БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Конахови Г.Ф., Пузыренко Ю.А.* Компьютерная стеганография: теория и практика. – Киев: МК-Пресс, 2006. – 283 с.
2. *Грибунин В.Г., Оков И.Н., Туринцев И.В.* Цифровая стеганография. – М.: Солон-Пресс, 2009. – 260 с.
3. *Самойленко Д.В., Финько О.А.* Имитоустойчивая передача данных в защищенных системах однонаправленной связи на основе полиномиальных классов // Нелинейный мир. – 2013. – Т.11, № 9. – С. 642-658.
4. *Бояринов И.М.* Помехоустойчивое кодирование числовой информации. – М.: Наука, 1983. – 196 с.
5. *Бухштаб А.А.* Теория чисел. – М.: Просвещение, 1966. – 384 с.
6. *Финько О.А.* Модулярная арифметика параллельных логических вычислений: Монография / Под ред. В.Д. Малюгина. – М.: ИПУ РАН, 2003. – 224 с.
7. *Акушский И.Я., Юдицкий Д.И.* Модулярная арифметика в остаточных классах. – М.: Сов. Радио, 1968. – 440 с.
8. *Гмурман В.Е.* Теория вероятности и математическая статистика. – М.: Высшая школа, 2003. – 480 с.

Статью рекомендовал к опубликованию д.т.н. В.Н. Марков.

**Рябинин Юрий Евгеньевич –** Филиал Военной академии связи (г. Краснодар); e-mail: jurandvau@inbox.ru; 350063, г. Краснодар, ул. Красина, 4; тел.: +79186218528; сотрудник.

**Финько Олег Анатольевич –** e-mail: ofinko@yandex.ru; тел.: +79615874848; кафедра криптографических средств защиты информации и математических основ криптологии; д.т.н.; профессор.

**Ryabinin Jurii Evgenevich –** Branch of the Military Academy of Communications (Krasnodar); e-mail: jurandvau@inbox.ru; 4, Krasina, Krasnodar, 350063, Russia; phone: +79186218528; employee.

**Finko Oleg Anatolievich –** e-mail: ofinko@yandex.ru; phone: +79615874848; the department of cryptographic protection of information and mathematical foundations of cryptology; dr. of eng. sc.; professor.

УДК 004.056 .55

**Л.К. Бабенко, Д.А. Беспалов, О.Б. Макаревич, Р.Д. Чесноков, Я.А. Трубников**

## РАЗРАБОТКА И ИССЛЕДОВАНИЕ ПРОГРАММНО-АППАРАТНОГО КОМПЛЕКСА ШИФРОВАНИЯ ПО АЛГОРИТМУ PRESENT ДЛЯ РЕШЕНИЯ ЗАДАЧ МАЛОРЕСУРСНОЙ КРИПТОГРАФИИ

*Приведены результаты исследования одного из современных методов малоресурсной криптографии - блочного шифра PRESENT, а также некоторые практические результаты для программного решения и аппаратного модуля, выполненного в качестве отдельного устройства на базе программируемых логических интегральных схем (ПЛИС). Как показал анализ полученного решения, максимально допустимая частота работы схемы алгоритма для такой системы на кристалле определяется максимально допустимой частой тактирования ПЛИС и составляет 160 МГц, количество задействованных логических элементов составляет 297, количество задействованных блоков памяти – 0. Аппаратное решение алгоритма было выполнено для ПЛИС фирмы ALTERA Cyclone II EP2C20F484C7 с рабочими частотами 27, 50 и 100 МГц. Следует также упомянуть, что программное решение также адаптировано для технологии .NET Micro Framework и может применяться в 32- и 64-разрядных микроконтроллерах с архитектурой ARM7, ARM9 и Blackfin. Таким образом, получен ряд практически значимых результатов: проведено исследование алгоритма PRESENT, рассчитана трудоемкость, получено программное решение, достаточно эффективное для применения во встраиваемых устройствах, а также синтезирован аппаратный блок для системы на кристалле, удовлетворяющий всем требованиям малоресурсной криптографии, выполнена его*

*отладка, моделирование и проведены необходимые эксперименты, доказавшие его работоспособность, эффективность и возможность применения на практике. Описаны тонкости аппаратной реализации конфигурационного проекта, S-блока, а также приведены числовые характеристики разработанного вычислительного ядра.*

