ИССЛЕДОВАНИЕ ЗАЩИЩЕННОСТИ ПОЛЬЗОВАТЕЛЬСКИХ ДАННЫХ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ НА ПРИМЕРЕ МЕССЕНДЖЕРА WHATSAPP

A STUDY OF MOBILE APPLICATION USER DATA SECURITY ON THE EXAMPLE OF WHATSAPP MESSENGER

A. Karondeev
D. Kluev

Summary. The paper discusses information security of mobile applications user data. The proposed method for conducting security audit of mobile applications. The method is focused on identifying ways to access user data. The results of the audit of WhatsApp messenger are given in order to demonstrate the capabilities of this method.

Keywords: security audit, mobile applications, static analysis, dynamic analysis, Android, iOS, reverse engineering.

Карондеев Андрей Михайлович
Аспирант, Московский Государственный Технический Университет им. Н. Э. Баумана; Специалист отдела исследований, ООО «Оксиджен Софтвер» (Москва)
karondeev@oxygensoftware.com
Клюев Даниэль Витальевич
Специалист по исследованию облачных сервисов, ООО «Оксиджен Софтвер» (Москва); Морской Государственный Университет им. адмирала Г. И. Невельского
kluev@oxygensoftware.com

Аннотация. В статье рассматриваются вопросы информационной безопасности пользовательских данных мобильных приложений. Предложена методика проведения аудита безопасности мобильных приложений. Методика ориентирована на выявление способов получения доступа к пользовательским данным. Для демонстрации возможностей методики приведены результаты аудита мессенджера WhatsApp.

Ключевые слова: аудит безопасности, мобильные приложения, статический анализ, динамический анализ, Android, iOS, обратная разработка.

Введение

Мобильные телефоны уже давно стали нечто большим, чем простое средство для связи. Они плотно интегрированы в нашу жизнь и содержат просто колоссальное количество информации о пользователе начиная от списка контактов и истории общения, до сведений о передвижении и платежных данных. И вся эта информация содержится в данных мобильных приложений. Многие пользователи даже не задумываются о реальной защищенности своих данных и слепо верят в порой необоснованные заявления разработчиков. Тем не менее, когда дело касается приложений для мобильного банкинга или мессенджеров, используемых в корпоративной среде, ситуация принимает совершенно другой характер и для оценки защищенности часто привлекаются специалисты по аудиту безопасности мобильных приложений.

Мобильные технологии интенсивно развиваются, постоянно появляются новые методы обработки, хранения и передачи информации. В связи с этим возникает необходимость разработки новых методик проведения аудита безопасности мобильных приложений, учитывающих современные тенденции, например интеграцию в мобильные устройства облачных технологий. Стоит отметить, что данные, полученные из мобильных устройств, часто используются как доказательства при проведении расследований в корпоративной, гражданской или общеуголовной сферах. Расследование практически каждого уголовного дела в настоящее время включает в себя поиск, изъятие и исследование электронных следов, причем наиболее информативными являются сведения, извлекаемые из мобильных телефонов и смартфонах, принадлежащих потерпевшему и преступнику [1]. Для извлечения данных из мобильных устройств используются специализированные комплексы такие как UFED, Мобильный Криминалист, XRY [2]. При разработке таких решений возникает необходимость в проведении аудита безопасности мобильных приложении, ориентированного на выявление способов получения доступа к пользовательским данным и предполагающего наличие у потенциального нарушителя доступа к устройству. Все это говорит о том, что разработка такой методики является актуальной задачей.

