

## **Тестовое задание для dr.Web.**

Выполнил Котиков Александр. +7 921 350 06 55. тг: [LexCircumflex](#)

ГитХаб: <https://github.com/AlexanderKott/apkChecksum> 07.12.2025

Здравствуйте, уважаемая команда Доктор Веб!

Здесь вы можете найти информацию касательно выполненного тестового задания.

### **Изначальный текст задания:**

*Разработайте приложение, показывающее информацию об установленных приложениях.*

*Приложение:*

*1. Содержит экран со списком установленных приложений*

*2. По нажатии на элемент списка переходит на экран с подробной информацией об установленном приложении*

*3. Экран с подробной информацией должен содержать следующие данные:*

*\* название приложения*

*\* версия;*

*\* имя пакета приложения;*

*\* контрольная сумма apk-файла.*

*4. Так же на экране с информацией о приложении должна быть кнопка, открывающая приложение*

*Алгоритм контрольной суммы можно выбрать самостоятельно (SHA-1, MD5, CRC32 или иной)*

*Ограничений по стеку технологий нет. Приложение должно работать на устройствах с ОС Android 7.0 и выше*

*Текст задания полный*

*Все на Ваше усмотрение*

### **Выполнение задания**

Проанализировав текст задания и спектр системных утилит на сайте F-Droid для работы с apk андроид, я решил, что будет правильным создать не «еще одну утилиту» на подобии того, что представлено на сайте, а написать решение которое поможет ИБ – специалистам (далее пользователь), в их повседневной работе.

Кроме того, я предположил, что разработанное решение должно обладать атрибутами законченного программного продукта, а именно:

- Решать определенную узкую задачу
- Учитывать специфику работы пользователя
- Быть надежным
- Обладать удобным (но не экзотическим) интерфейсом
- Иметь документацию

### **Назначение программы**

Программа является детектором модификаций в base apk файлах.

Виды таких модификаций: патчинг, подмена файла, повреждение структуры.

## **Тестовое задание для dr.Web.**

Выполнил Котиков Александр. +7 921 350 06 55. тг: [LexCircumflex](#)

ГитХаб: <https://github.com/AlexanderKott/apkChecksum> 07.12.2025

## **Работа программы:**

Детектирование модификаций происходит по средствам вычисления SHA-256 на файлах-образцах до модификации и после (с последующей сверкой и выводом результата).

## **Типовой сценарий использования:**

- 1) Подготовить тестовый стенд (эмулятор или устройство).
- 2) Установить приложение Цель (для экспериментов).
- 3) Установить Детектор (мое приложение).
- 4) С помощью детектора определить эталонный хэш целевого приложения.
- 5) Выполнить целевые действия над целью(патчинг и тд).
- 6) Замерить актуальный хэш с помощью Детектора.
- 7) Сравнить полученные хэши.
- 8) Сгенерировать отчет о сверке (при необходимости).

## **Обоснование выбора хэш функции для работы приложения**

Компрометация хеш-функций

Функция	Дата/Год первой практической коллизии	Ссылка на источник	Примечания
MD5	2004	MD5 Collision Example (2004)	Коллизии найдены группой исследователей под руководством Ван Сяоюнь (Wang Xiaoyun).
SHA-1	2017	SHA-1 Collision Shattered (2017)	Google и CWI Amsterdam опубликовали первую практическую коллизию.
CRC32	N/A	RFC 1952	Не является криптографической функцией, не может быть "скомпрометирована" в ИБ-смысле.
SHA-256	N/A	N/A	Криптографическая стойкость. Атак, приводящих к практическим коллизиям, на начало декабря 2025 года не найдено.
SHA-512	N/A	N/A	Криптографическая стойкость. Атак, приводящих к практическим коллизиям, на начало декабря 2025 года не найдено.

