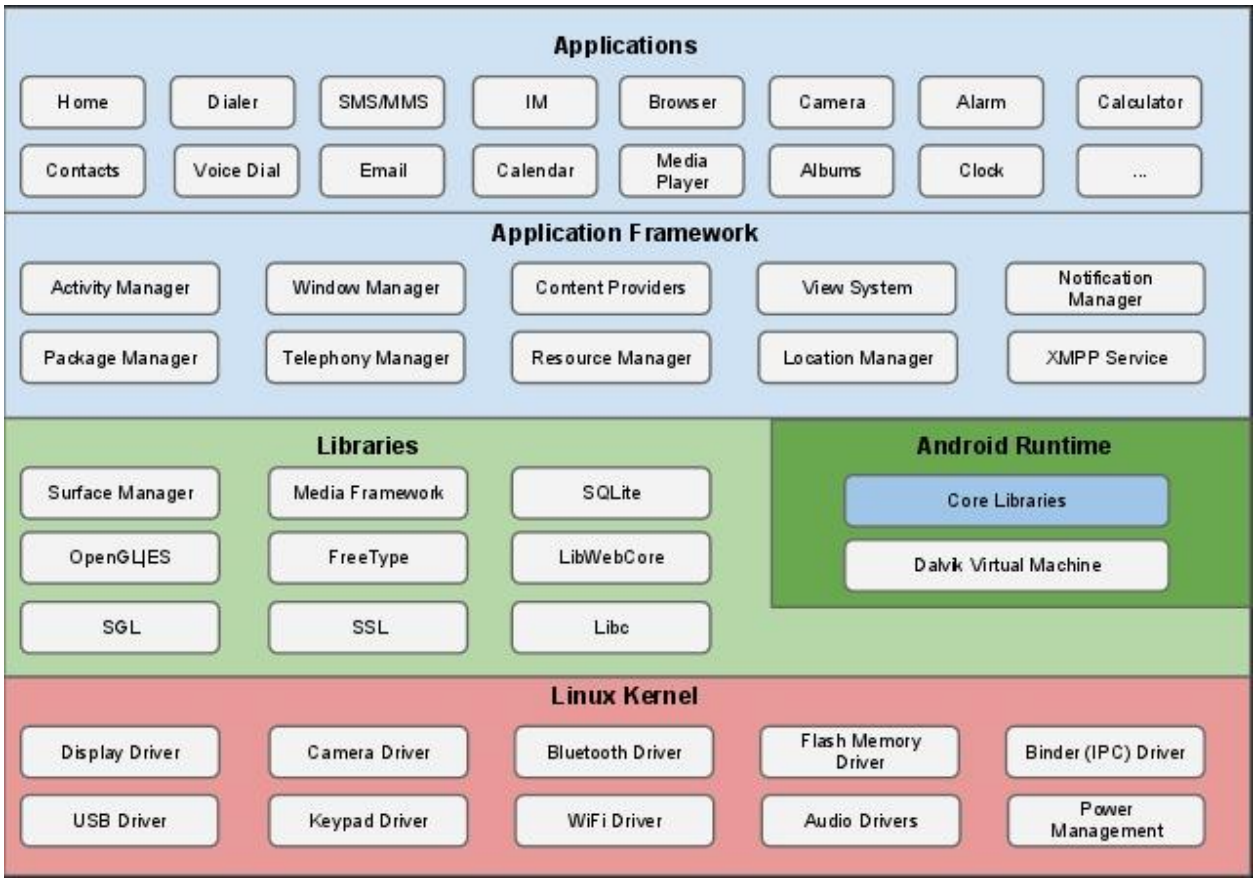
ЛЕКЦИЯ 2 – МЕХАНИЗМЫ ЗАЩИТЫ ОС СМАРТФОНА ANDROID

2.1. МЕХАНИЗМЫ ЗАЩИТЫ ANDROID

Система Android базируется на ядре Linux, но ее разработчики сильно модифицировали некоторые базовые механизмы, что в конечном итоге привело к усилению защиты. В частности, рабочая среда Android включает в себя драйверы оборудования, поддержку сетевого стека, файловую систему, а также механизмы управления памятью, процессорным временем и расходом электроэнергии. Все эти механизмы реализуются с помощью библиотек, написанных на языке Си/Си++, но все приложения для Android исполняются в виртуальной машине Dalvik VM, которая, по своей сути, является подмножеством Java 5 Standard Edition.



В отличии от Java, в Android используются свои библиотеки классов и более компактный метод сохранения исполняемых файлов (выполняемые программы для Android имеют расширение .dex). Приложения для Android формируются в специальные пакеты, которые имеют расширение .apk и очень похожи на jar-файлы Java.

2.1.1. Песочница

Каждое приложение Android - это отдельный пользователь со своими правами доступа и полномочиями, который имеет собственный идентификатор (UID) и запускается в собственной виртуальной машине. Для каждой такой машины действует принцип изоляции по потокам и низкоуровневому распределению памяти. Все взаимодействие отдельных процессов происходит только через ядро Linux, так что для обхода этой

системы безопасности придется не только получить права root на устройстве, но и каким-то образом скомпрометировать ядро, что является очень сложной задачей.

Каждое приложение получает собственный подкаталог внутри каталога /data в файловом хранилище Android, и все данные там защищаются с помощью прав доступа, которые разрешают самому приложению читать и писать собственные файлы, но запрещают делать это любому другому процессу. В Android это называется песочницей (sandboxing), которая позволяет сберечь данные соседних приложений друг от друга. В песочницу попадают все пользовательские приложения, включая заранее предустановленные на аппарат.

2.1.2. Привилегии (разрешения)

Наряду с песочницей, одним из основных механизмов системы безопасности Android являются права доступа приложений к функциям системы Android (привилегии), которые позволяют контролировать, какие именно возможности ОС будут доступны приложению. Это могут быть как функции работы с камерой и доступ к файлам на карте памяти, так и возможность использования функциональности, которая может привести к утечке информации со смартфона (доступ в Сеть), либо к трате средств пользователя со счета мобильного оператора (отправка SMS и совершение звонков).

Каждое Android-приложение обязано содержать в себе информацию о том, какие именно из функций Android оно может использовать. Эта информация заключена в файле AndroidManifest.xml внутри apk-файла и извлекается инсталлятором перед установкой приложения для того, чтобы пользователь смог ознакомиться с тем, к какой функциональности смартфона приложение сможет получить доступ. При этом пользователь должен в обязательном порядке согласиться с этим списком перед установкой приложения.

2.2. ОСНОВНЫЕ УГРОЗЫ НА ANDROID

2.2.1. Вирусы и вредоносное ПО

Список уверенно возглавляют СМС-троянцы (семейство Android.SmsSend). Целью таких программ является отправка сообщений с повышенной тарификацией на короткие номера. Часть стоимости этих сообщений поступает в карман злоумышленников, обогащая их. Чаще всего они распространяются под видом популярных игр и приложений, таких как Opera Mini, ICQ, Skype, Angry Birds и т. п., при этом используется соответствующая иконка.

Далее по списку следуют более «тяжеловесные» троянцы. К ним относятся, например, Android.Gongfu, Android.Wukong, Android.DreamExploid, Android.Geinimi, Android.Spy и пр. В зависимости от семейства, эти вредоносные программы обладают таким функционалом, как сбор конфиденциальной информации пользователя, добавление закладок в браузер, выполнение команд, поступающих от злоумышленников (функции бэкдора и бота), отправка СМС-сообщений, установка других приложений и т. п.

Существуют также коммерческие программы-шпионы. Эти приложения используются для слежки за пользователями. В их арсенал, в зависимости от класса, стоимости и производителя, входят такие функции, как перехват входящих и исходящих СМС-сообщений и звонков, аудиозапись окружения, отслеживание координат, сбор статистических данных из браузера (например, закладки, история посещений) и т. п.

2.2.2. Уязвимости архитектуры Android и ПО

Одна из главных проблем, с которыми могут столкнуться пользователи – это уязвимости системы, позволяющие получить права root. Существуют специальные приложения, скрипты и программные модули, выполняющие эту задачу. В повседневной жизни подобные вещи пользователям не страшны, так как чаще всего их используют осознанно для получения большего контроля над устройством. Другое дело, что эти же уязвимости (например, CVE-2009-1185 , CVE-2011-1823) взяли на вооружение создатели вредоносных приложений. Используя эксплойты для повышения своих прав до уровня root, они получают возможность, например, беспрепятственно устанавливать другие программы без разрешения пользователя (как это делают различные модификации Android.Gongfu и Android.DreamExploid). Некоторые вредоносные программы не используют эксплойты сами, а вводят пользователя в заблуждение и побуждают его самого выполнить необходимые действия, тем самым дав вредоносной программе требуемые ей возможности.

Существует возможность создания приложений, которые не будут требовать никаких разрешений для своей работы, что может создать ложное ощущение полной безопасности. Однако на самом деле такие приложения смогут получить доступ к определенной информации (например, файлам, хранящимся на карте памяти в незащищенном виде, списку установленных программ, используемому оператору мобильной связи) и даже отправить эту информацию злоумышленникам через Интернет.

2.2.3. Уязвимости прошивок и ПО

Угрозу представляют также неофициальные или сторонние прошивки. Поводов для беспокойства здесь несколько.

Во-первых, в такие прошивки изначально могут быть встроены вредоносные программы. Во-вторых, когда цифровой подписью образа системы подписывается какое-либо приложение, оно получает те же права, что и сама система, в которой оно работает. Подобный способ заражения применялся, в частности, вредоносной программой Android.SmsHider, которая могла незаметно для пользователей, использующих определенные сторонние прошивки, установить содержащийся в ней троянский apk.

Системные приложения, как стандартные, так и приложения от поставщиков Android-устройств, тоже подвержены уязвимостям. Например, некоторые уязвимости браузера WebKit позволяют потенциальным вредоносным программам выполнить произвольный JavaScript-код и получить доступ к защищенным данным браузера.

Если разработчики прикладного ПО не уделяют достаточное внимание безопасности при работе с данными пользователей, эти данные могут быть скомпрометированы. Атаке могут подвергаться хранящиеся в незащищенном виде регистрационные данные, пароли от банковских карт и прочая конфиденциальная информация.

2.2.4. Репозиторий приложений

Репозиторий приложений Google Play является слабым местом Android. Основная проблема в том, что приложение и его автор не подвергаются полной проверке. Чтобы решить эту проблему, не прибегая к ручной проверке приложений на безопасность, как сделано в Apple App Store, Google ввела сервис Bouncer, представляющий собой виртуальную машину, в которой автоматически запускается любое публикуемое в репозитории приложение. Bouncer выполняет многократный запуск программы, производит множество действий, симулирующих работу пользователя с приложением, и

анализирует состояние системы до и после запуска с целью выяснить, не было ли попыток доступа к конфиденциальной информации, отправки SMS на короткие платные номера и так далее. По словам Google, Bouncer позволил сократить количество вредоносов сразу после запуска сервиса на 40%.

2.3. УСОВЕРШЕНСТВОВАНИЕ ANDROID

В Google на постоянной основе работает команда безопасности Android (Android Security Team), задача которой заключается в том, чтобы следить за качеством кода операционной системы, выявлять и исправлять найденные в ходе разработки новой версии ОС ошибки, реагировать на отчеты об ошибках, присланные пользователями и секьюрити-компаниями. В целом эта команда работает в трех направлениях:

- анализ новых серьезных нововведений ОС на безопасность. Любое архитектурное изменение Android должно быть в обязательном порядке одобрено этими специалистами;
- тестирование разрабатываемого кода, в котором принимают участие также Google Information Security Engineering team и независимые консультанты. Идет постоянно на протяжении всего цикла подготовки нового релиза ОС;
- реагирование на обнаружение уязвимости в уже выпущенной ОС. Включает в себя постоянный мониторинг возможных источников информации о найденной уязвимости, а также поддержку стандартного баг-трекера.

Если уязвимость будет обнаружена, команда безопасности начинает следующий процесс:

- уведомляет компании, входящие в альянс OHA (Open Handset Alliance), и начинает обсуждение возможных вариантов решения проблемы;
- как только решение будет найдено, в код вносятся исправления;
- патч, содержащий решение проблемы, направляется членам OHA;
- патч вносится в репозиторий Android Open Source Project;
- производители/операторы начинают обновление своих устройств в режиме OTA или публикуют исправленную версию прошивки на своих сайтах.

Особенно важным в этой цепочке является тот факт, что обсуждение проблемы будет происходить только с теми членами OHA, которые подписали соглашение о неразглашении. Это дает гарантию, что общественность узнает о найденной проблеме только после того, как она уже будет решена компаниями, и фикс появится в репозитории AOSP. Если же об уязвимости станет известно из общедоступных источников (форума, например), команда безопасности сразу приступит к решению проблемы в репозитории AOSP, так чтобы доступ к исправлению получили сразу все и как можно скорее.

2.4. СРАВНЕНИЕ МЕХАНИЗМОВ БЕЗОПАСНОСТИ В ANDROID И IOS

С точки зрения разработчиков ПО основной риск безопасности - это взлом их приложения и, как следствие, потеря клиентов и бизнеса. Если рассматривать способность мобильных приложений противостоять локальным и веб-атакам, то обе операционные системы находятся примерно в равных условиях. Следуя практикам безопасной разработки, разработчикам вполне под силу создать хорошо защищенное приложение и для Android, и для iOS.

Как правило, приложения для Android представляют собой программы, написанные на языке Java, который невосприимчив к атакам на переполнение буфера, в отличие от программ для iOS, написанных на Objective-C. Приложения для Android можно легко декомпилировать и изменить исходный код на вредоносный, поэтому разработчики должны применять техники обфускации кода.

Приложения, написанные на Objective-C, потенциально уязвимы к переполнению буфера, однако в арсенале iOS-разработчиков присутствуют необходимые механизмы, способные предотвратить успешную эксплуатацию таких уязвимостей. К таким механизмам, прежде всего, относятся параметры компиляции, такие как PIE (Position Independent Executable), SSP (Stack Smashing Protection) и ARC (Automatic Reference Counting). Данные параметры обеспечивают эффективное управление памятью и отсутствие ошибок, которые приводят к переполнениям буфера. В дополнение к этому на презентации восьмой версии iOS был представлен новый язык программирования Swift, призванный заменить собой Objective-C. По заявлениям компании Apple, новый язык является более простым в изучении и более безопасным, чем его предшественник.

Таким образом, Android- и iOS-приложения могут быть одинаково хорошо защищенными, и вероятность взлома этих приложений напрямую зависит от мастерства разработчиков.

Безопасность рядовых пользователей мобильных девайсов зависит от безопасности мобильной ОС, которой они пользуются. Даже если на девайсе установлены только хорошо защищенные приложения, конечные пользователи все равно могут быть успешно атакованы через бреши в самой операционной системе. Если по критерию защищенности мобильных приложений обе ОС находятся на примерно одинаковом уровне, то с точки зрения безопасности самой системы Android и iOS значительно различаются между собой.

Прежде чем говорить о различиях в механизмах защиты двух ОС, стоит отметить наличие в них базовых принципов безопасности, таких как доступность системного раздела только для чтения и разграничение выполняемых процессов на уровне ядра. И в Android, и в iOS системный раздел не доступен для записи, что предотвращает случайное либо целенаправленное изменение файлов системы. Также в обоих ОС реализован принцип «песочницы» (sandbox). Это значит, что каждое приложение работает в изолированном контейнере и не имеет доступа к системным файлам либо ресурсам других приложений. В системе iOS почти все приложения выполняются под непривилегированным пользователем «mobile». В Android каждому приложению соответствует свой уникальный пользователь, что обеспечивает разграничение прав выполняемых приложений на уровне ядра операционной системы.

Основные различия в механизмах безопасности Android и iOS относятся к принципам разграничения доступа на уровне ядра, к процессу верификации загружаемого в магазины ПО и к принципам контроля прав доступа устанавливаемых приложений.

Перед тем, как появиться в магазине App Store, iOS-приложения тщательно проверяются на наличие уязвимостей и на соответствие стандартам разработки Apple. Также каждое приложение, устанавливаемое на iOS, должно быть подписано уникальным сертификатом программы «iOS Developer Program», который выдается компанией Apple только после необходимых верификаций разработчика. Описанные меры обеспечивают отсутствие вредоносного ПО в магазине приложений App Store.

Google не проверяет тщательно приложения перед загрузкой их в Google Play, но проводит регулярные сканирования своего магазина на предмет наличия потенциально вредоносного ПО. Такой подход компании Google может показаться достаточно небезопасным. И действительно, в магазине Google Play размещено большое количество потенциально вредоносного ПО (malware). Однако большинство таких программ на самом деле не несут в себе ничего вредоносного и на деле являются обычными рекламными приложениями.

Бывает, что в магазине Google Play присутствуют настоящие вредоносные приложения типа malware, но при должных пользовательских навыках можно полностью оградить себя от воздействия такого рода ПО. Дело в том, что при установке нового приложения на девайс под управлением Android пользователю показывается полный перечень прав доступа, требуемых данному приложению. По этому перечню пользователь может определить потенциально вредоносное ПО и отменить его установку. Например, если приложение «Фонарь» собирается запрашивать права на доступ к контактным данным либо на доступ к Интернету, то данное приложение с определенной долей вероятности можно отнести к вредоносному ПО.

А что насчет уязвимостей в самой операционной системе? Казалось бы, Android как полностью открытая ОС должна насчитывать огромное количество уязвимостей, найденных специалистами по всему миру. Однако iOS опережает Android по количеству известных уязвимостей (CVE), причем превосходство это довольно значительное. Несмотря на такое большое количество уязвимостей, пользователям iOS не стоит беспокоиться о своей защищенности, так как Apple, как правило, закрывает новые уязвимости достаточно быстро.

Таким образом, в контексте собственной безопасности операционной системы победителем не является ни одна из двух ОС. И Android, и iOS имеют мощные механизмы защиты от хакерских атак, и обе компании, Google и Apple, уделяют безопасности своих систем повышенное внимание.

В последние годы активно растет число пользователей, которые используют свои личные мобильные девайсы для выполнения рабочих задач. Эта тенденция, получившая название BYOD (Bring Your Own Device), несет в себе определенные риски безопасности для корпораций. При помощи уязвимого либо просто утерянного смартфона или планшета злоумышленники могут получить несанкционированный доступ к секретной документации компании либо к ее внутренним ресурсам, например к почте. В связи с этим возникает потребность в использовании решений типа Mobile Device Management (MDM), позволяющих централизованно управлять политиками безопасности мобильных девайсов, работающих в сетях компании.

С точки зрения безопасности на уровне корпорации ОС от Apple имеет ряд преимуществ перед Android. iOS имеет в своем арсенале мощные средства для централизованного управления девайсами, такие как профили конфигурации, возможность удаленного полного сброса и встроенная поддержка сторонних MDM-решений. Android в чистом виде таких возможностей не имеет. Для интеграции с MDM-системами на Android необходимо предварительно устанавливать специальное ПО.

Стоит отдельно отметить, что компания Samsung ушла далеко вперед в вопросах корпоративной безопасности по сравнению с другими производителями девайсов на Android. Речь идет о программе SAFE (Samsung For Enterprise) и надстройке KNOX, которая представляет собой хорошо защищенный контейнер для всех рабочих активностей пользователей с поддержкой сторонних MDM-систем. Таким образом, все аппараты от Samsung, работающие на Android 4.3 и выше, полностью соответствуют принципам защищенного бизнеса. Тем не менее, Apple имеет гораздо меньшую линейку продуктов, нежели производители Android-девайсов, поэтому ей не составляет труда обеспечить поддержку систем корпоративной безопасности для всех своих смартфонов,

планшетов и актуальных версий ОС. В категории наиболее безопасной для использования на уровне компаний операционной системы победителем выходит iOS.

Резюмируем основные плюсы и недостатки двух операционных систем с точки зрения безопасности:

Android:

Плюсы:

• Открытость для исследователей безопасности;

• Иммунитет приложений к переполнениям буфера;

• Строгий контроль доступа на уровне ядра.

Минусы:

• Большое количество потенциально вредоносного ПО в магазине Google Play;

• Слабые возможности в обеспечении корпоративной безопасности;

• Большое число версий ОС и моделей девайсов от различных вендоров, что усложняет стандартизацию методов защиты.

iOS:

Плюсы:

• Тщательный контроль загружаемых в App Store приложений, и, как следствие, практически полное отсутствие вредоносного ПО в магазине приложений;

• Быстрая реакция Apple на инциденты безопасности;

• Большие возможности по поддержке систем корпоративной безопасности.

Минусы:

• Большое количество известных уязвимостей в самой операционной системе;

• Рост числа возможных векторов для атаки (интеграция в экосистему Apple, система HomeKit).