



Уязвимости и **угрозы** мобильных приложений

2019

ptsecurity.com

Содержание

| | |
|--|----|
| Введение | 2 |
| Резюме | 3 |
| Как работают мобильные приложения | 3 |
| Уязвимости клиентских частей | 4 |
| Уязвимости серверных частей | 12 |
| Угрозы мобильных приложений | 15 |
| Рекомендации для пользователей | 16 |
| Какие риски подстерегают пользователей | 20 |
| Выводы | 24 |
| Об исследовании | 25 |

Г Введение

Согласно статистике, в 2018 году мобильные приложения были загружены на устройства пользователей более 200 миллиардов раз. По данным Marketing Land, 57% времени, проведенного в цифровом пространстве, — это время, потраченное на программы в смартфонах или планшетах. Мобильные устройства прочно вошли в нашу жизнь: мессенджеры, банкинг, бизнес-приложения, личные кабинеты сотовых операторов — при современном ритме жизни мы используем эти приложения практически ежедневно. Согласно данным Juniper Research, общее число пользователей мобильных банковских приложений приближается к двум миллиардам, что составляет порядка 40% всего взрослого населения. Мобильным банком пользуется каждый третий (34%) россиянин старше 18 лет.

Разработчики уделяют большое внимание дизайну программных продуктов для наших гаджетов, стараясь сделать их максимально удобными. Люди охотно устанавливают мобильные приложения и регистрируются в них, но мало кто из рядовых пользователей задумывается о безопасности данных, которые мы доверяем производителям этих приложений.

Эксперты Positive Technologies регулярно проводят анализ защищенности мобильных приложений. В данном отчете представлена статистика, полученная в ходе работ по тестированию защищенности мобильных приложений для iOS и Android в 2018 году.

Резюме

- Уязвимости высокого уровня риска обнаружены в 38% мобильных приложений для iOS и в 43% приложений для платформ под управлением Android.
- Большинство проблем безопасности являются общими для обеих платформ. Небезопасное хранение данных — основной недостаток, он выявлен в 76% мобильных приложений. Под угрозу попадают пароли, финансовая информация, персональные данные и личная переписка.
- Хакеру редко требуется физический доступ к смартфону, чтобы украсть данные: 89% уязвимостей могут быть проэксплуатированы с использованием ВПО.
- Большинство недостатков связаны с ошибками в механизмах защиты (74% и 57% — для приложений на iOS и Android соответственно, 42% — для серверных частей). Такие уязвимости закладываются еще на этапе проектирования, а их устранение потребует внесения существенных изменений в код.
- Риски возникают не только из-за отдельно взятых уязвимостей на клиенте или сервере; зачастую угрозы обусловлены несколькими, казалось бы, незначительными недостатками в разных частях мобильного приложения, которые в совокупности могут приводить к серьезным последствиям, вплоть до финансового ущерба для пользователей и репутационных потерь для производителя.
- Успех кибератаки на мобильное приложение напрямую зависит от того, насколько внимательно сам пользователь относится к сохранности своих данных. Предпосылкой ко взлому могут стать повышенные привилегии или загруженные из неофициального источника программы.

Как работают мобильные приложения

Разработка мобильных приложений в тренде, ее технологии непрерывно развиваются. Большинство современных решений имеют клиент-серверную архитектуру. Клиент работает под управлением мобильной операционной системы; чаще всего это Android или iOS. Клиентская часть загружается на устройство из так называемого магазина приложений — специализированной площадки, где разработчики размещают свои системы. С точки зрения обычного пользователя, установленная на смартфон программа — это и есть мобильное приложение, ведь именно с ней он взаимодействует напрямую: совершает покупки, оплачивает счета, просматривает почту. Но в действительности есть еще один компонент, который принято называть сервером.

Серверная часть находится на стороне разработчика. Зачастую ее роль выполняет то же программное обеспечение, которое отвечает за генерацию и обработку контента на сайте. Другими словами, чаще всего серверная часть — это веб-приложение, которое взаимодействует с мобильным клиентом через интернет посредством специального интерфейса (API). Сервер по праву можно считать главной частью: здесь обрабатывается и хранится информация; помимо этого, он отвечает за синхронизацию пользовательских данных между устройствами.

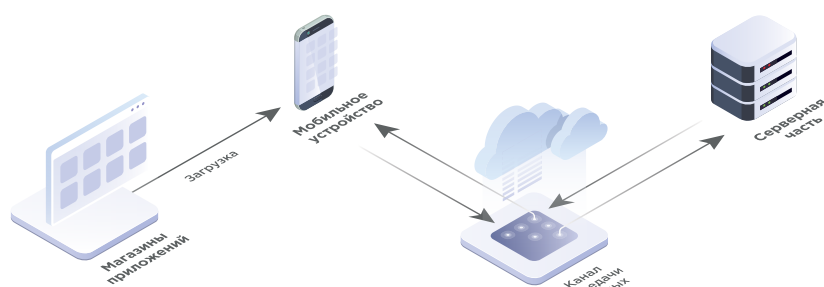


Рисунок 1. Клиент-серверное взаимодействие в мобильном приложении

Современные версии мобильных ОС имеют разнообразные встроенные механизмы защиты. Так, по умолчанию всем установленным программам разрешено работать только с файлами в собственных домашних каталогах, а права пользователя не позволяют редактировать какие-либо системные файлы. Несмотря на это, ошибки, допущенные разработчиками при проектировании и написании кода мобильных приложений, приводят к брешам в защите и открывают двери киберпреступникам.

Комплексная проверка безопасности мобильного приложения подразумевает поиск уязвимостей как в клиентской, так и в серверной частях; кроме того, не менее важно оценить защищенность канала передачи данных между ними. В данном исследовании мы рассмотрим все эти аспекты. Также мы расскажем об угрозах, которые подстерегают пользователей, в том числе о тех, которые обусловлены взаимодействием между клиентской и серверной частями мобильных приложений. С методикой исследования и портретом участников можно ознакомиться в конце отчета.

Уязвимости клиентских частей



Приложения для Android с критически опасными уязвимостями встречаются несколько чаще, чем программы для iOS (43% против 38%). Однако эта разница несущественна, и общий уровень защищенности клиентских частей мобильных приложений для Android и iOS примерно одинаков. Около трети всех уязвимостей в клиентских частях мобильных приложений для обеих платформ имеют высокий уровень риска.