## **Скорость и доступность**

Функция	Относительная скорость (на Android)	Доступность в Android OS	Примечания
---------	-------------------------------------	--------------------------	------------

## Тестовое задание для dr.Web.

Выполнил Котиков Александр. +7 921 350 06 55. тг: [LexCircumflex](#)

ГитХаб: <https://github.com/AlexanderKott/apkChecksum> 07.12.2025

<b>CRC32</b>	<b>Очень быстрая</b>	Встроена (java.util.zip.CRC32)	Используется для контроля ошибок передачи данных
<b>MD5</b>	<b>Быстрая</b>	Встроена (java.security.MessageDigest)	Быстрая
<b>SHA-1</b>	<b>Быстрая</b>	Встроена (java.security.MessageDigest)	Быстрая
<b>SHA-256</b>	<b>Средняя/Быстрая</b>	Встроена (java.security.MessageDigest)	Поддерживает аппаратное ускорение.
<b>SHA-512</b>	<b>Средняя</b>	Встроена (java.security.MessageDigest)	Может быть немного медленнее SHA-256 на старых 32-битных устройствах.

Вывод:

MD5, SHA-1, CRC32 – точно не подходят для цели детектора.

SHA-256 и SHA-512 обладают криптографической стойкостью, не скомпрометированы. Но я решил что SHA-512 слишком избыточна для данной задачи (не было требований повышенной безопасности в задании), а так же она может работать медленней на старых устройствах (в условиях тестового задания был указан андроид 7.0).

Выбор - SHA-256

## Тестирование

Перед загрузкой детектора в гит хаб было произведено тестирование:

написаны юнит тесты для хэш функции.

общее ручное тестирование функций приложения.

была произведена акаака на apk файл с нарушением целостности, чтобы убедиться что детектор работает. (см. папку «процесс» на моем гит хаб)

Тестирование происходило на устройствах:

- Эмулятор андроид студии

Properties  
avd.ini.displayname Android 7.0 (AOSP)  
avd.ini.encoding UTF-8  
AvdId Android\_7.0\_AOSP  
disk.dataPartition.size 6442450944  
fastboot.chosenSnapshotFile  
fastboot.forceChosenSnapshotBoot no  
fastboot.forceColdBoot no  
fastboot.forceFastBoot yes  
hw.accelerometer yes  
hw.arc false  
hw.audioInput yes  
hw.battery yes  
hw.camera.back virtualscene

## Тестовое задание для dr.Web.

Выполнил Котиков Александр. +7 921 350 06 55. тг: [LexCircumflex](#)

ГитХаб: <https://github.com/AlexanderKott/apkChecksum> 07.12.2025

```
hw.camera.front      emulated
hw.cpu.ncore        4
hw.device.hash2     MD5:64b26f5eaf2f4673290cdc23d0c65386
hw.device.manufacturer   Generic
hw.device.name       small_phone
hw.dPad             no
hw.gps              yes
hw.gpu.enabled      yes
hw.gpu.mode         auto
hw.gyroscope        yes
hw.initialOrientation portrait
hw.keyboard          yes
hw.lcd.density      320
hw.lcd.height       1280
hw.lcd.width        720
hw.mainKeys          no
hw.ramSize           1024
hw.sdCard            yes
hw.sensors.light    no
hw.sensors.magnetic_field yes
hw.sensors.orientation yes
hw.sensors.pressure  yes
hw.sensors.proximity yes
hw.trackBall         no
image.androidVersion.api 24
image.sysdir.1        system-images\android-24\default\x86_64\
PlayStore.enabled     false
runtime.network.latency none
runtime.network.speed  full
showDeviceFrame       yes
skin.dynamic          yes
tag.display
tag.displaynames
tag.id                default
tag.ids               default
vm.heapSize           80
```

## Андроид студия:

Android Studio Meerkat | 2024.3.1 Patch 1

Build #AI-243.24978.46.2431.13208083, built on March 13, 2025

Runtime version: 21.0.5+13047016-b750.29 amd64

VM: OpenJDK 64-Bit Server VM by JetBrains s.r.o.

