**#tinkoff fintech**

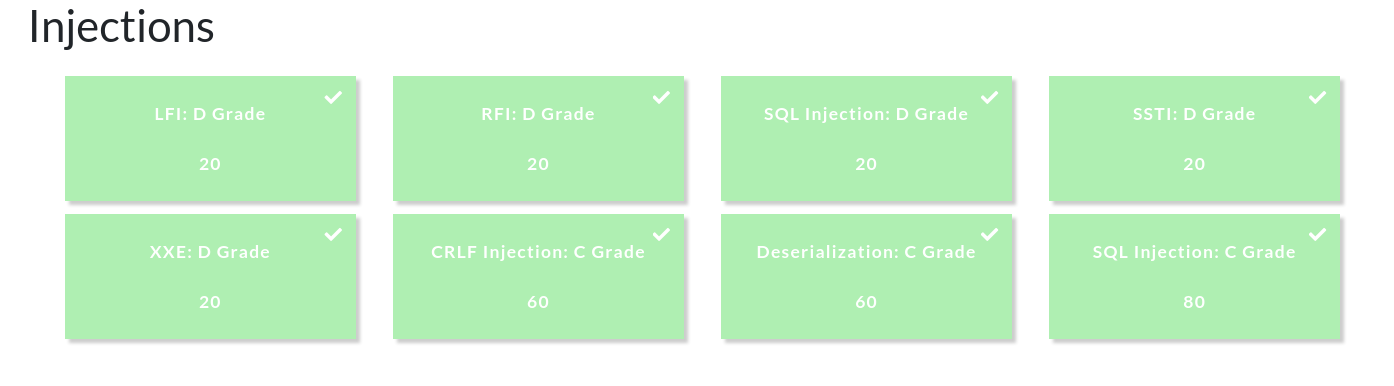
homework #3

MOBILE   
Security

**03.03.2019**

# Задачи:

1. **Необходимо выполнить максимальное количество заданий на нашем CTF портале.**
2. **Каждый захваченный флаг нужно прислать с подробным описанием захвата.**
3. **Оценки будут проставлены на основе количества захваченных флагов.**



# Restore Key: D Grade aka open-your-eyes

**The software uses a protection mechanism which depends in common on its obscurity. Knowledge of its algorithms or key data is sufficient to defeat the mechanism. This reliance on "security through obscurity" can lead to weaknesses if an attacker is able to reverse engineer the mechanism.**

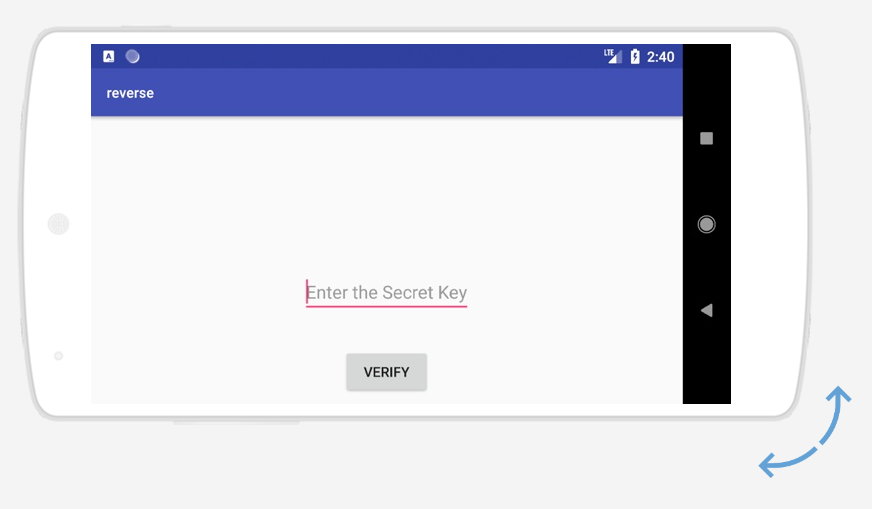
### 

# Objection

**You have android application, you need to find out the key.**

Что понадобится для поиска секретного ключа?

