :cond_2
```

```
const-string v0, "Missing permissions:"
```

Потом ищем (3 штуки):

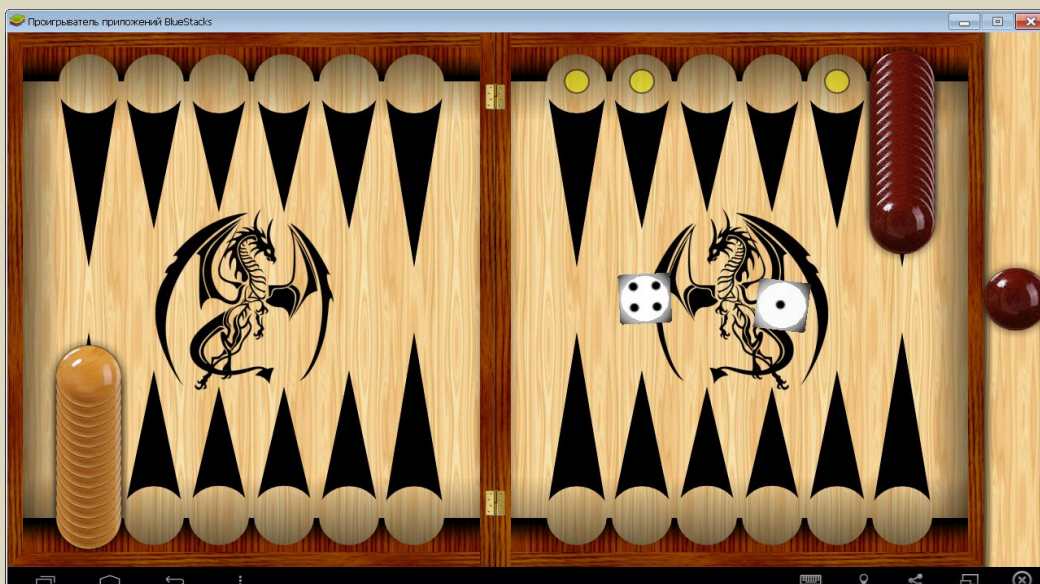
```
if-nez v0, :cond_0
```

```
const-string v0, "Missing activities:"
```

Меняем на:

```
if-eqz v0, :cond_0
```

```
const-string v0, "Missing activities:"
```



Все рекламы больше нет.