Toolkit: sun.awt.windows.WToolkit

Windows 11.0

Kotlin plugin: K2 mode

GC: G1 Young Generation, G1 Concurrent GC, G1 Old Generation

Memory: 7168M

Cores: 20

Registry:

ide.experimental.ui=true

Non-Bundled Plugins:

com.intellij.marketplace (243.24978.86)

com.github.deeepamin.gitlabciaid (1.0.5)

com.arcticicestudio.nord.jetbrains (0.13.0)

com.jetbrains.grayTheme (1.3)

## Тестовое задание для dr.Web.

Выполнил Котиков Александр. +7 921 350 06 55. тг: [LexCircumflex](#)

ГитХаб: <https://github.com/AlexanderKott/apkChecksum> 07.12.2025

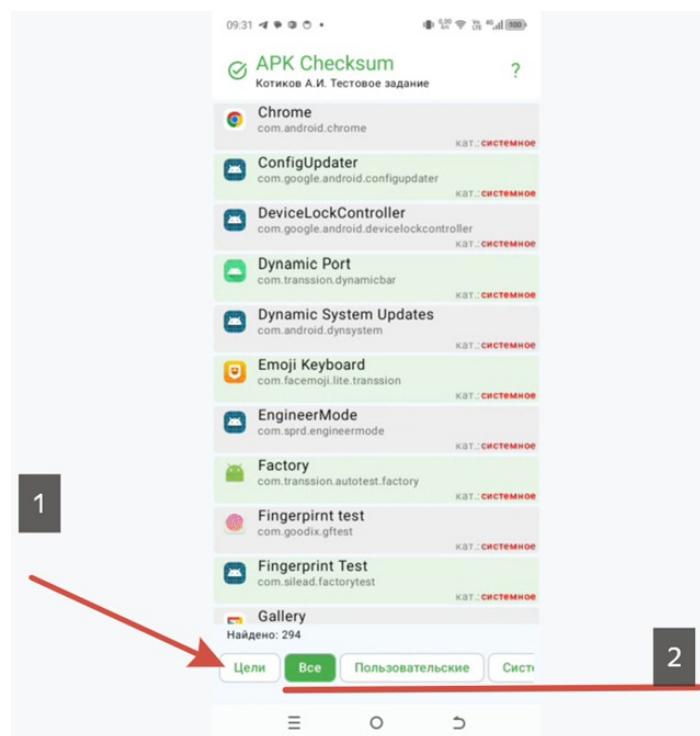
- Tecno Spark Android 14, arm 64

Наиболее тщательно было протестировано на Андроид 7.0

## Инструкция использования детектора

Процесс работы организованы таким образом, что вы сначала выбираете интересующие вас приложения (цели) во вкладках снизу (см. 2), потом (с помощью свайпа вправо вы подтверждаете выбор), и они попадают во вкладку Цели (см 1), после чего вы можете делать снимки состояния приложений.

- Запустите программу, вы увидите:



## Тестовое задание для dr.Web.

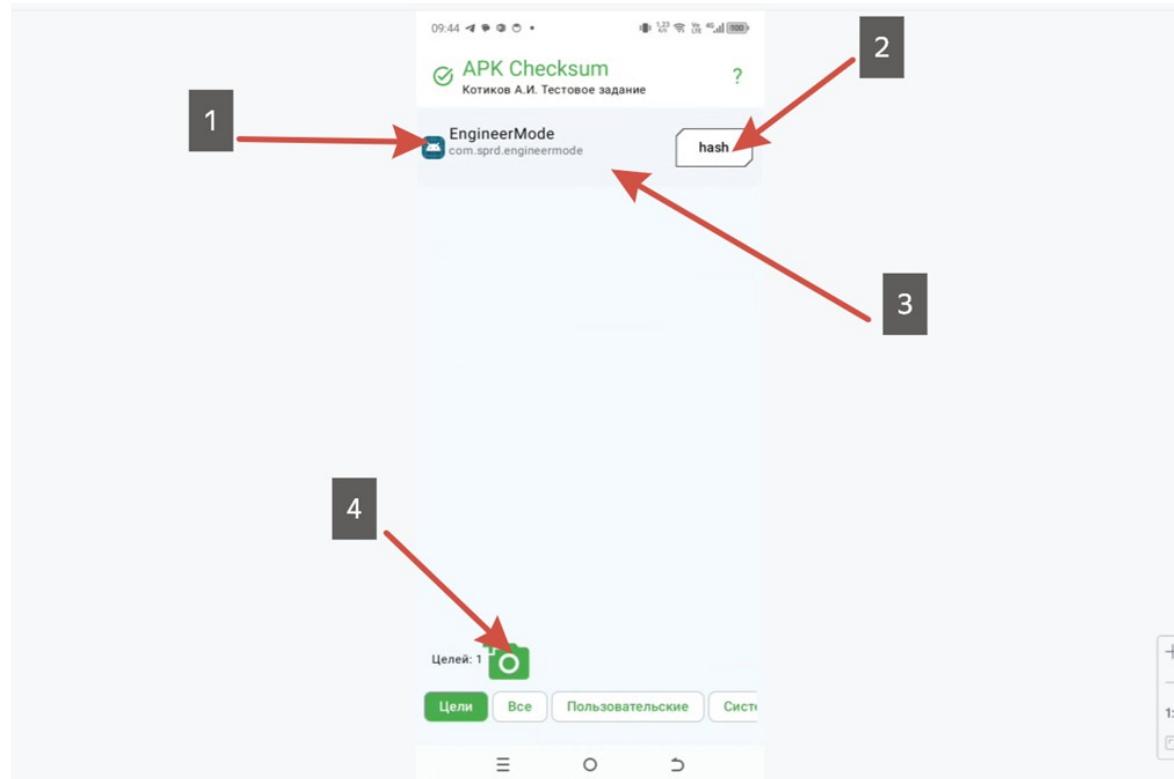
Выполнил Котиков Александр. +7 921 350 06 55. тг: [LexCircumflex](#)

