ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, РЕАЛИЗУЮЩЕЕ АЛГОРИТМ ШАМИРА В СТОЙКИХ ЧАСТНЫХ КРИПТОСИСТЕМАХ

*Гриднев В.А., Селиванов А.Ю.**

***Ключевые слова**: программное обеспечение, криптосистема, ключ шифрования, депонирование, пороговая схема разделения секрета, алгоритм, контроль аутентичности теней, безопасность, информационная система, пользователь.*

Аннотация.

Цели работы: облегчение работы пользователя при разделении секрета на основе прикладного программного обеспечения, реализующего систему депонирования ключей шифрования в стойких частных криптосистемах, использующих пороговую схему Шамира.

Методы: объектно-ориентированное программирование, имитационное моделирование.

Результаты: обоснована необходимость применения стойких частных криптосистем с депонированием ключей; приведены актуальные примеры использования таких систем; разработан программный комплекс для депонирования ключей на основе пороговой схемы Шамира; исследованы правовые вопросы депонирования ключей в частных криптосистемах.

Программный комплекс включает программу, предоставляющую простой интерфейс разделения и восстановления секрета, а также программу, встраиваемую в стороннее программное обеспечение в виде заголовочной библиотеки. Главной особенностью программного комплекса является простота, его использование не требует особой квалификации.

DOI: 10.21681/1994-1404-2021-3-50-59

Введение

В современном мире все более актуальными становятся проблемы цифровых преступлений и международного терроризма. В условиях тотальной распространенности доступа в сеть Интернет и развития общедоступных систем шифрования цифровое общение злоумышленников становится все труднее отслеживать [8]. В связи с этим проведено исследование эффективности применения пороговых схем разделения секрета на основе депонирования ключей шифрования [4, 11].

Депонирование ключей — это предоставление ключей шифрования или их частей (*теней*) третьей стороне[1]. Третья сторона имеет право использовать тени или ключи лишь при определенных обстоятельствах, в частном случае, по решению суда. Как правило, третьей стороной выступают государственные организации и правоохранительные органы. Конечно же, депонирование секрета — это сложная с этической точки зрения задача, часто граждане не хотят добровольно «делиться» секретом или его частями из-за вполне оправданных мыслей о слежке.

Данное исследование, помимо создания программного обеспечения для разделения секрета, имеет целью привести примеры разрешения названной сложной задачи.

Процедуры разделения ключей

Криптосистемы с депонированием ключей. Анализ показал, что депонирование ключей необходимо для обеспечения баланса между тайной личной переписки граждан и полной криптоанархией, позволяющей криминальным элементам безопасно общаться по

[1] *Shamir A. How to share a secret*. 1979. *Communications of the ACM* 22. P. 612-613.

* ***Гриднев Виктор Алексеевич,*** *кандидат технических наук, доцент, доцент кафедры информационных систем и защиты информации Тамбовского государственного технического университета, г. Тамбов, Российская Федерация.*
E-mail: vikadres@yandex.ru

Селиванов Александр Юрьевич, *студент 5 курса Тамбовского государственного технического университета, г. Тамбов, Российская Федерация.*
E-mail: istrebion@yandex.ru

открытым каналам связи, используя стойкое шифрование, таким образом угрожая интересам национальной безопасности.

В США начиная с 1992 г. были попытки депонировать ключ, используемый для шифрования данных, передаваемых по каналам *Integrated Services Digital Network*, факсам и любой связи того времени [4]. Для этого был разработан стандарт — *Escrowed Encryption Standard*[2]. Реализованы эти попытки были при помощи микросхемы *Clipper*, занимающейся шифрованием и расшифровкой данных. Агентство национальной безопасности США (*NSA*) также разработало дисциплину раскрытия уникального ключа микросхемы. Микросхема была разработана по технологии *Tempest* таким образом, чтобы было невозможно считать с нее данные по каналам побочных электромагнитных излучений и наводок.

Простейший пример схемы разделения секрета. Допустим, имеется тайное общество, строящее планы по захвату мира, которое состоит из N человек. Члены общества создали сообщение s, состоящее из бинарных символов. Затем они случайным образом подобрали N значений двоичных сообщений таким образом, что в сумме они давали изначальное сообщение s. Теперь, если все члены общества соберутся вместе, они без труда смогут подтвердить свое членство, просто сложив все N сообщений вместе. Если в итоге получится изначальное сообщение s, известное всем, то общество собралось в полном составе. В противном же случае — в общество пробрался «обманщик». У данной схемы есть множество проблем, но самая серьезная состоит в том, что если хотя бы один член группы захочет выйти из нее, то придется менять все N двоичных сообщений. Более того, если на сбор группы не придет хотя бы один из ее участников, подтвердить подлинность членов тайного общества не получится.

Существует более продуманная схема разделения секрета, позволяющая подтверждать подлинность участников нашего мнимого тайного общества, даже если оно не собирается в полном составе. Называется данная схема *пороговой*. Первые идеи схем, в которых число теней, необходимых для восстановления секрета, не равно числу участников (t и n, соответственно) принадлежат известному криптографу Ади Шамиру, одному из авторов известного алгоритма *RSA*.

Пороговая схема. Если собравшееся число участников таково, что возможно восстановление секрета, такую коалицию называют *разрешенной*. Следует заметить, что пороговые схемы, в которых разрешенные группы участников, имеющие число теней, равное или большее числу теней, необходимых для восстановления секрета, всегда могут однозначно восстановить зашифрованный ранее секрет, а неразрешенные не получают никакой информации о возможном значении секрета, называют *совершенными*[3].

