Эссе №6
Курс: Защита информации

Воробьев Олег

8 июня 2015 г.

В данной статье автор рассматривает вопросы безопасности данных в мобильных приложениях. Автор рассматривает систему Android как наиболее удачный объект исследований. Она привлекательна как очень популярная система для мобильных устройств следовательно, содержащая большой объем информации о данных пользователя. Так же дольшое количество приложений под эту систему делает оюъем данных, контактирующих с ней еще более внушительным.

Например режим ECB. Главная его уязвимость заключается в том, что одинаковые фрагменты данных с помошью этого алгоритма шифруются одинаковым шифром. Дананя система может быть усилена добавлением случауного числа "соли"в функцию, что создаст больше сложности с подбором.

Анализ безопасности проволится на основе собственноручно разработанного инструмента на базе androguard-framework. Основная идея состоит в поиске по исходному коду значений инициализирующих векторов, ключей и т.д. а так же криптоалгоритмов. При помощи этого приложения исследовано 11 748 приложений среди которых 10 327 используют криптографию неправильно, а это больше 85

Далее приводится три алгоритма шифрования с различными типами блочного шифрования. Анализируя алгоритмы автор предлагает нам в качестве резюме 6 правил для правильного использования защита информации в Android системах.

- Не использовать ECB режим при криптографии
- Не использовать константные ключи шифрования
- Не использовать константную соль для шифрования на основе пароля
- Не использовать менее 1000 итераций для шифрования на основе пароля
- 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, 1 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 и для сравнения результатов, самой популярной проблемой является нарушение первого и третьего правила. В качестве примера рассматривается три по- пулярных приложения и обнаруженные в них уязвимости. 2