ГитХаб: <https://github.com/AlexanderKott/apkChecksum> 07.12.2025

- Сдвигаете свайпом нужное приложение в цели:



- Перейдите в Цели



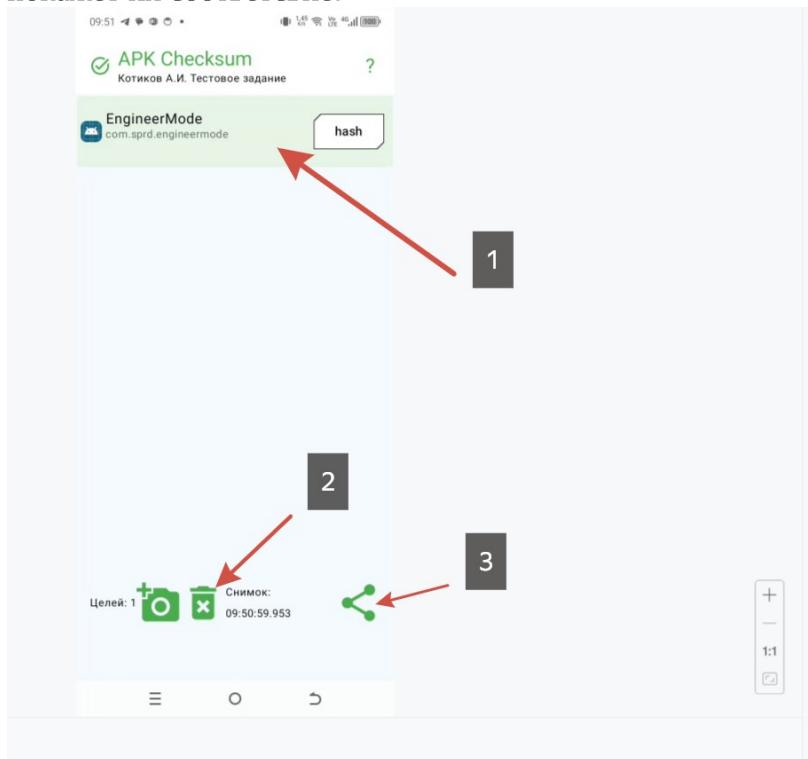
- приложения цели – те у которых будет замерен хэш
- кнопка, чтобы увидеть хэши после снимков
- нажмите на элемент списка чтобы увидеть дополнительную информацию об апк.