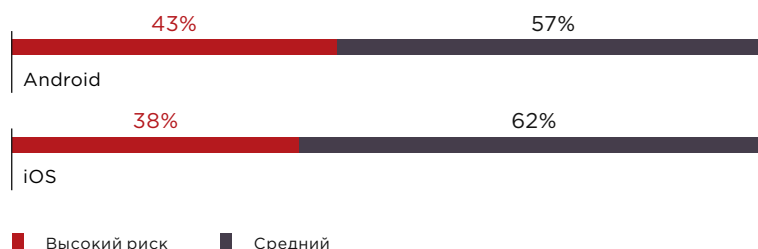


Рисунок 2. Максимальная степень риска уязвимостей (указана доля клиентских частей)

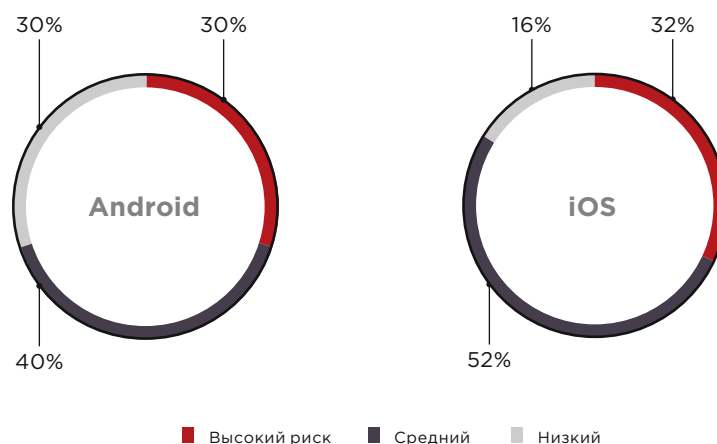


Рисунок 3. Доля уязвимостей различной степени риска

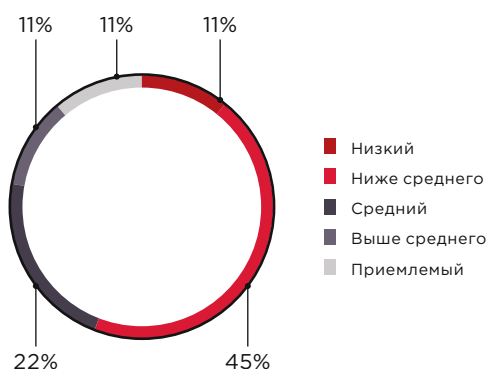


Рисунок 4. Уровень защищенности клиентских частей (доля систем)

38% Android
22% iOS

Доли приложений с небезопасным межпроцессным взаимодействием

Небезопасное использование межпроцессного взаимодействия — распространенная критически опасная уязвимость, которая позволяет злоумышленнику удаленно получить доступ к данным, обрабатываемым в уязвимом мобильном приложении. Остановимся на ней более подробно.

Операционная система Android предоставляет механизм взаимодействия компонентов приложения посредством сообщений (объектов класса Intent). Если для обмена сообщениями используются широковещательные рассылки, то чувствительные данные, содержащиеся в этих сообщениях, могут быть скомпрометированы вредоносным ПО, зарегистрировавшим свой обработчик широковещательных сообщений (компонент BroadcastReceiver).

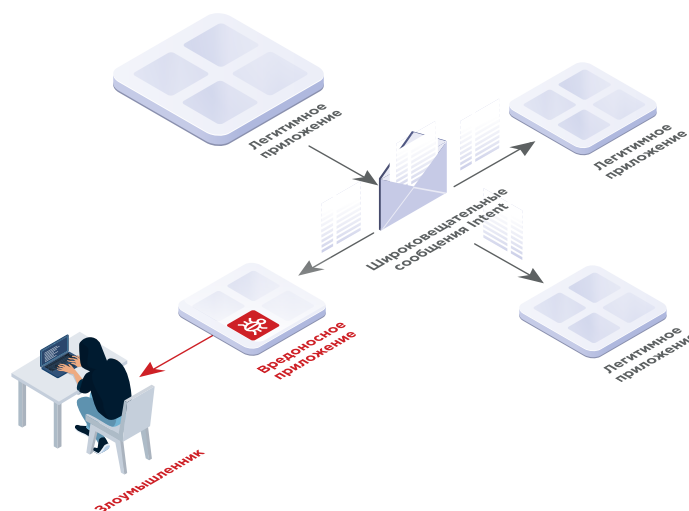


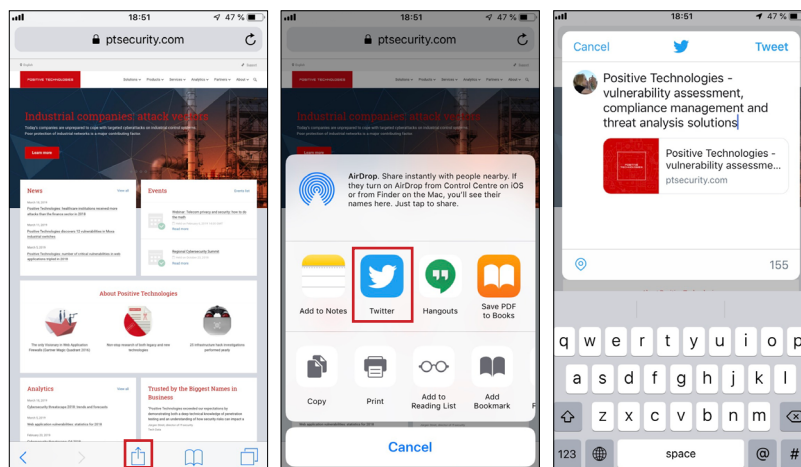
Рисунок 5. Схема небезопасного межпроцессного взаимодействия в ОС Android



Рекомендация для разработчиков

Используйте компонент LocalBroadcastManager для отправки и получения широковещательных сообщений, не предназначенных для сторонних приложений

Межпроцессное взаимодействие в iOS-приложениях, как правило, запрещено, однако существуют случаи, когда оно необходимо. В iOS версии 8 компания Apple представила новую технологию под названием App Extensions, с помощью которой приложения могут делиться своими функциональными возможностями с другими установленными на устройстве приложениями (например, мобильные приложения для социальных сетей позволяют быстро делиться контентом из браузера).



Вызывающее приложение (Host App)

В данном случае браузер Safari

Основное приложение (Containing App)

В данном случае Twitter

Расширение (App Extension)

В данном случае расширение для Twitter

Рисунок 6. Пример расширения для Twitter

Для организации взаимодействия между расширением (App Extension) и основным приложением (Containing App) разработчики нередко используют механизм Deep Linking. Вызов приложения при этом осуществляется посредством зарегистрированной в системе собственной схемы URL. В процессе установки основное приложение регистрирует себя в качестве обработчика схем, указанных в файле Info.plist. Подобного рода схемы не привязаны к приложению: если на устройстве присутствует вредоносное ПО, которое обрабатывает ту же схему URL, то невозможно предсказать, какое именно приложение будет запущено. Это открывает злоумышленнику возможности для проведения фишинговых атак и кражи учетных данных пользователей.



Рекомендация для разработчиков

При необходимости использовать ссылки для взаимодействия между компонентами приложения используйте защищенный механизм Universal Links

Уязвимость небезопасного межпроцессного взаимодействия закладывается на этапе проектирования интерфейсов взаимодействия компонентов приложения и относится к ошибкам в реализации механизмов защиты. Ошибки в механизмах защиты стали причиной 74% уязвимостей в приложениях для iOS и 57% уязвимостей для платформ Android.