Многие описанные ранее методики, например, используемые в работах [3][4], ориентированы для исследования программ на предмет соответствия требованиям информационной безопасности. Подобные методики предназначены для выявления типовых уязвимостей, таких как [5], [6]. Предложенная в данной статье методика

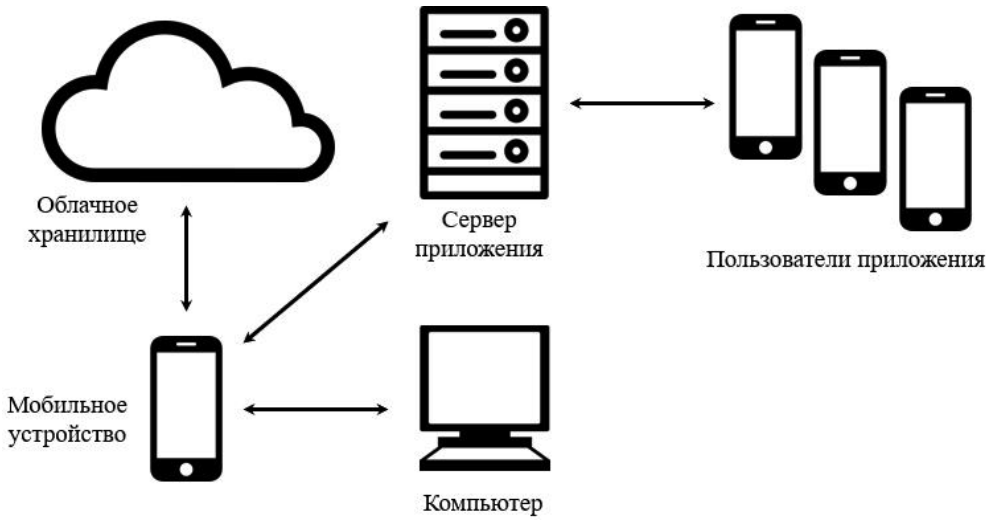

Рис. 1. Общая схема внешнего окружения мобильного приложения

в свою очередь является более узкоспециализированной и предназначена для выявления способов получения доступа к пользовательским данным.

Статья организована следующим образом. В разделе 1 приведена разработанная авторами методика проведения аудита безопасности мобильных приложений, а также описаны особенности её использования для Android и iOS, занимающих доминирующие положение на рынке мобильных операционных систем (ОС)[1]. В разделе 2 с целью продемонстрировать возможностей методики приведены результаты аудита мессенджера WhatsApp, как одного из наиболее популярных в России и мире.

1. Методика проведения аудита безопасности мобильных приложений

Методика

предполагает отсутствие исходных кодов анализируемого приложения и рассчитана на использование специалистом, владеющим навыками обратной разработки мобильных приложений и соответствующим опытом использования инструментов статического и динамического анализа. Предлагаемая методика проведения аудита безопасности мобильного приложения состоит из следующих этапов:

1. Настройка среды для анализа
2. Изучение внутреннего окружения приложения
3. Изучение внешнего окружения приложения
4. Анализ данных приложения
5. Анализ механизмов доступа приложения к пользовательским данным
6. Составление матрицы доступа к пользовательским данным

Первый этап состоит из подготовки тестовых устройств, установки на них изучаемого приложения и генерации тестовых данных, а также настройки инструментов статического и динамического анализа. Если в приложении реализована возможность взаимодействия между пользователями, то необходимо подготовить не менее двух тестовых устройств и не менее трех для приложений с возможностью группового взаимодействия. На тестовых устройствах необходимо обеспечить возможность полного доступу к памяти, а также интерфейс для взаимодействия со средствами динамического анализа. На Android/iOS устройствах для этого необходимо получить привилегии суперпользователя (root). На Android устройствах для этого можно разблокировать загрузчик и записать в устройство модифицированную операционную систему. На iOS устройствах для получения root привилегий необходимо пройти процедуру jailbreak.