## Тестовое задание для dr.Web.

Выполнил Котиков Александр. +7 921 350 06 55. тг: [LexCircumflex](#)

ГитХаб: <https://github.com/AlexanderKott/apkChecksum> 07.12.2025

- 4) Кнопка «фото» чтобы сделать снимок состояния приложения и вычислить хэши. Один клик – снять эталонный хэш, второй клик – снять актуальный хэш еще раз (после проведения действий над приложением - целью).
- После двух кликов на «фото», программа сравнит хэши автоматически и цветом покажет их соответствие:

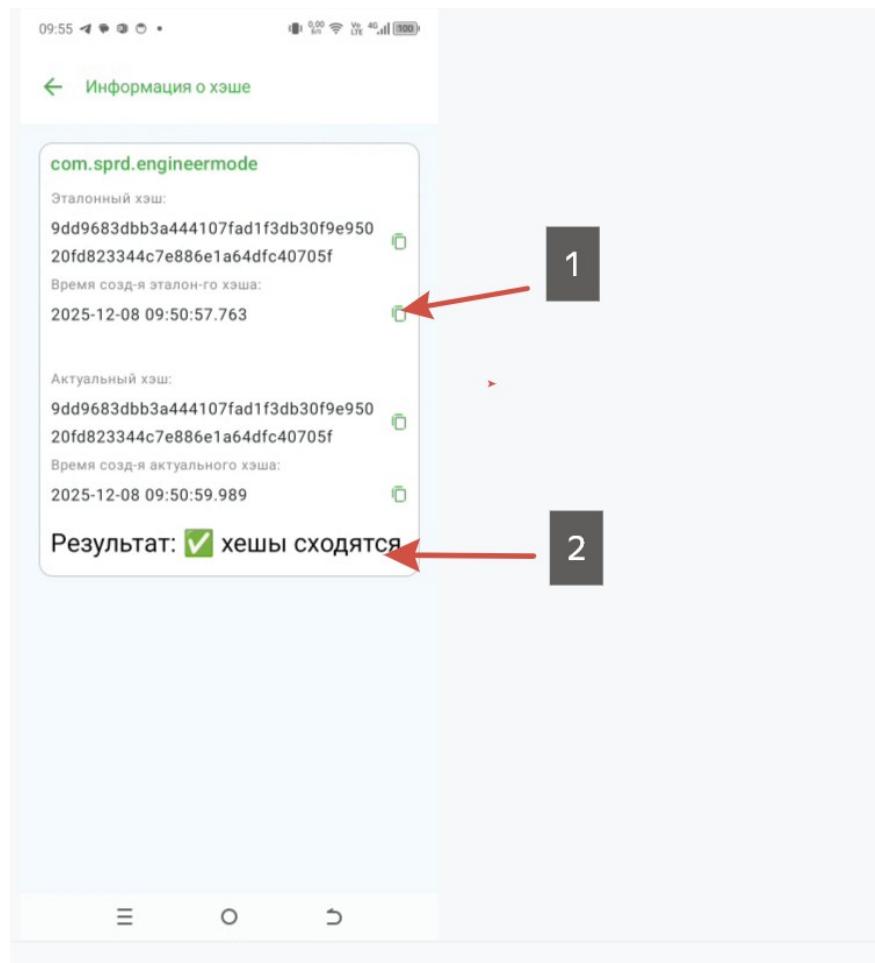


- Зеленый цвет означает что хэши сошлись (соответствуют друг другу). Красный – не сошлись.
  - Нажмите «корзина» чтобы вернуться в режим просмотра всех приложений.
  - Сгенерировать отчет если нужно.
- Нажмите кнопку «Hash», чтобы увидеть детали.

## Тестовое задание для dr.Web.

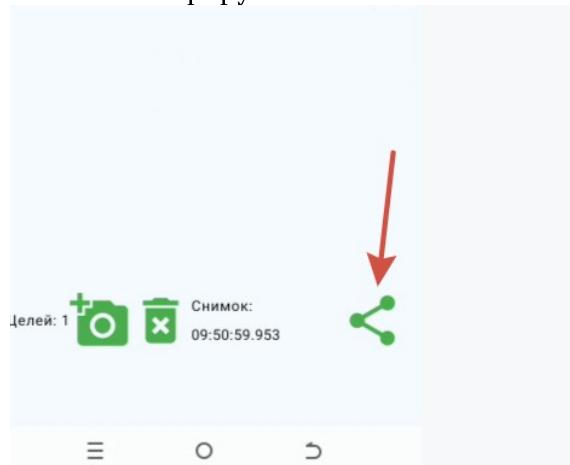
Выполнил Котиков Александр. +7 921 350 06 55. тг: [LexCircumflex](#)