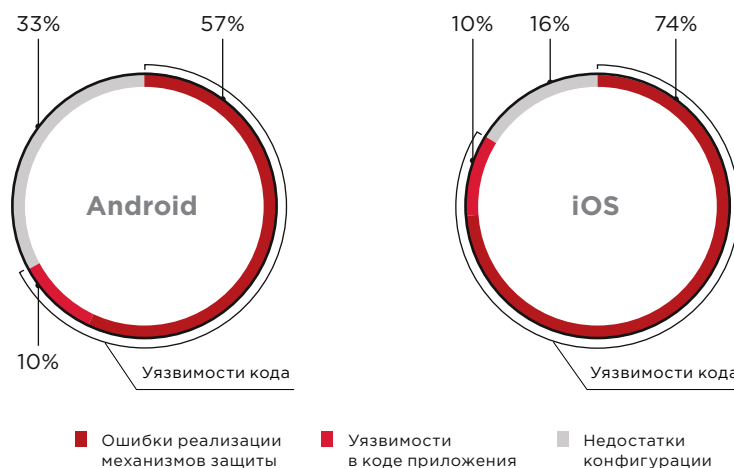


Рисунок 7. Доля уязвимостей разных типов



Производитель виртуальной клавиатуры Al.type собирал чувствительные данные с мобильных устройств. Об этом стало известно после утечки базы данных 31 млн пользователей

В 2018 году, анализируя мобильные приложения для iOS, мы сталкивались с такой ошибкой в механизмах их защиты, как отсутствие ограничений на использование установленных пользователем клавиатурных расширений. Компания Apple позволила использовать клавиатуры сторонних производителей начиная с iOS версии 8, в это время такая возможность уже существовала в Android. Стоит отметить, что iOS накладывает более строгие ограничения на использование клавиатуры, чем Android; однако Apple не может контролировать, что делают разработчики клавиатур с данными нажатия клавиш, если пользователь разрешает этим приложениям сетевое взаимодействие.



Рекомендация для разработчиков

Реализуйте метод `shouldAllowExtensionPointIdentifier` класса `UIApplicationDelegate`, запрещающий использовать клавиатурные расширения в приложении



Если в приложении предполагается ввод чувствительных данных (например, финансовой информации), реализуйте собственную клавиатуру. Это защитит от атак с подменой системной клавиатуры

25%

приложений для платформ Android позволяют создавать резервную копию при подключении мобильного устройства к компьютеру

Каждая третья уязвимость мобильных приложений для Android связана с недостатками конфигурации. Например, при анализе файла `AndroidManifest.xml` наши специалисты нередко обнаруживают директиву `android:allowBackup` в значении `true`. Это позволяет создавать резервную копию данных приложения при подключении к компьютеру. Недостатком может воспользоваться злоумышленник и получить данные приложения даже без прав пользователя root.



Рекомендация для разработчиков

Запретите создание резервной копии данных приложения при подключении мобильного устройства к компьютеру, установив директиву `android:allowBackup="false"` в значение `false`

```
<manifest >
...
<application android:allowBackup="false" >
...
</application>
</manifest>
```

Рисунок 8. Запрет на резервное копирование данных в файле AndroidManifest.xml

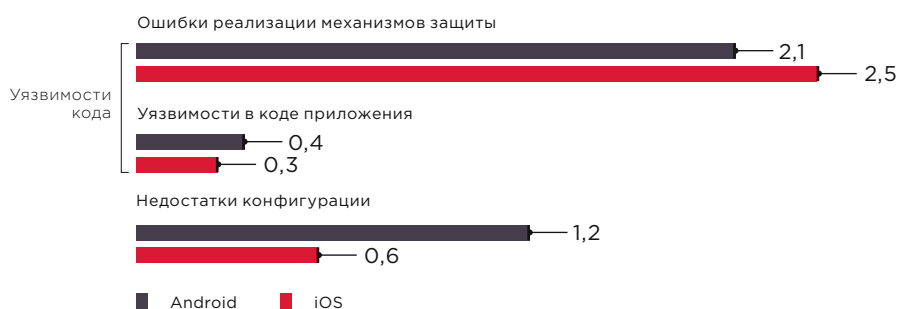


Рисунок 9. Среднее число уязвимостей на одно клиентское приложение

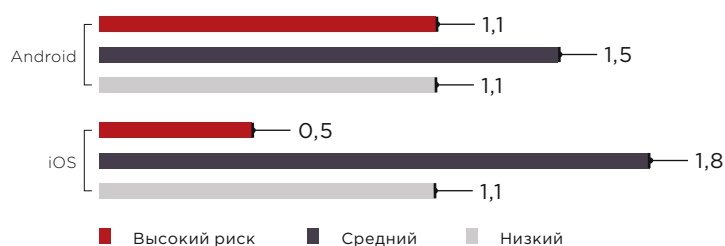


Рисунок 10. Среднее число уязвимостей на одно клиентское приложение

В рамках тестирования защищенности наши специалисты ищут характерные уязвимости приложений для каждой платформы. В то же время мы заметили, что в большинстве случаев разработчики допускают однотипные ошибки в системах как для Android, так и для iOS, поэтому в данном исследовании мы показываем общую статистику уязвимостей, не разделяя платформы.

На мобильном устройстве могут сохраняться разнообразные данные, например данные геолокации, персональные данные, личная переписка, учетные записи, финансовая информация, однако безопасности их хранения в мобильных приложениях не всегда уделяется должное внимание. Небезопасное хранение данных занимает второе место в рейтинге [OWASP Mobile Top 10-2016](#); эта уязвимость была обнаружена в 76% мобильных приложений.

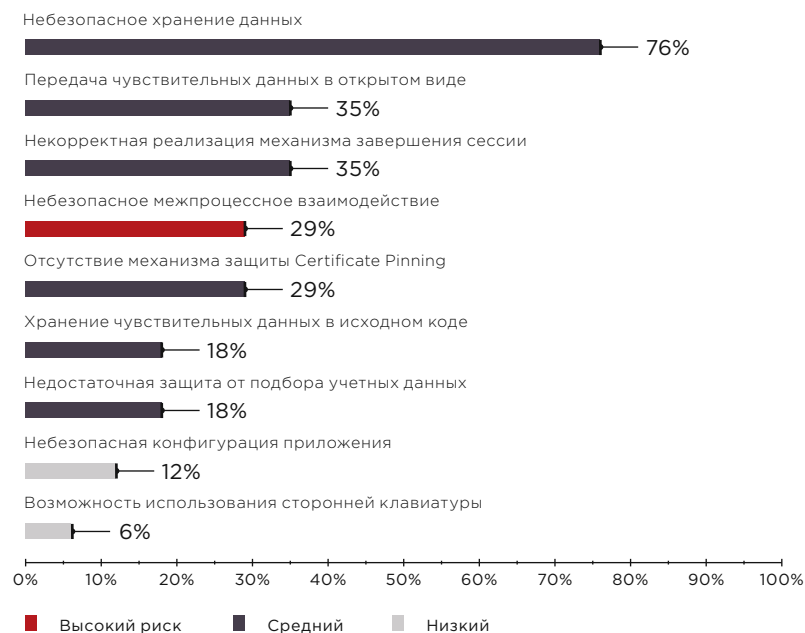


Рисунок 11. Уязвимости мобильных приложений (доля клиентских частей)

В мобильных устройствах есть возможность просмотра недавно использованных программ и быстрого переключения между ними. Для этого, когда пользователь сворачивает приложение, операционная система делает снимок состояния экрана. Прямой доступ к снимкам есть только на устройствах с административными привилегиями. Важно предусмотреть вариант, при котором на скриншотах экрана окажутся чувствительные данные; например, в случае с мобильным банком на изображении могут попасть данные платежной карты. Эти изображения могут быть похищены, например если устройство заражено вредоносным ПО.

Рекомендация для разработчиков

Используйте специальное фоновое изображение, которое будет перекрывать экран приложения, содержащий чувствительную информацию

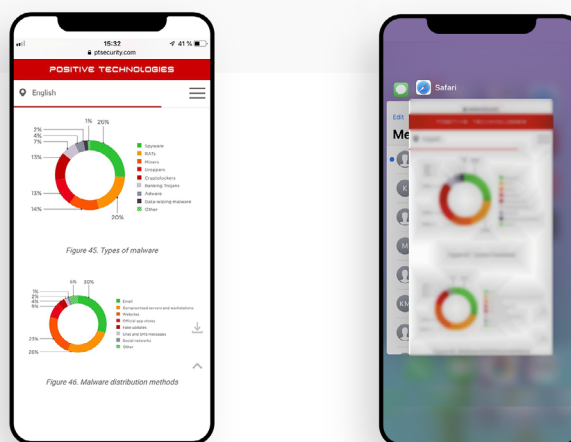


Рисунок 12. Пример защиты данных на снимке экрана при сворачивании приложения

41%

мобильных приложений
осуществляют проверку
аутентификационных данных
на стороне клиента

Во многих мобильных приложениях для аутентификации используются четырех- или шестизначные PIN-коды. Существует несколько вариантов реализации механизма проверки их корректности в момент входа пользователя в приложение; один из них — проверка на стороне клиента. Это небезопасно, поскольку такой способ верификации PIN-кода предполагает его хранение на мобильном устройстве, что повышает риск утечки. Аутентификационные данные небезопасно хранятся в 53% мобильных приложений.

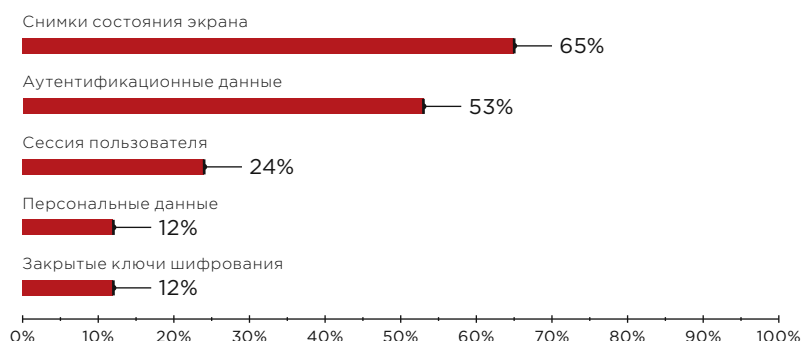


Рисунок 13. Топ-5 утечек данных в клиентских частях (доля уязвимых приложений)

Проверка введенного PIN-кода или пароля должна происходить на сервере, куда аутентификационные данные следует передавать в виде хеш-значений. Для работы хеш-функции необходима соль — набор случайных символов, которые позволяют усилить безопасность. Зачастую наши эксперты обнаруживают соль и другие чувствительные данные в исходном коде, что снижает уровень защищенности приложения. Хорошей альтернативой хранению соли в коде может стать ее динамическая генерация при регистрации пользователя на основе вводимых им данных, однако это решение можно считать надежным только при условии, что эти данные обладают высокой энтропией.

Рекомендация для разработчиков

Для аутентификации в приложениях на современных устройствах часто используются биометрические механизмы (Touch ID или Face ID). В этом случае PIN-код хранится на устройстве. Локальное хранение чувствительных данных допускается только в специальных каталогах с шифрованием. В Android существует хранилище ключей — Keystore, в iOS — связка ключей Keychain

Уязвимости серверных частей

Как мы отмечали выше, серверные части мобильных приложений по сути являются веб-приложениями. Об уязвимостях веб-приложений мы рассказали в отдельном [исследовании](#). Тем не менее рассмотрим уязвимости серверных частей мобильных приложений более подробно.

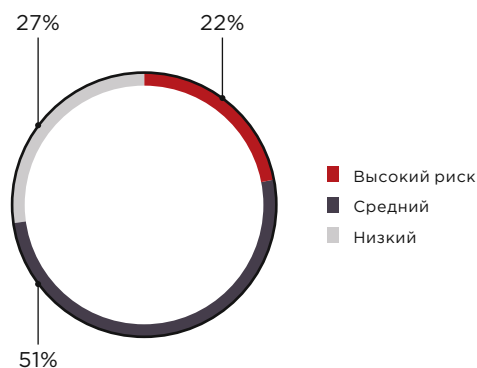


Рисунок 14. Доля уязвимостей различной степени риска



В августе 2018 года злоумышленники похитили персональные данные 20 000 пользователей мобильного приложения авиакомпании Air Canada

По данным компании McAfee, количество вредоносного ПО для мобильных устройств растет: ежеквартально выявляется от 1,5 до 2 млн новых экземпляров, а к концу 2018 года общий объем составил более 30 млн экземпляров. Постоянный рост числа и разнообразия вредоносного ПО для мобильных устройств существенно повышает популярность атак на клиентские части, и серверные уязвимости перестали быть главной угрозой безопасности мобильных приложений. Еще в 2012 году категория Weak Server Side Controls занимала второе место в рейтинге [OWASP Mobile Top 10](#); в рейтинге 2016 года серверные уязвимости были исключены из десятки самых распространенных угроз. Тем не менее риски, связанные с недостатками серверов, сохраняются, и в мире продолжают случаться громкие утечки данных из-за серверных уязвимостей. Как показывают результаты нашего исследования, серверные части не менее уязвимы, чем клиентские: 43% имеют низкий или крайне низкий уровень защищенности, при этом 33% содержат критически опасные уязвимости.

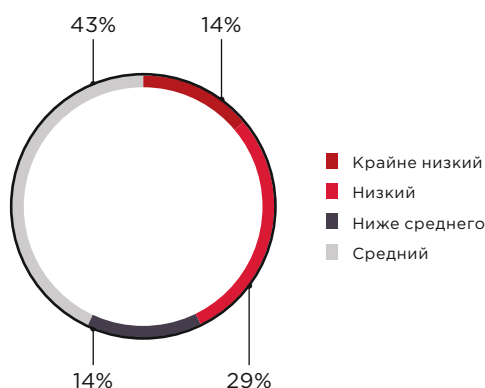


Рисунок 15. Уровень защищенности серверных частей (доля систем)

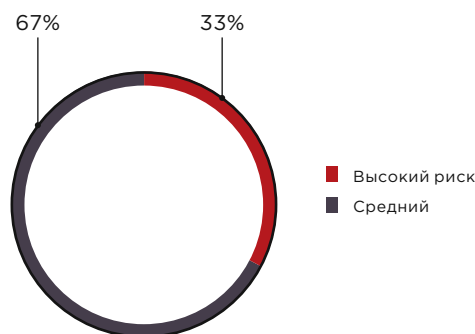


Рисунок 16. Максимальная степень риска уязвимостей (доля серверных частей)

Серверные части мобильных приложений в равной степени содержат уязвимости как в коде самого приложения, так и в механизмах его защиты. В числе последних стоит отметить недостатки реализации двухфакторной аутентификации. Рассмотрим уязвимость, которую мы обнаружили в одном из исследованных приложений. Если послать сразу друг за другом, с минимальным интервалом, два одинаковых запроса к серверу, то одноразовые пароли отправляются пользователю приложения на устройство и через push-уведомления, и в SMS на привязанный номер телефона. В результате злоумышленник, имея возможность перехватывать SMS-сообщения, может совершать операции от имени законного пользователя, например переводить деньги с его счета на свой.