В качестве инструмента для статического анализа Android/iOS приложений рекомендуется использовать IDA Pro или Ghidra. Оба инструмента позволяют производить дизассемблирование/декомпиляцию Android/iOS приложений, однако второй относится к свободному программному обеспечению[2]. В качестве средства для динамического анализа Android/iOS приложений рекомендуется использовать фреймворк

[1] Согласно данным компании Gartner, начиная с 2017 года суммарная доля рынка ОС Android и iOS превышает 99,9% https://www.gartner.com/en/newsroom/press-releases/2018–02–22-gartner-says-worldwide-sales-of-smartphones-recorded-first-ever-decline-during-the-fourth-quarter-of-2017

[2] https://github.com/NationalSecurityAgency/ghidra

Frida[1]. Данный фреймворк является кроссплатформенным инструментом для динамической бинарной инструментации (dynamic binary instrumentation — DBI).

В Android и iOS для приложений реализована так называемая концепция песочницы. Для каждого приложения ОС выделяет специальную изолированную среду для выполнения и хранения данных. На втором этапе изучается структура данной песочницы и составляется перечень хранимых в ней файлов.

На третьем этапе составляется схема внешнего окружения приложения. Общая схема внешнего окружения приложения изображена на рис. 1.

Схема, составленная в ходе исследования конкретного приложения, может отличаться от общей. В частности, могут отсутствовать некоторые из участников взаимодействия. Далее каждый узел схемы анализируется с целью выявления возможности хранения на нем пользовательских данных. Рассмотрим узлы общей схемы с точки зрения возможности хранения пользовательских данных. Помимо песочницы мобильные приложения могут хранить пользовательские данные в общей памяти устройства или на внешней SD-карте. Во многих мобильных приложениях реализован механизм резервного копирования/восстановления. В частности, для сохранения пользовательских данных на компьютер в ОС Android/iOS реализованы механизмы создания ADB/ iTunes бэкапов соответственно. Также в ОС Android/ iOS реализованы механизмы хранения/синхронизации данных с облачными хранилищами Google Drive/ iCloud соответственно. Приложение может обмениваться данными с сервером, исходя из этого у сервера есть возможность сохранять эти данные. Если же в приложении реализована возможность взаимодействовать с другими пользователями, то передаваемые данные могут храниться на сервере, или локально на устройствах каждого из участников взаимодействия. Далее для каждого узла схемы, к которым есть непосредственный доступ, составляется перечень переданных ему приложением файлов.

На четвертом этапе производится непосредственный анализ структуры файлов из перечней, составленных на втором и третьем этапах. Если у анализируемого файла есть признаки того, что он не содержит пользовательские данные, то он исключается из перечня для рассмотрения. Для хранения данных мобильные приложения Android/iOS наиболее часто используют форматы SQLite, XML, JSON, PLIST, а также различные фото/ауди/ медиа форматы. По ходу анализа составляется перечень файлов, которые имею неизвестный формат или хранят данные в зашифрованном виде. Файлы содержащие пользовательские данные заносятся в специальную таблицу, в которой указываются место хранения и краткое описание того какие именно пользовательские данные там храниться.

На пятом этапе посредством обратной разработки для файлов из перечня, составленного на четвертом этапе, производится восстановление формата или алгоритма шифрования. Для зашифрованных данных исследуется возможность их расшифровывания. Далее также посредством обратной разработки производится восстановления протоколов взаимодействия с узлами, к которым нет прямого доступа, но у которых есть возможность хранить пользовательские данные. Выявленные места хранения пользовательских данных заносятся в таблицу. Для пользовательских данных хранящихся в зашифрованном виде в специальное поле таблицы заносится описание способа расшифровывания. Для пользовательских данных хранящихся на внешних узлах в специальное поле таблицы заносится описание способа получения к ним доступа выявленного при восстановлении протокола взаимодействия.