ГитХаб: <https://github.com/AlexanderKott/apkChecksum> 07.12.2025



- 1) вы можете скопировать полученные данные для своей работы
- 2) вы видите результат сравнения хешей

- Нажмите назад
- Сгенерируйте отчет



## Тестовое задание для dr.Web.

Выполнил Котиков Александр. +7 921 350 06 55. тг: [LexCircumflex](#)

ГитХаб: <https://github.com/AlexanderKott/apkChecksum> 07.12.2025

Пример отчета из приложения:

### ОТЧЕТ

Дата проведения: 08.12.2025 10:42:37  
Формат хэша: SHA-256  
Система: Android

#### ТЕСТОВАЯ СРЕДА

Manufacturer:	TECNO
Model:	TECNO KL4
Device:	TECNO-KL4
Android Version (Release):	14
Android API Level:	34

#### СПИСОК ЦЕЛЕЙ

№	Название приложения	Эталонный хэш	Время создания эталонного хэша	Актуальный хэш	Время создания актуального хэша	Пакет
1	EngineerMode	9dd9683dbb3a444107fad1f3db30f9e95020f d823344c7e886e1a64dfc40705f	08.12.2025 10:42:36	9dd9683dbb3a444107fad1f3db30f9e95020f d823344c7e886e1a64dfc40705f	08.12.2025 10:42:36	com.sprd.engineermode

Отчет сгенерирован автоматически • 08.12.2025 10:42:37

Android API Level:

34

#### СПИСОК ЦЕЛЕЙ

№	Название приложения	Эталонный хэш	Время создания эталонного хэша	Актуальный хэш	Время создания актуального хэша
1	EngineerMode	9dd9683dbb3a444107fad1f3db30f9e95020f d823344c7e886e1a64dfc40705f	08.12.2025 10:42:36	9dd9683dbb3a444107fad1f3db30f9e95020f d823344c7e886e1a64dfc40705f	08.12.2025 10:42:36

Отчет сгенерирован автоматически • 08.12.2025 10:42:37

Отчет может быть просмотрен только в стороннем приложении.

- Нажмите «назад» чтобы вернуться
- Нажмите «корзина» чтобы выйти из режима снимков

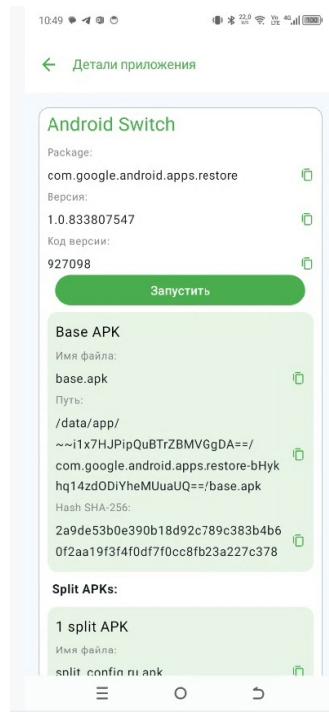
## Просмотр информации об apk.

Необходимо нажать на интересующее приложение из списка в любом разделе.

## Тестовое задание для dr.Web.

Выполнил Котиков Александр. +7 921 350 06 55. тг: [LexCircumflex](#)

ГитХаб: <https://github.com/AlexanderKott/apkChecksum> 07.12.2025



## Стек

Кotlin, Compose, MVVM

## Использование ИИ

При выполнении задания был использован ИИ для вспомогательных целей.

## Что можно улучшить

Сделать поддержку DI

Сделать большее количество снимков для apk

Адаптировать для большего количества устройств, поправить мелкие баги

Спасибо большое за ваше внимание! :) Буду ждать обратную связь 😊

тг: LexCircumflex

Котиков Александр