УДК: 004.773

Д.А. Косов [1], Р.Ю. Шлаустас [1]

[1] Иркутский государственный университет путей сообщения, г. Иркутск, Российская Федерация

РЕАЛИЗАЦИЯ ПРОГРАММНЫХ СРЕДСТВ ДЛЯ ЗАЩИЩЕННОЙ КОММУНИКАЦИИ МЕЖДУ МОБИЛЬНЫМИ УСТРОЙСТВАМИ

Аннотация: *Описываются методики организации защищенного обмена сообщениями между мобильными клиентами. Рассматривается применение асимметричного шифрования в централизованной сети поверх Интернет. Перечисляются средства реализации соответствующего программного обеспечения.*

Ключевые слова: *Android, программное средство, асимметричное шифрование.*

D.A.Kosov [1], R.Yu.Shlaustas [1]

[1] Irkutsk State Transport University, Irkutsk, Russia

IMPLEMENTATION OF SOFTWARE TOOLS FOR SAFE COMMUNICATION BETWEEN MOBILE DEVICES

Abstract: *Describes the methods of the organization secure messaging between mobile clients. Regarded the use of asymmetric encryption centralized network over the Internet. Lists relevant means of implementation software.*

Keywords: *Android, software tool, asymmetric cryptography.*

Для организации защищенного обмена сообщениями между мобильными устройствами могут быть применены различные подходы и методики. В первую очередь можно выделить централизованную организацию такой сети устройств, с одним или несколькими специализированными узлами, и децентрализованную, когда все абоненты связываются друг с другом напрямую или друг через друга без специально выделенных узлов, играющих роль сервера.

Как уже было описано в статье [1], децентрализованная сеть с ячеистой топологией и ретрансляцией сообщений через абонентов может быть построена в соответствии с протоколом CJDNS или подобными. При этом возможно строить сеть на основе таких технологий как Wifi Direct и Bluetooth, без необходимости подключения к сети провайдера сотовой связи и Интернет.

Такие сети имеют существенный недостаток – при недостаточном количестве и низкой подвижности узлов, скорость и надежность передачи оперативной информации может быть не удовлетворительной. Поэтому реализация сети поверх глобальной сети Интернет может быть более предпочтительным вариантом, при хорошей доступности высокоскоростных сетей 3G\LTE и низкой стоимости использования канала провайдера услуг связи в регионе внедрения сети. Использование одного или нескольких серверов для организации обмена сообщениями также положительно скажется на быстродействии и снизит накладные расходы на маршрутизацию в децентрализованной одноранговой сети. В данной статье будет описана реализация именно такой, централизованной защищенной сети передачи сообщений и соответствующего программного обеспечения.

Архитектура такой системы может представлять собой три основные компоненты: программное обеспечение сервера ретрансляции зашифрованных сообщений, сервера обмена публичными ключами и мобильное приложение конечного абонента сети.

Для обеспечения конфиденциальности передаваемых сообщений применено асимметричное шифрование. В частности, алгоритмы на основе RSA [2]. При этом генерация ключевой пары производится на устройстве абонента, закрытый ключ хранится в зашифрованном виде в базе данных пользовательского приложения. Публичный ключ хранится там же и после генерации автоматически загружается на сервер обмена публичными ключами. Пользователь может настроить приложение для работы с любым совместимым сервером ключей и

сервером ретрансляции сообщений. Для передачи сообщения отправитель должен знать публичный ключ получателя, для этого он может запросить сервер публичных ключей на предмет наличия такого ключа и загрузить публичный ключ в базу данных своего устройства. Также публичный ключ получателя можно добавить или изменить вручную. Таким образом, нет принципиальной необходимости в существовании сервера ключей или его использовании тем или иным пользователем сети. Каждому абоненту системы соответствует некий уникальный идентификатор, в качестве такового используется адрес электронной почты. При попытке отправки на сервер публичных ключей нового ключа, ассоциированного с уже сохраненным идентификатором, сервер ключей ответит отказом. Проверка на существование соответствующего адреса электронной почты не производится. Информация об известных абоненту пользователях и их публичных ключах хранится в базе данных абонентского устройства, по аналогии с контактами телефона.

При отправке сообщения происходит его зашифровка на открытом ключе получателя и сохранение в базе данных отправителя. Получив сообщение, сервер ретрансляции перенаправляет его получателю, который сохраняет его в зашифрованном виде у себя в базе данных, на экране приложения оно отображается уже расшифрованным. Для расшифровки получатель использует свой закрытый ключ. Таким образом, сервер ретрансляции или злоумышленник, прослушивающий канал связи. не могут прочитать или незаметно модифицировать сообщение, не имея доступа к закрытому ключу получателя.