Рекомендация для разработчиков

Нет необходимости дублировать одноразовые пароли или коды подтверждения в SMS-сообщениях и push-уведомлениях. Используйте выбранный пользователем канал получения паролей

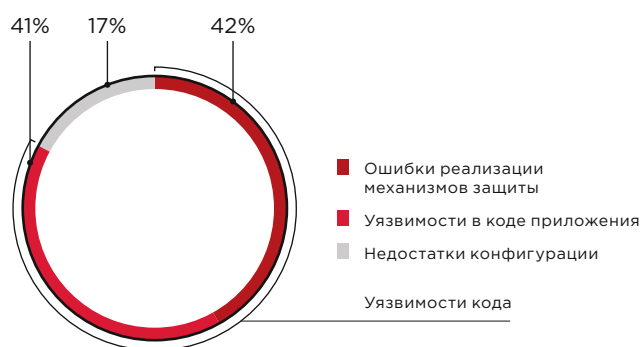


Рисунок 17. Доли уязвимостей разных типов

В среднем каждая серверная часть содержит пять уязвимостей кода и одну уязвимость конфигурации. Среди недостатков конфигурации мы отмечаем разглашение чувствительной информации в сообщениях об ошибках, раскрытие информации о версиях используемого ПО в HTTP-заголовках, а также доступность метода TRACE.

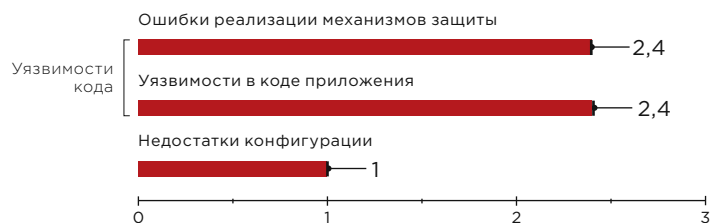


Рисунок 18. Среднее число уязвимостей на одну серверную часть

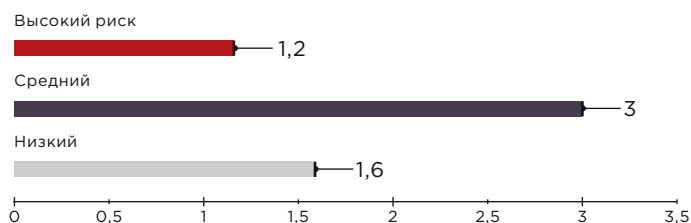


Рисунок 19. Среднее число уязвимостей на одну серверную часть

Возможность обработки TRACE-запросов в сочетании с уязвимостью «Межсайтовое выполнение сценариев» (Cross-Site Scripting, XSS) может помочь злоумышленнику украсть значения Cookie и получить доступ к приложению. Поскольку, как правило, серверная часть мобильного приложения и веб-сайт имеют общий программный код, уязвимость «Межсайтовое выполнение сценариев» в серверной части открывает возможности для проведения атак на пользователей веб-приложения.



Рекомендация для разработчиков

Метод TRACE может быть использован для обхода защиты Cookie флагом httpOnly.
Отключите обработку TRACE-запросов

Недостаточная авторизация выявлена в 43% серверных частей. Это один из самых распространенных недостатков высокого уровня риска, его доля составила 45% от всех критически опасных уязвимостей.

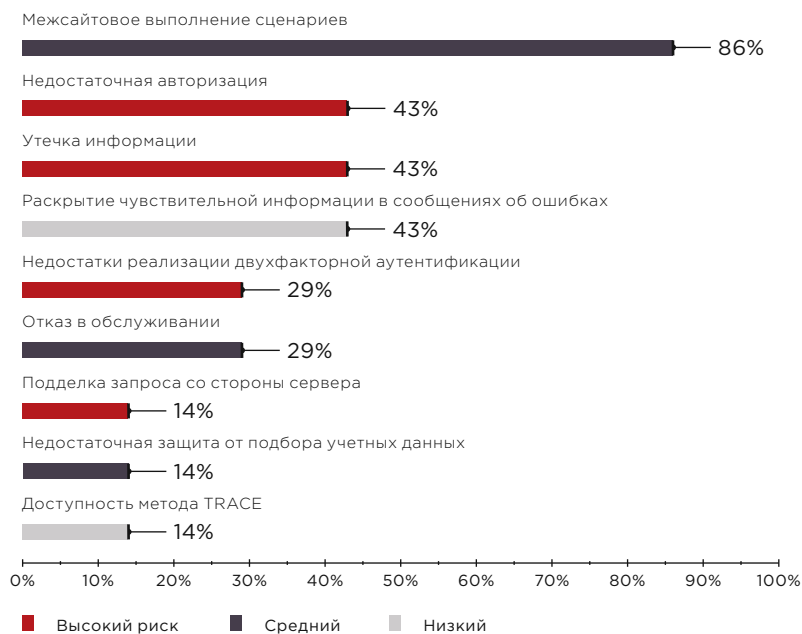


Рисунок 20. Наиболее распространенные уязвимости серверных частей мобильных приложений (доля систем)

Утечка информации — еще одна распространенная проблема серверных частей мобильных приложений, которая может привести к серьезным последствиям. Например, в одном из исследованных нами приложений при создании чата в ответе сервера были обнаружены имя, фамилия и номер телефона выбранного собеседника. Другой пример раскрытия критически важных данных — идентификатор сессии в ссылке на обрабатываемый в мобильном приложении документ. Если злоумышленник убедит пользователя отправить ему ссылку на документ, в которой содержится идентификатор сессии, то он сможет совершать действия в системе от лица этого пользователя.

Если сервер мобильного приложения обрабатывает числовые входные данные (например, координаты точки на карте), то необходимо предусмотреть ограничения. В отсутствие ограничений злоумышленник может указать произвольные координаты для поиска объекта на карте. Некорректное задание координат приводит к большим задержкам во времени ответа серверной части и, как следствие, — к отказу в обслуживании. Нарушение работы приложения отрицательно сказывается на репутации его владельца.

Угрозы мобильных приложений

Почти все исследованные приложения находятся под угрозой доступа к ним хакеров. В разделе про клиентские уязвимости мы отмечали, что самая распространенная проблема мобильных приложений — небезопасное хранение данных. Каким образом информация может попасть в руки злоумышленников? Самый распространенный сценарий — это заражение устройства вредоносным ПО, вероятность которого увеличивается в разы на устройствах с административными привилегиями (root или jailbreak). Однако вредоносное ПО может повышать права самостоятельно. Например, шпионский троян ZNIU с этой целью оперирует эксплойтом к нашумевшей уязвимости Dirty COW

29%

серверных частей содержат уязвимости, которые могут привести к нарушению работы приложения

(CVE-2016-5195). Попад на устройство жертвы, вредонос может запрашивать разрешения на доступ к пользовательским данным, а получив разрешение, передавать данные злоумышленникам. Так, эксперты TheBestVPN изучили 81 VPN-приложение из официального магазина Google Play и пришли к выводу, что многие из них запрашивают настоящие разрешения.