На шестом этапе для каждого источника пользовательских данных составляется перечень возможностей, которыми должен обладать потенциальный нарушитель для получения доступа к ним. На основе данных перечней возможностей составляется общий список возможностей, которыми должен обладать нарушитель для доступа ко всем выявленным в ходе анализа источникам пользовательских данных. Далее формируется матрица доступа к пользовательским данных, столбцы которой соответствуют тем или иным возможностям потенциального нарушителя, а строки источникам пользовательских данных. На пересечении строк и столбцов указывается необходимость соответствующей столбцу возможности для получения доступа к соответствующему строке источнику пользовательских данных. Для источников данных расположенных на внешних узлах в матрицу помещается ссылка на способ получения к ним доступа выявленный на пятом шаге. Для источников, хранящих пользовательские данные в зашифрованном виде, в матрицу помещается ссылка на способ расшифровывания или специальная пометка о том, что расшифровывание вычислительно невозможно.

При проведении аудита безопасности рассматривается конечное число моделей потенциального нарушителя, для каждой из которых задан определенный набор возможностей. Возможности каждой модели сопоставляются с матрицей доступа. Далее на основе результатов сопоставления для каждой модели потенциального нарушителя формируется перечень пользовательских

[1] https://github.com/frida/frida

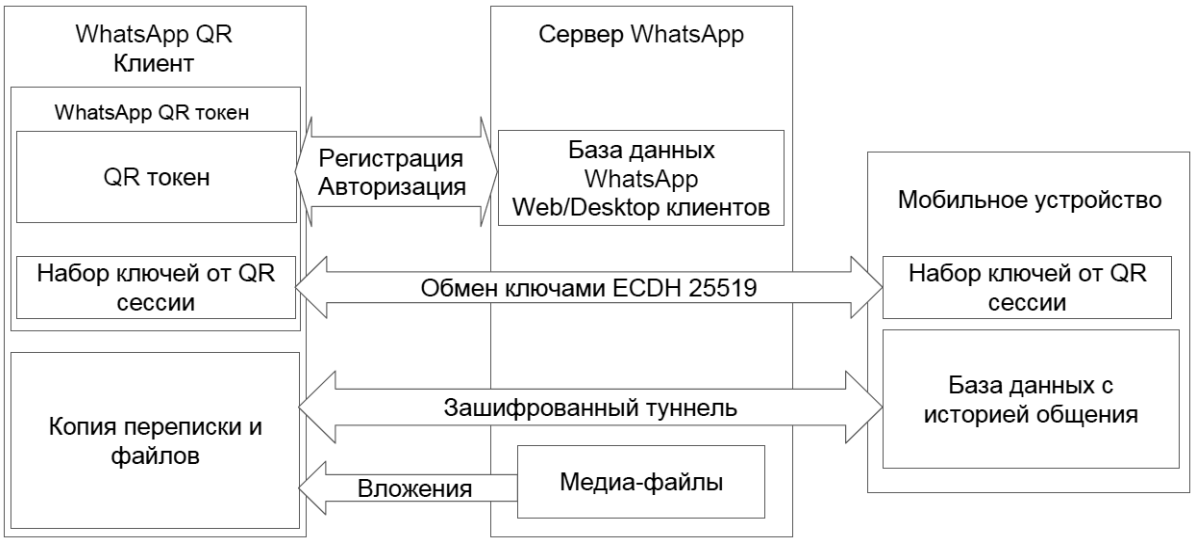

Рис. 2. Схема работы механизма WhatsApp QR

данных, к которым он может получить доступ. Для пользовательских данных хранящихся на внешних узлах указывается ссылка на способ получения к ним доступа. Для пользовательских данных хранящихся непосредственно в мобильном устройстве составляется типовой набор способов, который впоследствии используется и для других приложений.

7. Результаты аудита безопасности мессенджера WhatsApp

Для демонстрации возможностей предложенной методики рассмотрим выявленные с её помощью способы доступа к пользовательским данным мессенджера WhatsApp. Схема внешнего окружения приложения WhatsApp аналогична изображенной на рис. 1. В ходе проведения аудита были выявлены следующие источники пользовательских данных:

1. Песочница приложения WhatsApp
2. Общая память устройства
3. Компьютер
4. Облачное хранилище
5. Сервер WhatsApp

В песочнице WhatsApp как Android, так и iOS устройства хранят базу данных (БД) с историей общения. Данная БД имеет формат SQLite и храниться в незашифрованном виде. На Android устройствах в общую память, доступной по протоколу MTP, сохраняются файлы, которыми между собой обмениваются пользователи в ходе общения — фото, видео, аудио, текстовые документы и т.д. Также в общую память уже в зашифрованном виде сохраняются резервные копии БД с историей общения. Данные БД зашифрованы алгоритмом AES с использованием ключа, который храниться в песочнице WhatsApp в бинарном key файле. Локальная резервная копия автоматически создается раз в сутки и на устройстве хранятся последние 7 резервных копий.

Резервная копия iTunes снятая с iOS устройства содержит копию БД с историей общения в незашифрованном виде, а также все файлы, которыми обменивались пользователи. На Android устройствах в данный момент WhatsApp использует довольно жесткую политику и в резервную копию ADB не попадают пользовательские данные. Тем не менее на старых версиях WhatsApp использовалась более гибкая политика и в резервную копию ADB попадал key файл. Через протокол ADB можно, не изменяя данных из песочницы, временно понизить версию WhatsApp, затем снять резервную копию ADB в которую попадёт key файл, а затем также через протокол ADB вернуть свежую версию. Данный способ позволяет, используя штатные механизмы ОС, извлечь из песочницы key файл и с его помощью расшифровать последние семь локальных резервные копий.

При наличии доступа в интернет приложение WhatsApp автоматически производит резервное копирование БД с историей общения в облачное хранилище. На Android устройствах БД шифруется AES с использованием ключа из key файла и сохраняется в Google Drive. На iOS устройствах БД шифруется AES с использованием ключа из keychain и сохраняется в iCloud. В облачном хранилище содержится только последняя резервная копия. Для доступа к облачному хранилищу нужен специальный токен доступа от Google Drive или iCloud. Для случая, когда нет возможности извлечь токен от облачного хранилища из устройства, разработан способ, позволяющий на основе пары логин/пароль от учетной

записи Google или iCloud сгенерировать новый токен к соответствующему сервису.

Если нет возможности достать ключ от БД, то можно запросить его у сервера WhatsApp, однако для этого нужен токен от учетной записи WhatsApp. Еще на сервере WhatsApp хранятся пока не полученные пользователем сообщения, для доступа к ним также нужен WhatsApp токен. Если же возможности достать WhatsApp токен нет, то можно сгенерировать новый. Для генерации нового WhatsApp токена достаточно иметь возможность перехватить код из SMS или принять звонок. WhatsApp предоставляет пользователям возможность использовать двухфакторную аутентификацию. Если она включена, то для генерации нового WhatsApp токена помимо возможности перехватить код из SMS необходимо знать заданный пользователем PIN код. В ходе исследования для такой ситуации был разработан способ, позволяющий выключить двухфакторную аутентификацию, однако способ сработает только если устройство более 7 дней не выходило в сеть.

При исследовании механизма WhatsApp QR был выявлен еще один способ получения доступа к пользовательским данным. WhatsApp QR — механизм, позволяющий получить доступ к данным мессенджера, хранящимся на телефоне, через сервер WhatsApp. Обычно механизм WhatsApp QR применяется пользователями для того, чтобы открыть свою учетную запись на другом устройстве или компьютере. На рис. 2 изображена схема работы механизма WhatsApp QR, восстановленная посредством обратной разработки.

WhatsApp QR клиент подключается к серверу WhatsApp и авторизуется с помощью специального QR токена. Затем сервер WhatsApp отправляет специальное сообщение на мобильное устройство и между WhatsApp QR клиентом и мобильным устройством происходит обмен ключами шифрования. Далее с использованием этих ключей устанавливается защищенное соединение. Данное соединение проходит через сервер WhatsApp, но при этом сервер WhatsApp не имеет копий ключей и соответственно не может расшифровать проходящие через него данные. Через данный зашифрованный туннель WhatsApp QR клиент выкачивает с устройства БД с историей общения. Медиа-файлы хранятся на сервере WhatsApp в зашифрованном виде, ключи для их расшифровывания WhatsApp QR клиент получает от устройства через туннель.

В ходе исследования был восстановлен протокол WhatsApp QR и разработан способ, позволяющий сгенерировать специальный QR-код. Если данный QR-код будет просканирован устройством, то токен соответствующий QR-коду будет зарегистрирован на сервере WhatsApp. После этого становится возможным установление туннеля с устройством и выкачивание через него БД с историей общения. Соответственно, также реализован способ выкачивания БД из устройства с использованием WhatsApp QR токена извлеченного с компьютера.

Заключение

В работе предложена методика проведения аудита безопасности мобильных приложений, ориентированная на выявление способов получения доступа к пользовательским данным. Специализированные комплексы для проведения автоматизированной компьютерно-технической экспертизы (КТЭ) мобильных устройств для получения доступа к пользовательским данным активно используют уязвимости в мобильных ОС. Таким образом способы, которые применяет эксперт при использовании специализированного комплекса, аналогичны возможностям потенциального нарушителя из предложенной методики. Исходя из этого описанная в работе методика может быть использована при разработке комплексов для проведения КТЭ экспертизы мобильных устройств. Результаты аудита безопасности приложения WhatsApp, проведенного по данной методике, позволили выявить ряд весьма нетривиальных способов получения доступа к пользовательским данным, большинство из которых могут быть эффективно использованы при проведении КТЭ.

ЛИТЕРАТУРА

1. Бессонов А.А. О некоторых возможностях современной криминалистики в работе с электронными следами // Вестник Университета имени О. Е. Кутафина. 2019. № 3 (55). URL: https://cyberleninka.ru/article/n/o-nekotoryh-vozmozhnostyah-sovremennoy-kriminalistiki-v-rabote-s-elektronnymi-sledami (дата обращения: 10.06.2019).
2. Ярмак К. В. Инновационные направления развития криминалистических средств и методов // Вестник экономической безопасности. 2015. № 2. URL: https://cyberleninka.ru/article/n/innovatsionnye-napravleniya-razvitiya-kriminalisticheskih-sredstv-i-metodov (дата обращения: 10.06.2019).
3. Александров Я.А., Сафин Л. К., Трошина К. Н., Чернов А. В. Статический бинарный анализ мобильных приложений для платформы Android по требованиям информационной безопасности // Вестник Московского университета. Серия 15. Вычислительная математика и кибернетика. 2016. № 3. URL: https://cyberleninka.ru/article/n/staticheskiy-binarnyy-analiz-mobilnyh-prilozheniy-dlya-platformy-android-po-trebovaniyam-informatsionnoy-bezopasnosti (дата обращения: 10.06.2019).

4. Сафин Л.К., Чернов А. В., Александров Я. А., Трошина К. Н. Исследование информационной защищенности мобильных приложений // Вопросы кибербезопасности. 2015. № 4 (12). URL: https://cyberleninka.ru/article/n/issledovanie-informatsionnoy-zaschischennosti-mobilnyh-prilozheniy (дата обращения: 10.06.2019).
5. OWASP Mobile Security Project — OWASP [Электронный ресурс] URL: https://www.owasp.org/index.php/OWASP_Mobile_Security_Project (дата обращения: 10.06.2019).
6. The Web Application Security Consortium / Threat Classification [Электронный ресурс] URL: http://projects.webappsec.org/w/page/13246978/Threat%20Classification (дата обращения: 10.06.2019).

© Карондеев Андрей Михайлович (karondeev@oxygensoftware.com), Клюев Даниэль Витальевич (kluev@oxygensoftware.com).
Журнал «Современная наука: актуальные проблемы теории и практики»

Московский государственный технический университет им Н.Э. Баумана