*Шифр; present; малоресурсная криптография; интернет вещей; ПЛИС.*

**L.K. Babenko, D.A. Bespalov, O.B. Makarevich, R.D. Chesnokov, Y.A. Trubnikov**

## SOFTWARE AND HARDWARE DEVELOPMENT AND RESEARCH OF ENCRYPTION ALGORITHM PRESENT FOR SOLVING PROBLEMS OF THE LIGHTWEIGHT CRYPTOGRAPHY

*This article presents the results of the block cipher PRESENT overview, as well as some practical results for software solutions and hardware modules, configured as a single FPGA device. The analysis of developed solution showed that maximum frequency of this algorithm implementation depends on maximum frequency of FPGA IC, uses 297 logical elements and 0 memory blocks. Hardware description of algorithm was implemented in Altera Cyclone II EP2C20F484C7 with 27, 50 and 100 MHz duty frequencies. In addition to hardware description of PRESENT, software implementation was developed and adapted for .NET Micro Framework technology. This software implementation can be used in 32- and 64-bit microprocessors and microcontrollers such as ARM7, ARM9 and Blackfin. As the result of this work, we have developed hardware and software definition of the PRESENT cipher and got synthesizable implementations, which meet requirements of lightweight cryptography and can be effectively used in different software and hardware applications such as RFID systems. Moreover, describes the hardware implementation of the subtleties of the project configuration, S-block, and also provides the numerical characteristics of the developed coprocessor core.*

*Cipher; present; lightweight cryptography; internet of things; system on chip (FAPGA).*

Большинство современных алгоритмов защиты информации и, в частности, шифрования, рассчитаны на применение в ЭВМ в составе программных комплексов без учета оптимизации на уровне аппаратного обеспечения. Этот факт делает невозможным применение большинства существующих криптографических алгоритмов в устройствах с ограниченной вычислительной мощностью, малым объемом и малым энергопотреблением. Данные системы также имеют название «системы с низкой стоимостью», а методы криптографической защиты данных в них - «методы малоресурсной криптографии».

Особо здесь стоит упомянуть так называемый «интернет вещей», который представляет собой беспроводную самоконфигурирующуюся сеть между объектами различного класса, примерами которых могут являться бытовые приборы, транспортные средства, интеллектуальные датчики и метки радиочастотной идентификации (RFID). RFID системы как раз предъявляют наивысшие требования, так как в большинстве своем являются системами с малой доступной площадью и питанием от электромагнитного поля.

Стандартными подходами к решению проблемы создания эффективных методов и средств малоресурсной криптографии являются:

1. Использование классических криптографических алгоритмов, если это возможно.
2. Модификация классических алгоритмов с адаптацией к аппаратным особенностям и ограничениям систем с низкой стоимостью.
3. Разработка новых специализированных решений в методологическом, алгоритмическом и программно-аппаратном плане.

Каждый из этих подходов имеет свои недостатки. До сих пор большинство решений в этой области знания относится к третьему подходу и показывают неплохие результаты. При этом, однако, следует помнить, что при адаптации криптографиче-

ского алгоритма к особенностям аппаратного базиса в условиях ограниченности ресурсов, могут возникать нежелательные последствия. Они могут выражаться в появлении дополнительных слабостей алгоритма или в ослаблении их общей стойкости.

В малоресурсной криптографии в основном используются как блочные, так и поточные алгоритмы. Классическими алгоритмами, ориентированными на аппаратную реализацию, являются: алгоритм потокового шифрования MICKEY, симметричный алгоритм синхронного потокового шифрования Trivium, алгоритм потокового шифрования GRAIN, симметричные алгоритмы блочного шифрования DESL и PRESENT. Также существуют совсем молодые алгоритмы KATAN и KTANTAN, а также MIBS и TWIS HIGHT или mCrypton, которые еще не достаточно хорошо исследованы.

Однако и они в силу индивидуальных особенностей не могут применяться повсеместно. Например алгоритм Trivium требует для своей реализации на кристалле площадь, в полтора раза превышающую все допустимые пределы, алгоритм GRAIN в малоресурсной версии успешно подвергается атаке на связанных ключах, а алгоритм MICKEY достаточно устойчив не ко всем видам атак, и некоторые специалисты не до конца уверенны в его надежности.

Среди блочных алгоритмов ситуация несколько лучше. Шифр DESL, разработанный на основе всем известного алгоритма DES, является удачным решением в малоресурсной криптографии благодаря тому, что последний уже разрабатывался с учетом аппаратной реализации. Авторами DESL доказано, что изменения, внесенные в алгоритм при его адаптации, не влияют на его стойкость к атакам в рамках дифференциального и линейного криптоанализа. Единственным серьезным недостатком является ключ длиной всего в 56 бит – этот ключ раскрывается на мощной многопроцессорной системе полным перебором в течении нескольких суток.

Альтернативой данного алгоритма является шифр PRESENT. Он разработан группой исследователей из Германии, Дании и Франции: A. Bogdanov, L.R. Knudsen, G. Leander, C. Paar, A. Poschmann, M.J.B. Robshaw, Y. Seurin, C. Vikkelsoe [1], представлен на конференции CHES-2007 и включен в стандарт ISO/IEC 29192-2. Авторы алгоритмы подчеркивают, что разработали его для специального применения в областях, где не подходят более универсальные алгоритмы типа AES. Его аппаратная реализация является самой компактной из всех существующих [3]. Кроме того, модификации этого алгоритма нашли свое применение и в других ресурсозависимых устройствах: например H-PRESENT-128 является самой компактной из известных хэш-функций, а другие его модификации используются в качестве генератора псевдослучайных чисел для схемы crypto-GPS.

PRESENT является блочным шифром и классической SP-сетью (Substitution permutation network) с 64-битными информационными блоками, 80-битным или 128-битным ключом и состоит из 31+1 цикла (раунда) шифрования. Укрупненное представление алгоритма показано на рис. 1.

Каждый раунд выполняется операция XOR с раундовым ключом $K_i$. Ключ имеет разрядность 64 бита и определяется функцией обновления ключа. После этого проводится так называемое рассеивающее преобразование, то есть блок пропускается через 16 одинаковых S-блоков, имеющих разрядность 4, составленных таким образом, чтобы максимально повысить устойчивость алгоритма к линейному и дифференциальному криптоанализу. Затем в блоке перестанавливаются (перемешиваются) биты.

Трудоемкость алгоритма не велика [2]. Для функции add_round_key() требуется выполнение операции $b_j \rightarrow b_j \oplus k_j^i$, где $j$ варьируется от 0 до 63, а $i$ – от 1 до 32. s_box_layer выполняет преобразование из 4-х бит в 4 бита. p_layer выполняет перемещение бита $i$ на позицию $P(i)$. Ключ задается пользователем в особом регистре ключа $K$. $K_i = K_{63}K_{62}K_{62} \ldots K_1K_0$, то есть содержит 64 бита левой

части значащих бит $K$. Таким образом в раунде $i$, $K_i = K_{63}K_{62}K_{62} \dots K_1K_0 = K_{79}K_{78}K_{77} \dots K_{17}K_{16}$. Регистр ключа обновляется по ходу алгоритма следующим образом: $K_{79}K_{78} \dots K_1K_0 = K_{18}K_{17} \dots K_{20}K_{19}$; $K_{79}K_{78}K_{77}K_{76} = S[K_{79}K_{78}K_{77}K_{76}]$; $K_{19}K_{18}K_{17}K_{16}K_{15} = K_{19}K_{18}K_{17}K_{16}K_{15} \oplus round\_counter$. Ключ сдвигается на 61 позицию влево, 4 бита левой значащей части проходит через S-блок, а над младшими правыми значениями циклического счетчика $i$ и битами $K_{19}K_{18}K_{17}K_{16}K_{15}$ регистра ключа $K$ выполняется операция XOR.

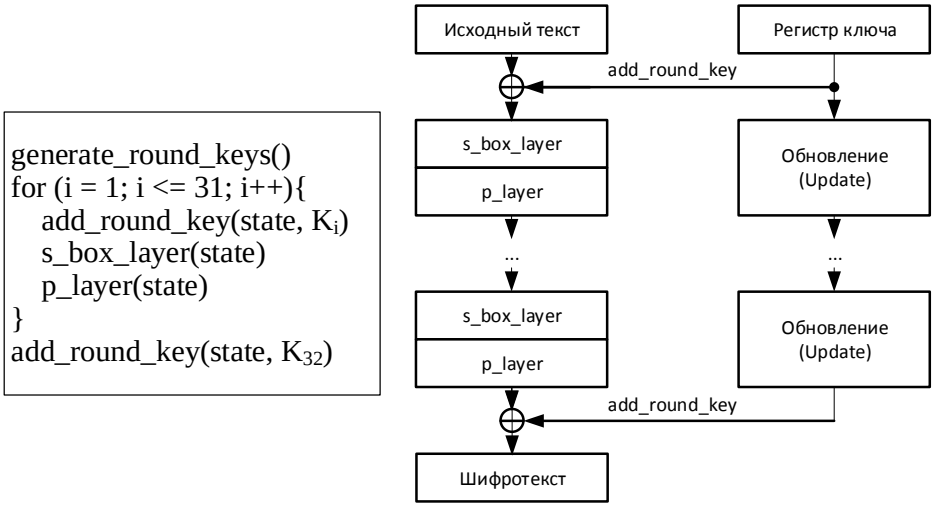


*Рис. 1. Укрупненное представление алгоритма PRESENT*

Данный алгоритм достаточно защищен от атак на основе связанных ключей, слайд-атак и других распространенных методов атак на криптосистемы.

Для оценки эффективности алгоритма, был разработан ряд практических решений, включающий в себя программные коды и аппаратную реализацию на базе системы на кристалле в виде программируемой логической интегральной схемы (ПЛИС).

Программное решение было выполнено на языке C# и может быть представлено в виде граф схемы (рис. 2).

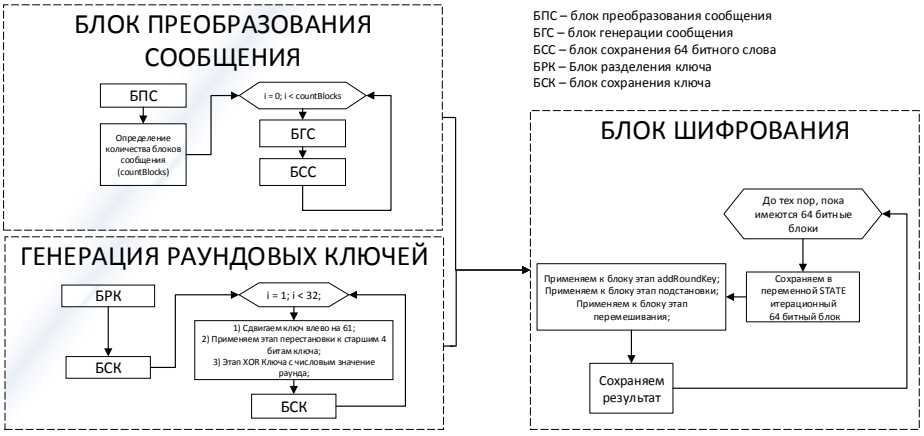


*Рис. 2. Граф-схема алгоритма PRESENT для программного решения*

Следует упомянуть, что программное решение также адаптировано для технологии .NET Micro Framework и может применения в 32- и 64-разрядных микроконтроллерах с архитектурой ARM7, ARM9 и Blackfin.

Аппаратное решение алгоритма выполнено для ПЛИС фирмы ALTERA Cyclone II EP2C20F484C7 с рабочими частотами 27, 50 и 100 МГц в виде отдельного вычислительного ядра. Оно, по сути, является системой на кристалле, удовлетворяющей классическим требованиям малоресурсной криптографии.

Структура конфигурационного проекта для аппаратной реализации алгоритма PRESENT показана на рис. 3.

Поскольку Present является малоресурсным шифром, к его аппаратной реализации предъявляются особые требования: площадь, занимаемая аппаратным блоком на кристалле по возможности не должна превышать 3000 GE, где GE (gate equivalent) – площадь, занимаемая на кристалле одним элементом типа 2И-НЕ.

Исходя из данного требования, необходимо отметить, что хранение таблиц замены для S-блоков в отдельных регистрах не является оптимальным решением с точки зрения аппаратных затрат ввиду того, что регистры строятся на основе D- или T-триггеров, занимающих от 5GE до 12GE.