Во-первых: софт. А именно, для просмотра содержимого апк файла юзаем jadx. Для проверки найденного ключа я использовал онлайн эмулятор appetize.io



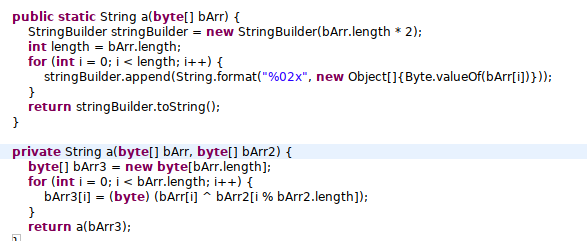
А еще, пара глаз :)

**Открываем наш апк и смотрим.**

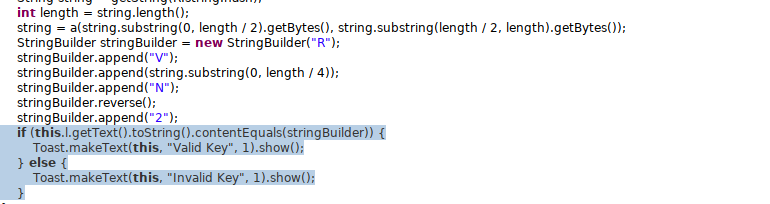
Стартуем отсюда:

****

По сути - это мэйн файл с логикой проверки правильности ключа.

****

Эта часть (вверху) что-то типа шифрования. Из функции GetKey мы отправляем байт-массивы, а на возврате получаем видоизмененные строки.



На прошлой странице идет проверка введенного ключа. Если equal -> то ключ верный.

Можно проверять ключ в андроид эмуляторе. Осталось только найти ключ.



И

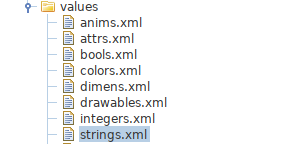


Вот по этим двум строчкам видно что нам надо, для валидации ключа, чтобы две строки (this и stringBuilder) были равны. Значит нам надо вывести значение стринг билдера, но для начала найти хэш стринга.