Рекомендации для пользователей

Внимательно относитесь к уведомлениям от приложений о запросе доступа к каким-либо функциям или данным. Не стоит предоставлять разрешение на доступ, если есть сомнение в его необходимости для нормального функционирования приложения

Кроме того, смартфон легко потерять или он может быть украден. Несмотря на то, что по умолчанию мобильные ОС требуют установки пароля, это требование можно отключить, что и делают некоторые пользователи. В этом случае злоумышленник, получивший физический доступ к устройству, подключит его к своему компьютеру и, используя специальные утилиты, извлечет из памяти устройства чувствительную информацию. Например, если в Android включено резервное копирование, то, используя инструмент Android Debug Bridge (ADB), можно попытаться получить данные приложения из резервной копии. Если есть привилегии root, то извлечь данные можно даже с отключенным резервированием. На устройствах Apple с jailbreak пользователи часто оставляют для SSH стандартную учетную запись (root:alpine), что позволяет злоумышленнику скопировать данные приложения на свой компьютер, подключившись по SSH. Угроза имеет особую актуальность в случае с корпоративными смартфонами или планшетами, которыми пользуются несколько сотрудников, знающие пароль от устройства.



Рисунок 21. Возможные сценарии кражи пользовательских данных из мобильных приложений

18%

приложений не ограничивают
число попыток ввода
аутентификационных данных

Иногда для взлома мобильного приложения не требуются ни ВПО, ни хакерские утилиты. Например, в приложении могут отсутствовать ограничения на число неуспешных попыток ввода PIN-кода. Другой пример: ограничения на число попыток ввода пароля накладываются только со стороны клиентской части и при перезапуске приложения счетчик попыток обнуляется. Оба этих недостатка позволяют злоумышленнику вводить пароль неограниченное число раз.

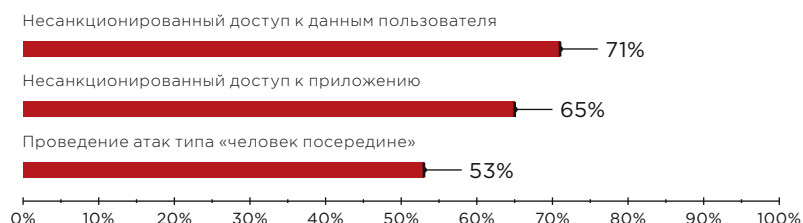


Рисунок 22. Угрозы для клиентских частей (доля систем)

Рекомендации для пользователей

Устанавливайте в качестве PIN-кода случайные комбинации цифр. Дата рождения, номер телефона, серия и номер паспорта — не лучший выбор. Отдавайте предпочтение биометрическим механизмам аутентификации (отпечаток пальца, голос, лицо), если они поддерживаются вашим устройством

Рекомендация для разработчиков

Счетчик числа попыток аутентификации должен быть реализован как на стороне сервера, так и на стороне клиента

Атаки на пользователей возможны из-за уязвимостей в серверных частях. Уязвимость «Межсайтовое выполнение сценариев» обнаружена в 86% серверов. Это самая распространенная веб-уязвимость. С ее помощью злоумышленники могут украсть учетные данные жертвы, например файлы Cookie, используя вредоносный скрипты. Но уязвимость может представлять опасность и для мобильных приложений, если в них используются компоненты с поддержкой HTML и JavaScript. Например, WebView — это системный компонент, который позволяет приложениям на Android отображать веб-контент непосредственно внутри приложения^{*}; аналогичные компоненты есть и в iOS: UIWebView и WKWebView.

^{*} В начале 2019 года наши эксперты обнаружили в этом компоненте уязвимость (CVE-2019-5765), которая позволяет получать доступ к данным пользователей Android через установленное вредоносное приложение или приложение с мгновенным запуском (Android instant apps).

Уязвимости внедрения почтовых заголовков или HTML-тегов открывают возможности для проведения фишинговых атак. В результате внедрения почтовых заголовков нарушитель потенциально может отправлять письма пользователям приложения от имени любого сотрудника компании — владельца мобильного приложения.

Рекомендации для пользователей

Проявляйте бдительность, просматривая электронную почту. Внимательно проверяйте ссылки, по которым собираетесь перейти, даже если вы являетесь клиентом компании, от которой пришло письмо. Если в ссылке перепутаны буквы, это явный признак того, что письмо отправил злоумышленник. Помните, что сотрудники банка никогда не просят клиентов сообщить полные данные платежных карт

Рекомендация для разработчиков

На стороне сервера обязательно должна осуществляться фильтрация вводимых пользователем данных. Для спецсимволов рекомендуется использовать HTML-кодировку

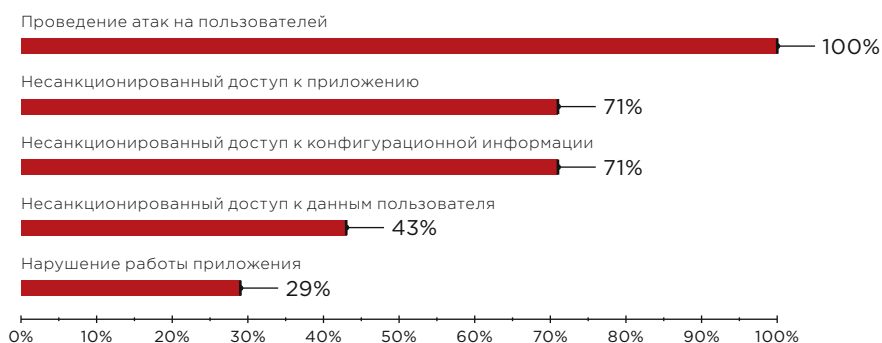


Рисунок 23. Топ-5 угроз для серверных частей (доля систем)

18%

приложений содержат уязвимости, позволяющие проводить атаки на сессию

Зачастую угрозы обусловлены сочетанием недостатков клиентской части и сервера. Например, представим, что при выходе пользователя из приложения идентификатор сессии не удаляется на стороне клиента и отправляется на сервер с каждым новым запросом, в том числе при повторной аутентификации. Сервер, в свою очередь, не проверяет время жизни сессии и после процедуры аутентификации вновь активирует старый идентификатор. В этом случае злоумышленник, получивший значение идентификатора сессии, может проводить атаки, выполняя всевозможные действия от лица правомерного пользователя приложения.

Рекомендации для разработчиков

Время жизни сессии пользователя обязательно следует ограничивать, а удаление идентификатора сессии должно происходить не только на клиенте, но и на стороне сервера. Сервер должен создавать новую сессию для пользователя каждый раз, когда требуется аутентификация

50% Android
22% iOS

Доли приложений с небезопасной передачей данных

Канал связи между клиентской частью приложения и сервером также может быть уязвим. Если клиентская часть взаимодействует с сервером по открытому протоколу HTTP, злоумышленник сможет перехватить чувствительные данные. Например, в случае с мобильным банком под угрозу попадает вся платежная информация. Чтобы предотвратить перехват, используется защищенный протокол HTTPS. В этом случае безопасность соединения обеспечивается за счет шифрования передаваемой информации. На устройстве при этом хранятся сертификаты — специальные файлы, благодаря которым клиент «знает» имя сервера, куда передаются данные.

Однако даже защищенное с помощью HTTPS клиент-серверное взаимодействие не всегда безопасно. Дело в том, что сертификаты хранятся на устройстве в общем для всех приложений хранилище. Вредоносное ПО может установить на смартфон жертвы корневой сертификат злоумышленника. В этом случае все сертификаты, удостоверяемые поддельным корневым сертификатом, будут считаться доверенными. Теперь если жертва подключится к сетевому оборудованию (например, к Wi-Fi-роутеру), который контролируется злоумышленником, он сможет провести атаку «человек посередине», другими словами — вклиниться в канал связи и прослушивать трафик, что означает полную компрометацию передаваемых данных.

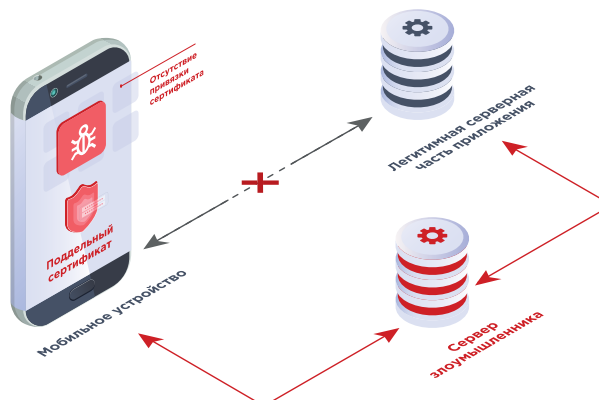


Рисунок 24. Схема атаки «человек посередине» в случае отсутствия привязки сертификата

Рекомендация для разработчиков

Для максимальной безопасности взаимодействия клиентской и серверной частей мы рекомендуем использовать технологию Certificate Pinning, которая предполагает внедрение сертификата сервера непосредственно в код мобильного приложения. Приложение становится независимым от системного хранилища сертификатов. Это позволяет предотвратить атаку «человек посередине»

18%

мобильных приложений
содержат небезопасные
внешние ссылки

Помимо небезопасного обмена данными между клиентской и серверной частями, приложение может содержать ссылки для передачи информации по незащищенному протоколу HTTP во внешнюю среду. Проблема небезопасной передачи данных чаще встречается в ОС Android. Начиная с версии 9 в iOS была встроена технология App Transport Security, которая запрещает работу по незащищенному каналу по умолчанию. Однако разработчик может явно указать исключения — адреса, с которыми разрешено взаимодействие по открытому протоколу. Иногда это требуется на этапе отладки и тестирования приложения, но зачастую небезопасные ссылки попадают в продуктивные версии систем.

```
public interface BannerNetworkService {
    @defpackage.amv(a = "http://d.../show/p/354.json")
    defpackage.amv(a = "http://d.../show/p/370.json")
    defpackage.amv(a = "http://d.../show/p/370.json")
    defpackage.amv(a = "http://d.../show/p/370.json")
}
```

Рисунок 25. Небезопасные ссылки, обнаруженные в исходном коде мобильного приложения

Какие риски подстерегают пользователей

Как показывают результаты нашего исследования, все мобильные приложения уязвимы. В отдельных случаях злоумышленник может взломать устройство только тогда, когда оно попадет к нему в руки, но в целом физический доступ обычно не требуется. Каждое мобильное приложение содержит хотя бы одну уязвимость, которую злоумышленник может проэксплуатировать удаленно с использованием вредоносного ПО.

Иногда хакеру нужен полный доступ к файловой системе: jailbreak в iOS или права пользователя root в Android. Однако и это не всегда является проблемой для злоумышленника. Многие владельцы мобильных устройств намеренно повышают свои привилегии в ОС, стремясь обойти различные ограничения, например чтобы установить пиратское ПО или изменить внешний вид графического интерфейса. По данным исследователей, 8% пользователей iOS установили

jailbreak, 27% устройств на платформе Android обладают правами root. Устройства с максимальными правами подвержены повышенному риску, поскольку привилегиями может воспользоваться вредоносное программное обеспечение. Например, зловард [KeyRaider](#), который распространялся через магазин приложений для устройств с jailbreak, похищал учетные записи, сертификаты и ключи шифрования; от него пострадали 225 тысяч пользователей iOS.



Рекомендация для пользователей

Не повышайте привилегии до административных. Это открывает доступ к файловой системе и отключает механизмы защиты ваших данных

Понимая масштаб проблемы заражения мобильных устройств вредоносным ПО, компании Google и Apple активно противодействуют киберпреступникам. Для защиты от хакеров Google предлагает использовать набор сервисов Google Play Protect, который сканирует приложения не только на Android-устройствах пользователей, но и в официальном каталоге Google Play. Для противодействия распространению вредоносного ПО через App Store в компании Apple проводят обязательный ручной анализ предлагаемых разработчиками приложений.

Тщательная проверка снижает количество вредоносных приложений, хотя и не исключает их появления, и заразиться вредоносным ПО можно даже через официальные магазины приложений. Так, хакерам удалось загрузить в App Store 39 вредоносных программ с использованием [XcodeGhost](#) — поддельной версии легитимного инструмента Xcode, используемого разработчиками программ для устройств Apple. Другой пример — банковский троян [Anubis](#), который успешно обходил проверки безопасности не только в Google Play, но и в системе защиты ОС Android.



Рекомендация для пользователей

Регулярно устанавливайте обновления ОС и приложений. Если вы обладатель устройства с root или jailbreak, помните, что ваше устройство может не обновляться автоматически

Заражение вредоносным ПО через официальные магазины приложений — далеко не единственный путь компрометации устройства. Даже новый смартфон может содержать вредоносный код. Так, в результате атаки на разработчика на смартфоны AlcateI было предустановлено шпионское ПО, и пользователи приобретали устройства, которые были заведомо скомпрометированы. Еще один пример — бэкдор TimpDoor; его хакеры распространяли отправляя жертвам ссылку по SMS.

Для противодействия атакам iOS запрещает загрузку приложений из любых источников, кроме App Store, но существуют способы обойти ограничение, например с помощью собственных сертификатов и протокола MDM (Mobile Device Management). Для этого пользователю необходимо вручную подтвердить доверие к сертификату разработчика приложения и его установку из недоверенного источника, на что пользователя может сподвигнуть злоумышленник в процессе фишинговой атаки.

В качестве меры борьбы с киберпреступниками компания Apple запрещает размещение в App Store приложений, в которых используются так называемые private API — методы, которые позволяют приложениям выполнять ряд действий на устройстве, в том числе загружать другие приложения. Троянская программа с private API сможет устанавливать на устройство жертвы любые другие программы в обход App Store, а значит — в обход любых проверок. Однако, как выяснилось, защита ведется неидеально, благодаря чему, например, распространялся вредонос YiSpecter. Схема заражения YiSpecter очень простая: пользователь открывает зараженную ссылку, подтверждает установку ПО не из App Store, и его мобильное устройство оказывается заражено. В свою очередь YiSpecter, попадая на устройство жертвы, использовал методы private API и автоматически загружал другие программы, которые похищали персональные данные жертвы.