Общая схема системы и ее компонентов показана на рис. 1. На схеме отражены использованные программные модули и библиотеки

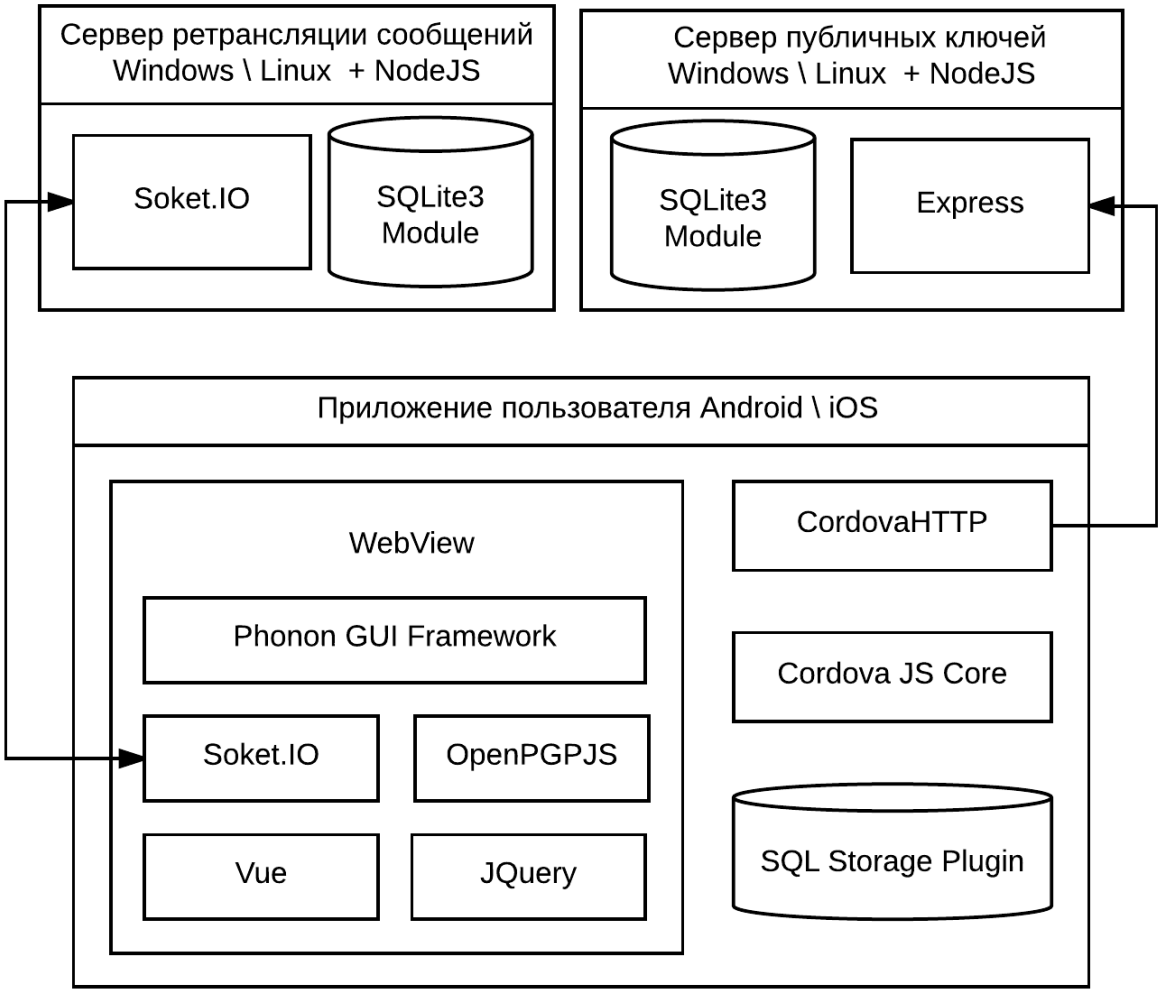

Рис. 1. Основные компоненты системы и их взаимодействие

В случае, если получатель не подключен к серверу ретрансляции в момент получения этим сервером запроса на ретрансляцию, сообщение сохраняется в базе данных сервера ретрансляции, где хранится ограниченное количество времени. При ближайшем по времени подключении получателя сообщение отправляется ему и удаляется из базы данных сервера ретрансляции. Отправитель получает от сервера ретрансляции соответствующие уведомления о приемке сообщения, его доставке и прочтении получателем. При прочтении сообщения получателем (отображении на экране в открытом виде заданный промежуток времени) его приложение автоматически отправляет на сервер ретрансляции соответствующее уведомление. Для идентификации сервер ретрансляции присваивает каждому сообщению уникальный номер.