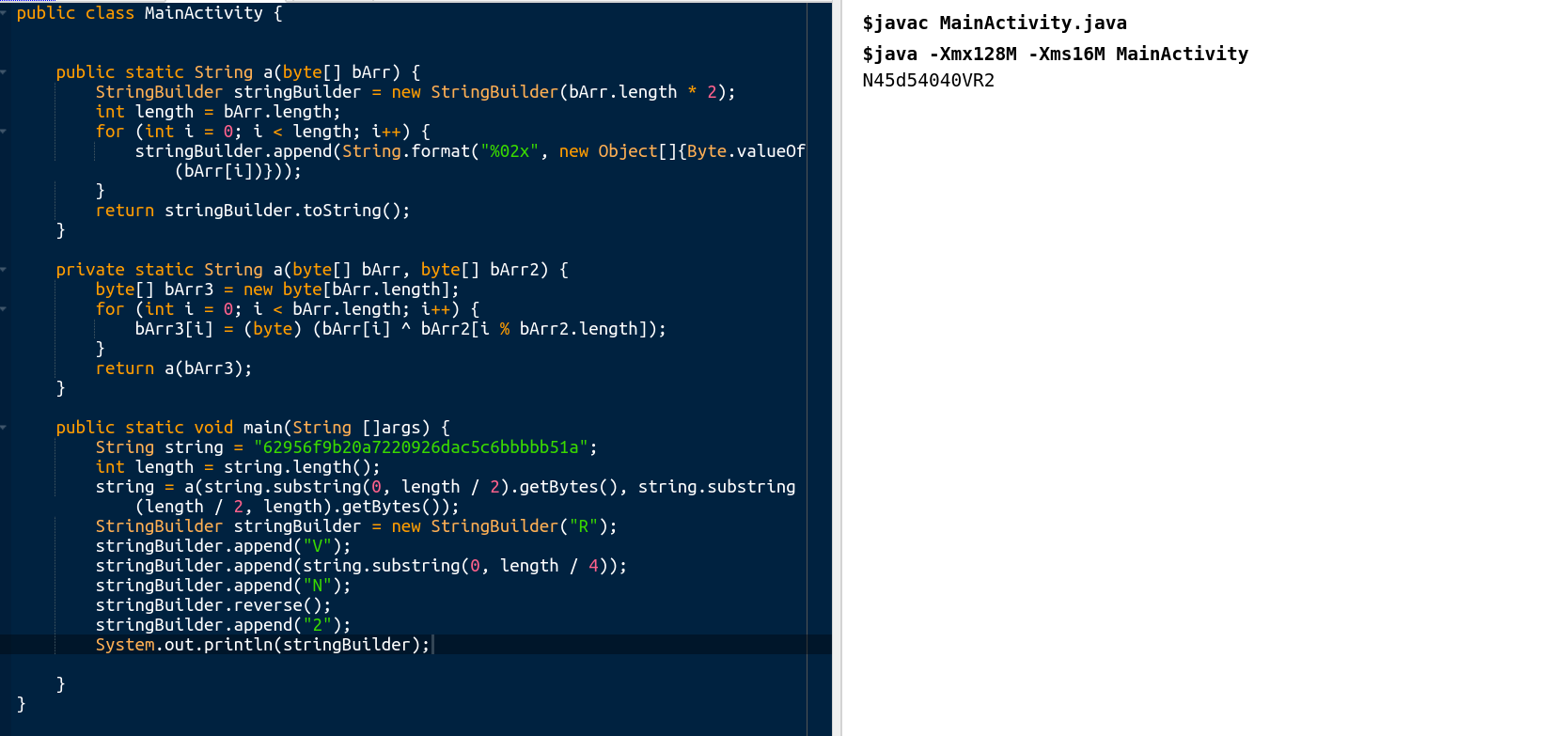


Тут нашли айди хэшстринга. Но нам надо реальное значение хэша,а именно

<string name="hash">62956f9b20a7220926dac5c6bbbbb51a</string>

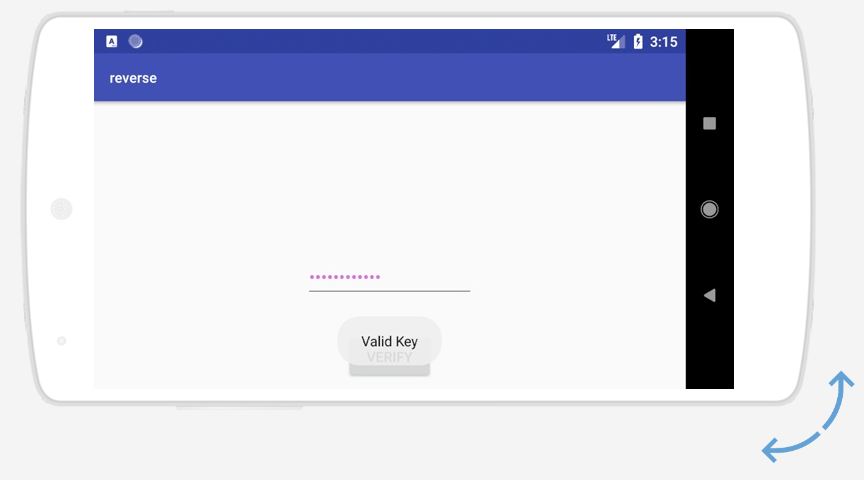


А теперь, копируем код из программы в какой-нибуть компилятор и принтуем наш стрингбилдер.



Ключ: N45d54040VR2

Проверяем в эмуляторе:

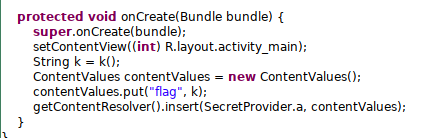


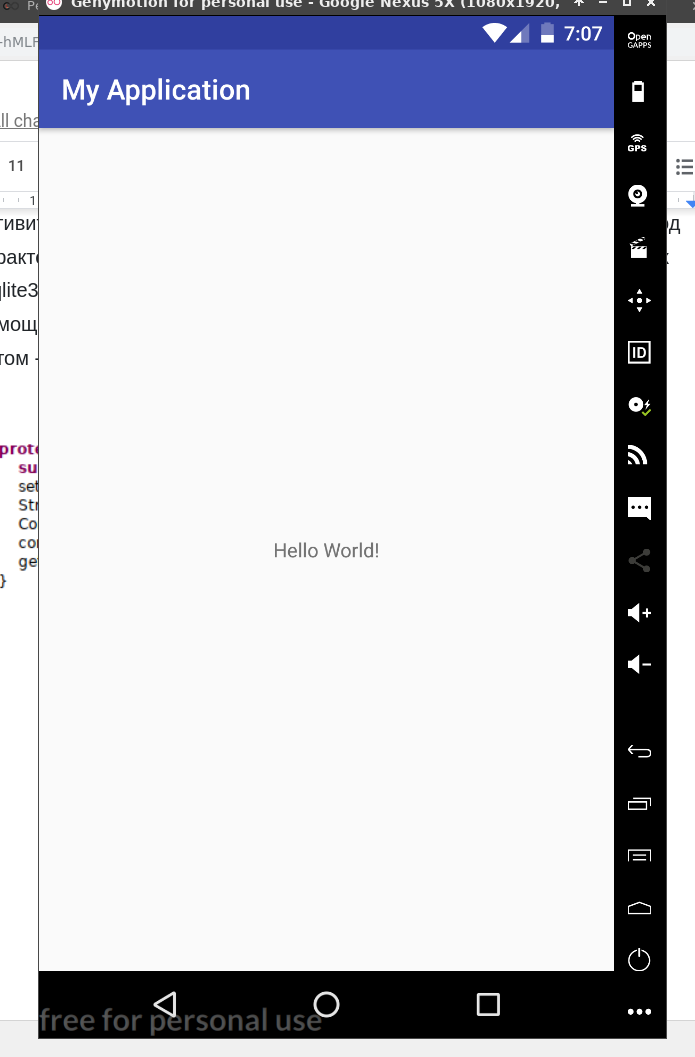
# Query the Flag: D Grade aka open-your-eyes

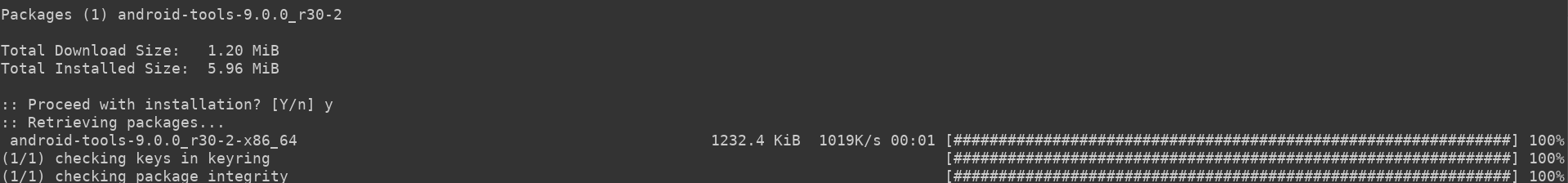
**The software does not restrict or incorrectly restricts access to a resource from an unauthorized actor.**

По классике, открываем джадексом, смотрим код, вникаем :) в мэйм активити обращаем внимание на строки с contentValues.put(“flag”, k); Метод характерный для баз данных. Понимаем что флаг хранится с базе данных (sqlite3). Самый просто способ - телефон с рутом и на нем открыть с помощью sqlite editor’а или аналога. К сожалению не было телефона с рутом - поэтому используем эмулятор (Genymotion) а также адб шелл.

### 







Если по шагам, то примерно алгоритм такой:

Adb devices (проверяем чтобы эмулятор определился)

Adb shell

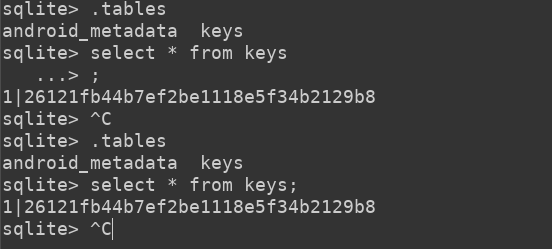
Ls ; cd data/data/<ru.tinkoff………>/databases

Sqlite3

.tables

Selecto \* from keys;

Profit :)



### ВТОРОЙ ПОДХОД К РЕШЕНИЮ ПРЕДЫДУЩЕГО ТАСКА:

Суть подхода - использование утилиты drozer и экспортируемых provider’oв.

Как устанавливать дрозер писать не буду. Если коротко, то устанавливаем апк клиент дрозера на телефон. А через компьютер, через адб пробрасываем порты и подключаем консоль дрозера



После этого ищем провайдеров с помощью инструментария дрозера и отправляем запрос на контент в data.:

