An Empirical Study of Cryptographic Misuse in Android Applications

Александр Сухинин

4 июня 2015 г.

В данной статье рассказывается об исследовании безопасности мобильных приложений. Автором разработаны автоматические методы анализа безопасности приложений выложенных на play market. Данное исследование показало, что более 88% программ содержит по крайней мере одну ошибку. Также в данной статье предлагается ряд советов для повышения безопасности android приложений.

Одним из примеров плохой практики является использование режима ECB. Такой режим уязвим так, как в нем одинаковые фрагменты шифруются в одинаковую последовательность. Для обхода данной уязвимости следует к шифруемым блокам добавлять соль, чтобы сделать взлом более сложным.

Для проверки набора приложений используется статический анализ, позволяющий обнаружить общие недостатки. Для этих целей используется инструмент CryptoLint разработанный авторами. При помощи него исследовано 11 748 приложений среди которых 10 327 используют криптографию неправильно.

Далее приводится три алгоритма шифровани и их анализ. В качестве резюме авторами предлагается 6 правил, позволяющий значительно повысить безопасность системы.

1. Не использовать ECB режим при криптографии
2. Не использовать константные ключи шифрования
3. Не использовать константную соль для шифрования на основе пароля
4. Не использовать менее 1000 итераций для шифрования на основе пароля
5. Не использовать постоянные seed для получения псевдослучайных последовательностей SecureRandom()

Приложения для android отличаются от обычных java приложений. Более того, они выполняются на виртуальной машине Dalvik, которая отличается от java oracle. Такие приложения получают доступ к графическому интерфейсу и подсистемам. Итересующая нас подсистема - Java Cryptography Architecture (JCA). При помощи JCA регистрируются cryptographic service providers (CSP), предоставляющие реализацию большинства алгоритмов. Для получения доступа к этим алгоритмам необходимо вызвать метод Cipher.getInstance.

В таком вызове только название алгоритма является необходимой частью, остальные настройки могут быть приняты по умолчанию. К сожалению, очень часто по умолчанию выбирается режим ECB.

Далее в статье рассказывается об общей архитектуре инструмента и том, как именно из приложений извлекались графы потока управления и как в них возможно было обнаружить нарушение вышеуказанных правил.

Далее идет сравнение результатов, самыми частыми нарушениями являются нарушения первого и третьего правила. В качестве примера рассматривается три популярных приложения и обнаруженные в них уязвимости. Следует отметить, что данные приложения имеют миллионы скачиваний и содержат по несколько нарушений правил.