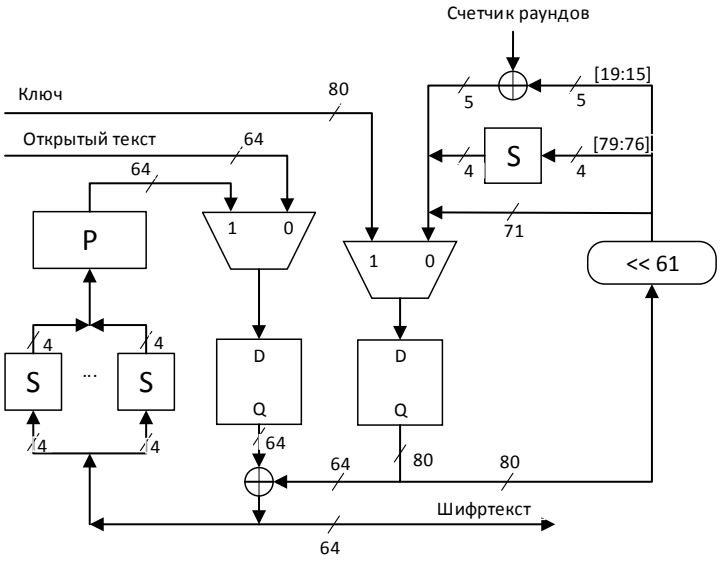


*Рис. 3. Структура конфигурационного проекта для системы на кристалле*

Однако, существует возможность существенного уменьшения количества используемых логических элементов за счёт реализации блоков замены как логических функций от нескольких аргументов.

S-блоки шифра Present имеют идентичную структуру, поэтому системы уравнений для таких блоков будут иметь одинаковый вид:

$$S_0(x) = x_3x_2\overline{x_1}x_0 + \overline{x_3}x_2\overline{x_1x_0} + \overline{x_3x_2x_1}x_0 + x_3x_1\overline{x_0} + \overline{x_3}x_2x_1x_0 + \\ + \overline{x_3x_2}x_1x_0 + x_3\overline{x_2x_0};$$

$$S_1(x) = \overline{x_3}x_2x_1\overline{x_0} + x_3x_2x_0 + \overline{x_2}x_1\overline{x_0} + x_3\overline{x_2x_1}x_0 + \overline{x_3x_2}x_1x_0 + x_3\overline{x_2x_0};$$

$$S_2(x) = \overline{x_3x_2x_1x_0} + \overline{x_3x_2x_1}x_0 + x_3x_2\overline{x_1} + \overline{x_3}x_2x_1x_0 + \overline{x_2}x_1\overline{x_0} + x_3\overline{x_2x_1}x_0;$$

$$S_3(x) = \overline{x_3}x_2x_1\overline{x_0} + \overline{x_3x_2x_1x_0} + \overline{x_3}x_2\overline{x_1x_0} + x_3\overline{x_2}x_1 + \overline{x_3}x_2x_1x_0 \\ + +x_3\overline{x_2x_1}x_0; + \overline{x_3x_2}x_1x_0,$$

где $x_3$, $x_2$, $x_1$, $x_0$ – биты входного вектора, $S_3$, $S_2$, $S_1$, $S_0$ – соответствующие биты выходного вектора.

Процесс синтеза таких уравнений с их последующей минимизацией достаточно трудоёмок, и сложность лишь возрастает с увеличением разрядности S-блока. Кроме того, ручное определение блоков замены является нерациональным с точки зрения скорости разработки и внедрения в маршрут проектирования. Поэтому современные системы автоматизированного проектирования, как правило, могут производить синтез подобных логических схем по описанию, представленному на языках описания аппаратуры, таких как Verilog и VHDL. В частности, для языка Verilog можно воспользоваться директивой function для синтеза комбинационных схем.

Сравнение вышеуказанных двух подходов к синтезу блоков замены в среде Altera Quartus II для ПЛИС Altera Cyclone II не выявило существенных преимуществ ручной разработки перед полностью автоматической. В обоих случаях S-блоки имеют структуру, представленную на рис. 4.

Как показал анализ полученного решения, максимально доступная частота работы схемы алгоритма для такой системы на кристалле определяется максимально допустимой частотой ПЛИС и составляет 160 МГц, количество задействованных логических элементов составляет 297, количество задействованных блоков памяти – 0.

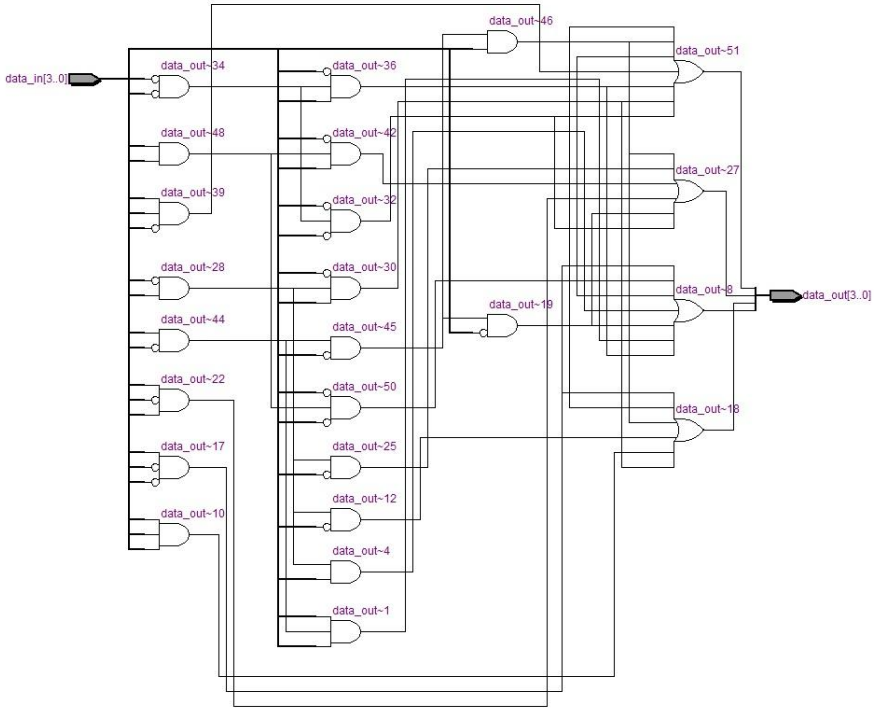


*Рис. 4. Структура аппаратного решения S-блока*

Обеспечивается пропускная способность алгоритма при тактовой частоте 20 МГц составляет 37,6 Мбит/с., а при тактовой частоте 160 МГц – 301,2 Мбит/с., количество тактов на блок – 34, эффективность решения (бит/сек на элемент) – 126599, потребляемая мощность: при 20 МГц – от 20 до 50 мВт, при 160 МГц – от 47 до 114.77 мВт.

**Выводы.** Получен ряд практически значимых результатов: проведено исследование малоресурсного алгоритма PRESENT, рассчитана трудоемкость, получено программное решение, достаточно эффективное для применения во встраиваемых устройствах, а также синтезирован аппаратный блок для системы на кристалле,

удовлетворяющий всем требованиям малоресурсной криптографии, проведена его отладка, моделирование и проведены необходимые эксперименты, доказавшие его работоспособность, эффективность и возможность применения на практике.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Bogdanov A., Knudsen L.R., Leander G., Paar C., Poschmann A., Robshaw M.J.B., Seurin Y., Vikkelsoe C.* PRESENT: An Ultra-Lightweight Block Cipher. 9. International Workshop on Cryptographic Hardware and Embedded Systems, CHES 2007, Vienna, Austria, LNCS, Springer-Verlag, September 10-13, 2007. – 18 p.
2. *Jacob J.* Performance Analysis of New Light Weight Cryptographic Algorithms. Sinhgad Institute of Technology, Pune, India. 2012. – 4 p.
3. *Panasayya Y., Jens-Peter K.* Lightweight Cryptography for FPGAs. Department of ECE, Volgenau School of IT&E George Mason University Fairfax, VA, USA. 2009. – 7 p.

Статью рекомендовал к опубликованию д.т.н., профессор Я.Е. Ромм.

**Бабенко Людмила Клементьевна** – Южный федеральный университет; e-mail: blk@tsure.ru. 347928, г. Таганрог, ул. Чехова, 2, корп. «И»; тел.: 88634312018; кафедра безопасности информационных технологий; профессор.

**Макаревич Олег Борисович** – e-mail: mak@tsure.ru; кафедра безопасности информационных технологий; профессор.

**Беспалов Дмитрий Анатольевич** – e-mail: bda82@mail.ru; кафедра безопасности информационных технологий; доцент.

**Чесноков Роман Дмитриевич** – e-mail: rd.chesnokov@gmail.com; кафедра безопасности информационных технологий; инженер.

**Трубников Ярослав Александрович** – e-mail: powerman23rus@gmail.com; кафедра безопасности информационных технологий; программист.

**Babenko Ludmila Klimentievna** – Southern Federal University; e-mail: blk@tsure.ru; 2, Chekhov street, Taganrog, 347928, Russia; phone: +78634312018; the department of security of information technologies; professor.

**Makarevich Oleg Borisovich** – e-mail: mak@tsure.ru; the department of security of information technologies; professor.

**Bespalov Dmitry Anatolievich** – e-mail: bda82@mail.ru. the department of security of information technologies; associate professor.

**Chesnokov Roman Dmitrievich** – e-mail: rd.chesnokov@gmail.com; the department of security of information technologies; engineer.

**Trubnikov Yaroslav Aleksandrovich** – e-mail: powerman23rus@gmail.com; the department of security of information technologies; programmer.