Для реализации сервера ретрансляции сообщений выбрана платформа NodeJS [3]. Программный код написан на языке JavaScript (ECMA5). Для хранения данных используется СУБД SQLite3 [4] и соответствующий модуль NodeJS. Для организации обмена данными с абонентами использована библиотека Socket.IO [5]. Применение связки асинхронного NodeJS и кроссплатформенной библиотеки Socket.IO дает существенное преимущество в ресурсозатратах при высокой загруженности сервера запросами пользователей, по сравнению с решениями на базе PHP и подобных.

Применение сокет соединения позволило организовать дуплексную связь между сервером и клиентским приложением. Благодаря этому нет необходимости периодического опроса сервера клиентом, что было бы необходимо при использовании таких технологий как использование объекта XMLHttpRequest в встроенном браузере, или обычных HTTP POST запросах. Дуплексная связь снизила издержки на использование сетевого ресурса и упростила логику приложения.

Сервер публичных ключей основан на тех же технологиях, за исключением того, что сетевое взаимодействие организовано не с помощью библиотеки Socket.IO, а посредством программного модуля Express. Это связано с отсутствием необходимости в плотном обмене данными, т.к. запросы к серверу ключей осуществляются только при регистрации новых пользователей, либо при добавлении ими новых контактов в адресную книгу.

Для установки программного обеспечения сервером необходимо установить программные модули NodeJS, в том числе интерпретатор JavaScript (ECMA5), а также консольный менеджер пакетов NPM. Тестирование работоспособности разработанных серверов успешно произведено на удаленном VPS под ОС CentOS6 (RHEL).

Выбор средств реализации серверов обусловлен необходимостью обеспечить быстрый обмен небольшими по объему данными и относительно низкими требованиями к хранилищу данных. При этом сервера могут быть развернуты как на операционных системах семейства Windows, так и Linux\Unix.

Непосредственно пользовательское приложение с графическим интерфейсом и всей необходимой программной логикой разработано под мобильные платформы, в частности ОС Android версии 4 и выше для мобильных устройств с поддержкой связи в сетях EDGE\3G\4G. Также исходный код может быть скомпилирован под ОС iOS. Все данные приложения, в том числе БД, хранятся во внутренней памяти. Для повышения стойкости по отношению к возможным вредоносным действиям сторонних программ и закладок, скрытно установленных на мобильной ОС, рекомендуется установка специализированного антивирусного ПО и принятие других мер.

На сегодняшний день известно множество технологий и подходов для реализации мобильных приложений. В данной работе выбран подход, подразумевающий однократное написание программного кода на языке JavaScript (ECMA5), верстку графического интерфейса на HTML и CSS и последующую компиляцию в установочные пакеты для каждой мобильной операционной системы. Для этого используется средство разработки Apache Cordova [6]. включающее ряд консольных утилит и библиотек кода. Также использованы сторонние библиотеки (плагины в терминологии Cordova), для обеспечения доступа к возможно-

стям операционной системы целевой платформы, в том числе сетевого взаимодействия и хранения данных.

Для реализации графического интерфейса пользователя применены библиотеки JQuery, Vue [8] и Phonon [9]. Библиотека JQuery упрощает манипуляцию с DOM HTML страницы. предоставляя кроссбраузерное API. Библиотека Vue использована. главным образом, для прозрачного обновления динамического содержимого HTML страниц путем связывания его с структурами данных JavaScript. Фреймворк Phonon позволил визуализировать интерфейс приложения соответственно стилям целевой платформы (Android). При этом графический интерфейс разбивается на ряд страниц (Activity в терминологии Android), поочередно отображаемых пользователю, однако приложение остается по сути одностраничным (SPA), что упрощает его поддержку. Phonon предоставляет необходимые API для построения компонентов графического интерфейса соответственно платформе.

Заметим, что все необходимые HTML, CSS, JS файлы хранятся на самом устройстве, входят в установочный пакет приложения. Таким образом. подгрузка какого-либо исполняемого кода во время работы приложения не происходит, что положительно сказывается на безопасности и быстродействии.

Выбор средств реализации приложения для мобильного устройства обусловлен требованиями кроссплатформенности. Кроме того, использование одного языка программирования как на стороне сервера, так и клиента упростило проектирование и поддержку исходного кода.

Разработанное приложение было успешно протестировано на эмуляторе Genymotion в виртуальном устройстве под ОС Android 5.1, а также на физическом устройстве Alcatel 4034D под Android 6.0.

Для криптографических операций как на стороне сервера, так и в приложениях абонентов при разработке применена библиотека OpenPGPJS [7]. Она представляет собой реализацию популярной библиотеки OpenPGP на языке JavaScript (ECMA) и содержит все необходимые функциональные возможности для асимметричного (RSA) и симметричного (AES, DES, Blowfish и пр.) шифрования, работы с электронной цифровой подписью и хэширования (Ripemd, SHA).

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Д. А. Косов, Р. Ю. Шлаустас МЕТОДИКИ ПОСТРОЕНИЯ БЕСПРОВОДНЫХ ЗАЩИЩЕННЫХ СЕТЕЙ. //Информационные технологии и проблемы математического моделирования сложных систем.Вып.15. Иркутск: ИрГУПС, 2016. с.72-78.

2. Rivest R., Shamir A., Adleman L. A method for obtaining digital signatures and public-key cryptosystems // Commun. ACM New York City: ACM, 1978. Vol. 21, Iss. 2. P. 120–126. — ISSN 0001-0782; 1557-7317

3. [Электронный ресурс] URL: https://nodejs.org/en/ (дата обращения: 23.05.2017).

4. [Электронный ресурс] URL: https://www.sqlite.org/ (дата обращения: 23.05.2017).

5. [Электронный ресурс] URL: https://socket.io/ (дата обращения: 23.05.2017).

6. [Электронный ресурс] URL: https://cordova.apache.org/ (дата обращения: 23.05.2017).

7. [Электронный ресурс] URL: https://openpgpjs.org/ (дата обращения: 23.05.2017).

8. [Электронный ресурс] URL: https://ru.vuejs.org/ (дата обращения: 23.05.2017).

9. [Электронный ресурс] URL: http://phonon.quarkdev.com/ (дата обращения: 23.05.2017).

REFERENCES

1. D.A. Kosov, R.Yu. Shlaustas of the TECHNIQUE of CONSTRUCTION WIRELESS ZASHCHISHCHEN-NYH of NETWORKS.//Information technologies and problems of mathematical modeling of difficult systems. Issue 15. Irkutsk: ИрГУПС, 2016. page 72-78.

2. Rivest R., Shamir A., Adleman L. A method for obtaining digital signatures and public-key cryp-tosystems // Commun. ACM New York City: ACM, 1978. Vol. 21, Iss. 2. P. 120–126. — ISSN 0001-0782; 1557-7317
3. [Electronic resource] URL: https://nodejs.org/en/(date of the address: 23.05.2017).
4. [Electronic resource] URL: https://www.sqlite.org/(date of the address: 23.05.2017).
5. [Electronic resource] URL: https://socket.io/(date of the address: 23.05.2017).
6. [Electronic resource] URL: https://cordova.apache.org/(date of the address: 23.05.2017).
7. [Electronic resource] URL: https://openpgpjs.org/(date of the address: 23.05.2017).
8. [Electronic resource] URL: https://ru.vuejs.org/(date of the address: 23.05.2017).
9. [Electronic resource] URL: http://phonon.quarkdev.com/(date of the address: 23.05.2017).

Информация об авторах

Дмитрий Александрович Косов— магистр кафедры «Информационные системы и защита информации», Иркутский государственный университет путей сообщения, г. Иркутск, e-mail: kosov_idstu@mail.ru

Ромас Юргевич Шлаустас— к. ф.-м. н., доцент кафедры «Информационные системы и защита информации», Иркутский государственный университет путей сообщения, г. Иркутск, e-mail: shlaustas@gmail.com

Authors

Dmitryi Aleksandrovich Kosov— Bachelor, “Information Systems and Information Protection”, Irkutsk State Transport University, Irkutsk, e-mail: jukovtv@icloud.com

Romas Yurgevitch Shlaustas— Ph.D., in phisics and mathematics, Assistant Professor of “Information Systems and Information Protection”, Irkutsk State Transport University, Irkutsk, e-mail: shlaustas@gmail.com

Для цитирования

Шлаустас Р.Ю. Реализация программных средств для защищенной коммуникации между мобильными устройствами / Шлаустас Р.Ю., Косов Д.А. // «Информационные технологии и математическое моделирование в управлении сложными системами»: электрон. науч. журн. – 2018. – №1. – С. 84-88 – Режим доступа: http://ismm-irgups.ru/toma/11-2018, свободный. – Загл. с экрана. – Яз. рус., англ. (дата обращения: 01.10.2018)

For citation

Shlaustas R. Yu., Kosov D.A. Realizaciya programmnyh sredstv dlya zashchishchennoj kommunikacii mezhdu mobil'nymi ustrojstvami [Implementation of software tools for safe communication be-tween mobile devices] // Informacionnye tehnologii i matematicheskoe modelirovanie v upravlenii slozhnymi sistemami: ehlektronnyj nauchnyj zhurnal [Information technology and mathematical modeling in the man-agement of complex systems: electronic scientific journal], 2018. No. 1. P. 84-88. [Accessed 01/10/18]