Рисунок 26. Запрос на подтверждение установки стороннего ПО
(источник: zdnet.de/88248255/ios-malware-yispecter-auch-fuer-geraete-ohne-jailbreak-gefaehrlich/)

Рекомендация для пользователей

Не переходите по ссылкам от незнакомых людей в SMS и мессенджерах. Даже если человек, предлагающий установить таким образом привлекательное приложение, вам знаком, будьте бдительны. Никогда не подтверждайте запросы на установку сторонних программ на ваш смартфон

Один из альтернативных способов установки вредоносного ПО на устройства Apple — это загрузка файлов приложения (.ipa) на компьютер жертвы и их установка на устройство с помощью Apple ID (учетной записи в iCloud) и программы для установки приложений (например, Cydia Impactor) посредством соединения по USB-кабелю. Распространяться такое ВПО может в неофициальных магазинах под видом бесплатных («взломанных») версий ПО из App Store. Подобным образом устройства жертв заражались вредоносным ПО [WireLurker](#).



Рекомендация для пользователей

Не подключайте мобильное устройство к недоверенным ПК или зарядным станциям. Современные версии мобильных ОС запрашивают у пользователя подтверждение доверия. Никогда не подтверждайте доверие, если не уверены в безопасности компьютера, к которому подсоединен ваш гаджет

Политика Google в вопросе загрузки приложений из альтернативных источников менее строгая: возможность загружать программы из неофициальных источников оставлена на усмотрение пользователя и регулируется при настройке ОС. По [статистике](#), каждое пятое устройство на Android разрешает устанавливать приложения из сторонних источников. Кроме того, на 7% устройств Apple и 3% Android-устройств установлено как минимум одно приложение из неофициальных магазинов. Важно отметить, что административные права, о которых говорилось выше, снимают любые ограничения по загрузке программ как в iOS, так и в Android.



Рекомендация для пользователей

Не доверяйте сторонним магазинам мобильных приложений. Подозрительные программы (например, якобы «взломанные» бесплатные версии коммерческих приложений) могут содержать вредоносный код

Как мы видим, безопасность устройств зависит и от пользователей. Владельцы гаджетов должны ответственно подходить к вопросам защиты своих данных, которые они хранят в мобильных приложениях. Однако мер безопасности со стороны пользователей недостаточно, если приложение содержит уязвимости, которые были допущены разработчиками. К сожалению, не все производители программ для мобильных устройств обеспечивают надежную защиту своих систем.

Г Выводы

Мобильные устройства — привлекательная мишень для хакерских атак, ведь в них хранятся и обрабатываются большие объемы личной информации и данные банковских карт. Как показывают результаты нашего исследования, при разработке мобильных приложений вопросам безопасности уделяется недостаточно внимания, и основная проблема связана с небезопасным хранением данных. Хранение пользовательской информации в открытом виде, незащищенные данные на снимках состояния экрана, ключи и пароли в исходном коде — это лишь немногие из тех недостатков, что открывают просторы для проведения кибератак.

Пользователи могут сами способствовать компрометации своих устройств: расширять стандартные возможности смартфона, лишая его защиты, переходить по подозрительным ссылкам в SMS-сообщениях, загружать программы из неофициальных источников, поэтому безопасность пользовательских данных — ответственность не только разработчиков приложений, но и самих владельцев мобильных устройств.

Говоря о мобильных приложениях, нельзя недооценивать риск кибератаки в результате эксплуатации серверных уязвимостей. Серверы мобильных приложений защищены не лучше, чем клиентские части. В 2018 году каждая серверная часть содержала хотя бы одну уязвимость, которая позволяет проводить разнообразные атаки на пользователей, включая фишинговые рассылки от имени сотрудников компании-разработчика, что ставит под удар ее репутацию. Для предотвращения эксплуатации серверных уязвимостей мы рекомендуем использовать межсетевой экран уровня приложений (web application firewall).

Риски связаны не только с брешами в защите клиента или сервера, но и с уязвимостями, которые возникают в процессе клиент-серверного взаимодействия. Обмен данными по открытому протоколу грозит полной компрометацией передаваемого трафика. Но даже защищенные соединения не всегда надежны, что говорит об отсутствии у разработчиков глубокого понимания важности вопросов безопасности.

Механизмы защиты являются слабым звеном мобильных приложений. Большинство уязвимостей были заложены еще на этапе проектирования и стали следствием недостаточной проработки концепции защиты. Мы рекомендуем тщательно прорабатывать вопросы безопасности мобильного приложения и регулярно проводить тестирование его защищенности, начиная с самых ранних стадий жизненного цикла. Наиболее эффективным методом проверки является метод белого ящика, при котором предполагается наличие у эксперта доступа к исходному коду системы.

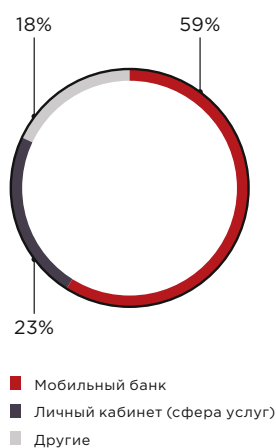
Об исследовании

Отчет содержит результаты исследования 17 полнофункциональных мобильных приложений, для которых в 2018 году проводился углубленный анализ с наиболее полным покрытием проверок. В выборке не представлены системы, владельцы которых не дали своего согласия на использование результатов анализа защищенности в исследовательских целях, и те системы, для которых был проведен анализ только части функциональных возможностей.

8 клиентов
Android

9 клиентов
iOS

7 серверных
частей



Оценка защищенности проводилась методами черного, серого и белого ящика с использованием вспомогательных автоматизированных средств. Метод черного ящика заключается в проведении работ по оценке защищенности информационной системы со стороны внешнего атакующего без предварительного получения какой-либо дополнительной информации о ней от владельца. Метод серого ящика аналогичен, но в качестве нарушителя рассматривается пользователь, имеющий определенные привилегии в системе. При анализе методом белого ящика для оценки защищенности информационной системы используются все имеющиеся данные о ней, включая исходный код приложений.

В настоящем документе приведены уязвимости клиентских и серверных частей. В дополнение рассмотрены угрозы мобильных приложений, в том числе и те, которые обусловлены клиент-серверным взаимодействием. Отчет содержит только описания уязвимостей, связанных с ошибками в коде и конфигурации мобильных приложений. Другие распространенные проблемы информационной безопасности (к примеру, недостатки процесса управления обновлениями ПО) не рассматриваются. Уязвимости кода мы разделили на две группы:

- уязвимости в коде мобильного приложения (ошибки, которые допустил программист при разработке);
- ошибки реализации механизмов защиты (появляются в системе еще на этапе проектирования).

Уровень риска уязвимостей оценивался исходя из степени влияния потенциальной атаки на пользовательские данные и само приложение, а также с учетом сложности проведения атаки; выделены качественные оценки высокого, среднего и низкого уровней риска.

О компании

ptsecurity.com

pt@ptsecurity.com

facebook.com/PositiveTechnologies

facebook.com/PHDays

Positive Technologies — один из лидеров европейского рынка систем анализа защищенности и соответствия стандартам, а также защиты веб-приложений. Организации во многих странах мира используют решения Positive Technologies для оценки уровня безопасности своих сетей и приложений, для выполнения требований регулирующих организаций и блокирования атак в режиме реального времени. Благодаря многолетним исследованиям специалисты Positive Technologies заслужили репутацию экспертов международного уровня в вопросах защиты SCADA- и ERP-систем, крупнейших банков и телеком-операторов.

Деятельность компании лицензирована Минобороны России, ФСБ России и ФСТЭК России, продукция сертифицирована Минобороны России и ФСТЭК России.