ПЕРЕДАЧА КРИПТОГРАФИЧЕСКИ ЗАЩИЩЕННЫХ SMS-СООБЩЕНИЙ МЕЖДУ УСТРОЙСТВАМИ НА БАЗЕ ОС ANDROID

Ержан Э.А.[1], Атанов С.К.[2]

[1]Ержан Эзимхан Арманулы – магистрант;
[2]Атанов Сабыржан Кубейсинович – доктор технических наук, профессор,
факультет информационных технологий, кафедра вычислительной техники и программного обеспечения,
Евразийский национальный университет им. Л.Н. Гумилева,
г. Астана, Республика Казахстан

Аннотация: *в данной статье описан метод защиты SMS-сообщений и его преимущества, а также показана реализация метода под операционную систему Android, являющейся лидером среди современных операционных систем для мобильных устройств. Особенностью используемого метода является использование в нем симметричного шифра в совокупности со сжатием данных.*
Ключевые слова: *криптографическая защита, GSM, SMS, Android.*

SMS («сервис коротких сообщений») представляет собой технологию, позволяющую осуществлять прием и передачу коротких текстовых сообщений абонентами сотовой сети с помощью мобильных устройств. Согласно данным Международного союза электросвязи в 2010 году было отправлено 6.1 триллионов SMS-сообщений[1,2] по всему миру и это число продолжает расти с каждым годом.

Сегодня значительную часть сообщений получаемых абонентами сети составляют не личные сообщения, а сообщения от различных online сервисов. В таких сообщениях зачастую передаются конфиденциальные данные. К ним данным можно отнести коды авторизации, финансовую информацию, пароли и прочее. Получение злоумышленником такой информации может привести к взлому аккаунтов, выполнению несанкционированных действий в системе, краже средств и другим потерям.

Несмотря на меры безопасности, применяемые в GSM-сетях, данные передаваемые по ним могут быть уязвимы к перехвату [7, 8]. В связи с этим шифрование SMS-сообщений может быть использовано для обеспечения большей степени защищенности данных, передаваемых через мобильные сети[3]. Кроме того, существуют готовые продукты тактического назначения, предназначенные для перехвата GSM данных [6].

Описание метода защиты сообщений

Для шифрования сообщений в данном методе используется симметричный алгоритм блочного шифрования Advanced Encryption Standard (AES)[4], созданный в 1998 г. По результатам одноименного конкурса данный алгоритм был принят правительством США как новый стандарт шифрования. Особенностями данного алгоритма являются криптостойкость и простота реализации[5]. В отличие от предыдущих алгоритмов блочного шифрования AES не использует сети Фейстеля. Длина блока составляет 128 бит, а длина ключа составляет 128,192 или 256 бит в соответствии с которыми меняется и число раундов шифрования – 10, 12 и 14 соответственно.

Процесс защищенной передачи сообщений включает в себя следующие шаги (рис. 1):

1. Преобразование открытого сообщения в массив байт согласно используемой кодировке, стоит обратить внимание, что при расшифровке сообщений должна использоваться та же кодировка;
2. Инициализация шифра AES - представляет собой установку 128-битного ключа шифрования, сгенерированного из секретного слова, а так же его установку в режим шифрования;
3. Шифрование и сжатие сообщения;
4. Преобразование байтов шифр-текста в BASE64-строку;
5. Добавление в начало шифрованного сообщения служебных символов для их идентификации.

Рис. 1. Схема защищенной передачи сообщения

Расшифровка сообщений включает в себя следующие шаги (рисунок 2):

1. Удаление служебных символов;
2. Преобразование байтов BASE64-строки шифрованного сообщения в массив байтов;
3. Инициализация шифра AES - представляет собой установку 128-битного ключа шифрования, сгенерированного из секретного слова, а так же установку его в режим дешифрования;
4. Дешифрование и распаковка сообщения;
5. Преобразование байтов открытого текста в строку в соответствии с используемой кодировкой.

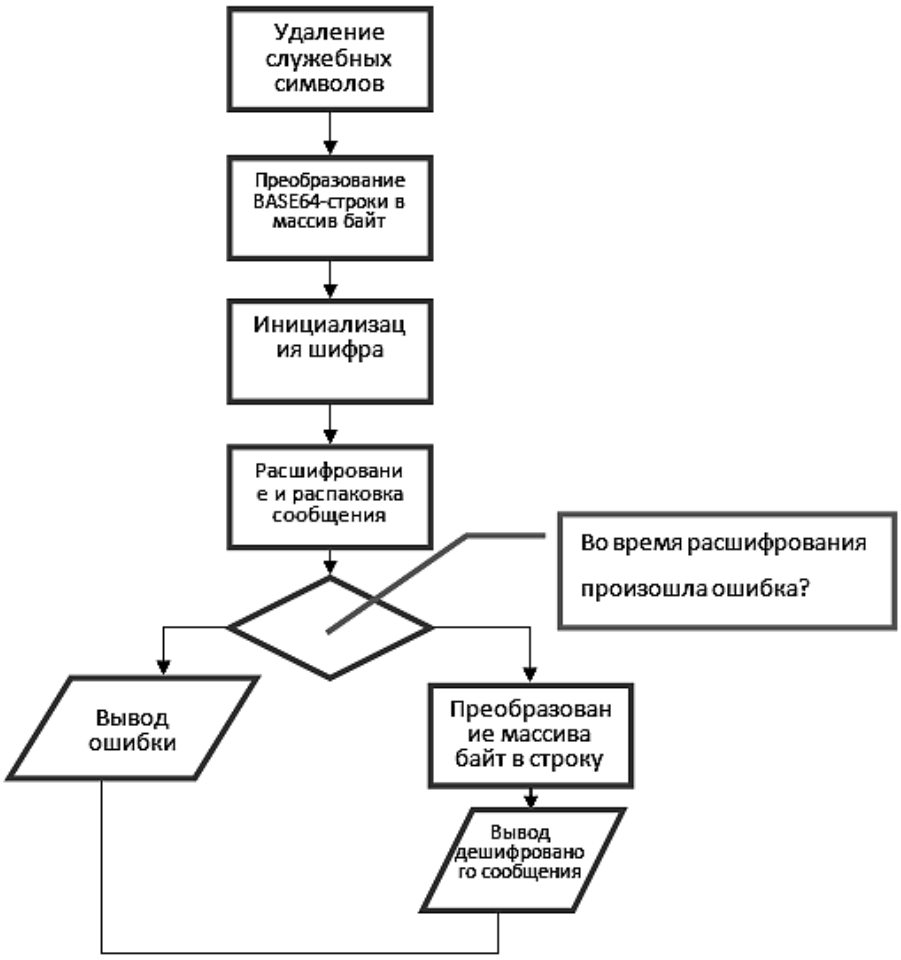

Рис. 2. Схема расшифровки зашифрованных сообщений

Реализация под ОС Android

Приложение для отправки защищенных сообщений, реализованное в рамках данной статьи, обладает следующими функциональными возможностями:

1. Просмотр списка зашифрованных сообщений.
2. Расшифровка зашифрованных сообщений.
3. Отправка зашифрованного сообщения.

Главное окно приложения (рис. 3), в нем представлен список зашифрованных SMS-сообщений, номер отправителя и дата отправки. Зашифрованные сообщения определяются по специальным служебным символам, которые добавляются в начало сообщения.

Рис. 3. Главное окно, список с зашифрованными сообщениями

По нажатию на сообщение в списке происходит переход в другое окно (рис. 4), в котором данное сообщение отображается пользователю в открытом, расшифрованном виде.

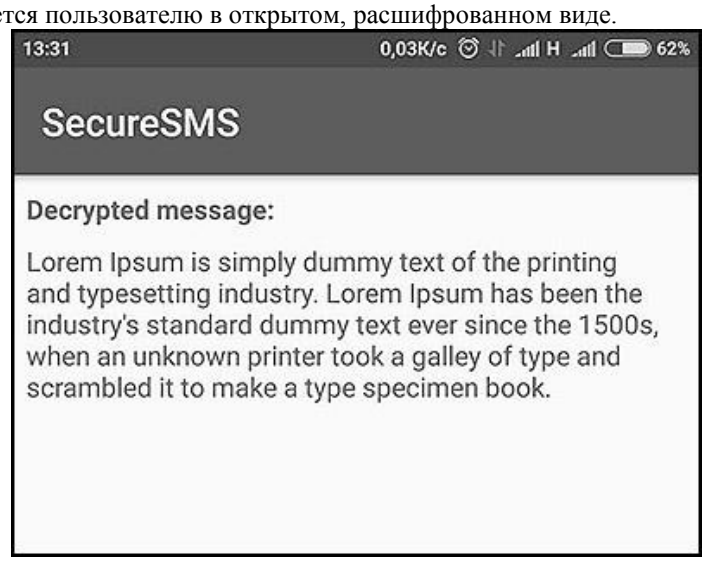

Рис. 4. Окно просмотра расшифрованного сообщения

По нажатию на кнопку с изображением конверта в правом нижнем углу происходит переход в окно отправки сообщения (рис. 5), в котором пользователь может отправить зашифрованное сообщение другому абоненту.

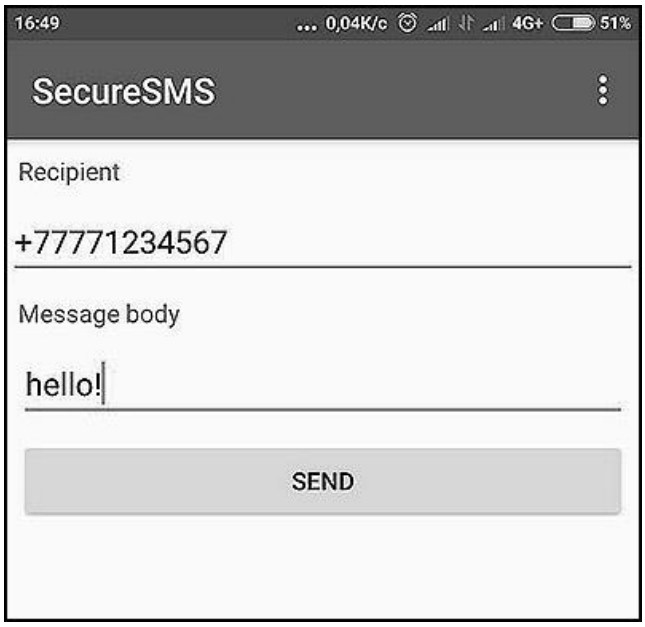

Рис. 5. Окно отправки зашифрованного сообщения

Оценка эффективности алгоритма

Для оценки эффективности алгоритма был проведен следующий эксперимент: со смартфона на базе ОС Android 6.0.1 с 8-ядерным процессором 1.9 GHz Cortex-A53 было отправлено 50 сообщений длинной в 100 символов, содержащих буквы английского и русского алфавитов. Результаты эксперимента представлены в таблице 1.

Таблица 1. Результаты эксперимента

	Степень сжатия	**Время шифрования (мс)**	**Время сжатия(мс)**	**Общее время обработки (мс)**
1	1.4186	0.135	0.694	0.829
2	1.3852	0.155	0.726	0.881
3	1.4609	0.15	0.464	0.614
4	1.4351	0.14	0.467	0.607
5	1.4762	0.2	0.4	0.6
6	1.4762	0.2	0.387	0.587
7	1.4015	0.205	0.352	0.557
8	1.472	0.225	0.422	0.647
9	1.3723	0.14	0.342	0.482
10	1.3759	0.17	0.512	0.682
11	1.2786	0.14	0.41	0.55
12	1.4015	0.14	0.826	0.966
13	1.3684	0.19	0.406	0.596
14	1.4275	0.14	0.349	0.489
15	1.6	0.17	0.362	0.532
16	1.3561	0.14	0.336	0.476
17	1.4015	0.155	0.403	0.558
18	1.4297	0.135	0.397	0.532
19	1.4341	0.145	0.912	1.057
20	1.3881	0.14	0.374	0.514
21	1.2786	0.14	0.4	0.54
22	1.4297	0.21	0.397	0.607
23	1.3704	0.73	0.422	1.152
24	1.2806	0.14	0.426	0.566
25	1.3066	0.21	0.326	0.536
26	1.3835	0.155	0.365	0.52
27	1.363	0.135	0.358	0.493
28	1.4427	0.14	0.502	0.642
29	1.4609	0.16	0.362	0.522
30	1.3456	0.14	0.435	0.575
31	1.3212	0.045	0.438	0.483

32	1.2606	0.04	0.301	0.341
33	1.4242	0.05	0.346	0.396
34	1.3969	0.04	0.41	0.45
35	1.3939	0.04	0.448	0.488
36	1.374	0.04	0.32	0.36
37	1.3806	0.04	0.346	0.386
38	1.6121	0.045	0.39	0.435
39	1.4091	0.045	0.333	0.378
40	1.2394	0.04	0.32	0.36
41	1.4091	0.045	0.342	0.387
42	1.5214	0.04	0.342	0.382
43	1.3481	0.105	1.101	1.206
44	1.4308	0.04	0.32	0.36
45	1.4683	0.045	0.669	0.714
46	1.4198	0.055	0.403	0.458
47	1.4688	0.06	3.283	3.343
48	1.3939	0.04	0.374	0.414
49	1.4496	0.045	0.39	0.435
50	1.4385	0.05	0.253	0.303
	1.404	**0.127**	**0.493**	**0.62**

Одной из особенностей реализованного алгоритма помимо шифрования является сжатие сообщений. В стандарте GSM сжатие SMS-сообщений не предусмотрено. В качестве алгоритма сжатия в данной реализации был выбран алгоритм DEFLATE, который оказался наиболее эффективным для сжатия строк небольшого размера.

Как видно из таблицы 1, средняя степень сжатия для сообщений длиной в 100 символов в среднем составляет 1.4 раза, что, в конечном счете, может положительно отразиться на стоимости отправки сообщения. Средняя скорость шифрования вместе со сжатием составила 0.62 миллисекунды, что является более чем приемлемым результатом. Такая задержка при обработки сообщений для конечного пользователя является незаметной. Следует отметить, что обратный процесс преобразования зашифрованного сжатого сообщения в исходное занимает практически такое же время.

Заключение

В данной статье описан метод защиты SMS-сообщений, а также приведено описание разработанного приложения под операционную систему Android.

В частности, были изложены алгоритмы передачи данных и проведены эксперименты. Для шифрования сообщений был использован симметричный алгоритм блочного шифрования AES (Advanced Encryption Standard), а для их сжатия был применен алгоритм DEFLATE. Эксперименты показали, что средняя степень сжатия сообщений длиной в 100 символов составляет 1.4 раза, что, в конечном счете, может положительно отразиться на стоимости отправки сообщения. Средняя скорость шифрования вместе со сжатием составила 0.62 миллисекунды.

На основе экспериментального тестирования можно сделать вывод об эффективности используемого подхода, что проявляется в скорости алгоритма, скорости шифрования, а также малом размере шифр-текста приемлемом для практического использования.

Список литературы

1. *Lai T.L.* Service quality and perceived value's impact on satisfaction, intention and usage of short message service (SMS) // Information Systems Frontiers, 2004. С. 353-368.
2. The World in 2010. ICT Facts and figures // International Telecommunication Union [Электронный ресурс]: Режим доступа: http://www.itu.int/ITU-D/ict/material/FactsFigures2010.pdf/ (дата обращения: 17.03.2017).
3. *Pesonen Lauri*. «GSM interception» // Department of Computer Science and Engineering.
4. Helsinki University of Technology [Электронный ресурс]: Режим доступа: http://www.tml.tkk.fi/Opinnot/Tik-110.501/1999/papers/gsminterception/netsec.html/ (дата обращения: 18.03.2017).
5. *Hans Dobbertin, Vincent Rijmen.* Advanced Encryption Standard – AES // 4th International Conference, 2004, с. 188.
6. *Daemen Joan and Vincent Rijmen.* The design of Rijndael: AES - the advanced encryption standard. // Springer Science & Business Media, 2013. с. 238.
7. *Christoph Kemetmüller.* Manipulating Mobile Devices with a Private GSM Base Station-A Case Study // INC, 2010. С. 10-12.

8. *Hulton David*. Intercepting GSM traffic // BlackHat Briefings, 2008. C. 2-6.