Схема Шамира. Перейдем к конкретным примерам схем, которые могли бы использовать члены нашего гипотетического тайного общества. Основная идея состоит в том, что в геометрии двумерного пространства по двум точкам можно построить прямую, по трем параболу, по четырем кубическую параболу и так далее, повышая степени. Иными словами, для задания степенного многочлена k требуется число точек, равное $(k + 1)$[4].

Для того чтобы восстановить секрет могло только определенное число членов тайного общества n, секрет сворачивают в многочлен степени $n + 1$ над конечным полем G. Таким образом, чтобы восстановить секрет, необходимо будет собрать вместе n членов тайного общества, а значит, и значений многочлена в определенных точках. Если тайное общество не сможет собраться в нужном количестве, точек будет недостаточно, и восстановление секрета будет невозможно. В теории, число точек многочлена не ограничено, но ввиду конечности памяти ЭВМ и разумной достаточности (в нашей организации вряд ли будет большее число членов чем, к примеру, миллион) оно всегда ограничено размерностью поля Галуа порядка G.

Попробуем описать алгоритм более кратко. Допустим, мы имеем поле Галуа G. Возьмем n случайных элементов данного поля, обладающих свойством дискретности и не попадающих в ноль. Выберем произвольный набор элементов t поля Галуа G. Число элементов набора будет необходимым для восстановления секрета. Именно из этого множества чисел будет составляться искомый *пороговый* многочлен над полем Галуа степени $(t — 1)$, $1 < t \leq n$.

Допустим, искомый многочлен получен, теперь найдем его значения (x, y) в n точках. Остается лишь распределить полученные значения каждому из членов нашего тайного общества.

Заметим, что через две точки всегда можно провести неограниченное число квадратичных парабол (*рис*. 1). Но чтобы выбрать из них нужную параболу, понадобится третья точка.

Восстановление секрета осуществляется с помощью любой формулы интерполяции, например, при помощи наиболее популярной формулы Ньютона или Бесселя.

Основное достоинство пороговой схемы Шамира — масштабируемость. Если в нашем тайном сообществе появится новый участник, то нам будет необходимо добавить новый несекретный элемент к уже созданным ранее несекретным элементам. Если же, наоборот, какой-либо член нашего сообщества был исключен, то это не приводит к необходимости генерации нового

[2] Блэкли Д., Кабатянский Г. А. Обобщенные идеальные схемы, разделяющие секрет, и матроиды // Пробл. передачи информ. 1997. Т. 33. Вып. 3. С. 102—110.

[3] Блейхут Р. Теория и практика кодов, контролирующих ошибки (*Theory and Practice of Error Control Codes*). М. : Мир, 1986. 576 с.

[4] Ященко В. В., Варновский Н. П., Нестеренко Ю. В. Введение в криптографию. М. : МЦНМО «ЧеРо», 1999. 272 с.

набора точек. По сути, это просто ослабит схему, переведя ее из (n, t)-пороговой в $(n — 1, t — 1)$-пороговую.

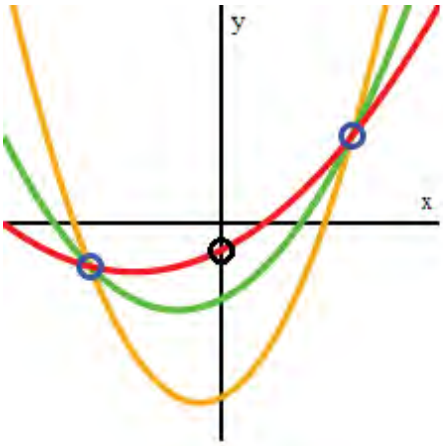

Рис. 1. Квадратичные параболы

Схема Блэкли. Какие еще варианты могут быть у нашей мнимой тайной организации? Известным фактом является то, что две непараллельные прямые на плоскости всегда пересекаются в одной точке. Таким образом, две некомпланарные плоскости всегда пересекаются по одной прямой, соответственно, три некомпланарные плоскости всегда пересекаются в одной точке. Более того, *n n*-мерных гиперплоскостей всегда пересекаются в одной точке (*рис*. 2).

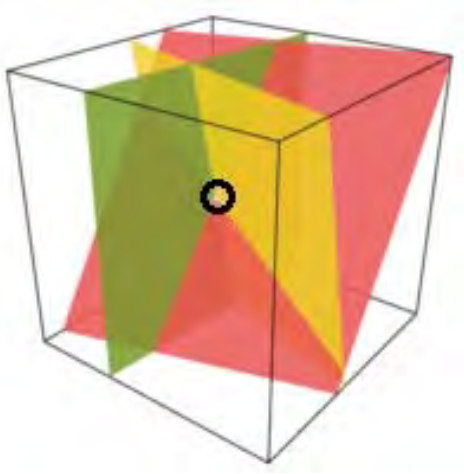

Рис. 2. Пересечение гиперплоскостей

Основная идея состоит в том, что одна из координат этой точки пересечения будет являться секретом [17].

Если мы закодируем секрет как несколько координат точки, то восстановление части секрета будет возможно уже по одной доле секрета, в более широком смысле — по одной гиперплоскости. Секретом здесь, в математическом смысле, будет являться взаимосвязанность координат точки пересечения [15].

Таким образом, с помощью схемы Блэкли мы сможем создать (t, n)-пороговую схему разделения секрета для любых t и n. Для этого придется использовать большие размерности пространства t. Каждому из n членов нашего сообщества необходимо будет выдать одну гиперплоскость, проходящую через точку, которая и будет являться секретом. Тогда любые t из общего числа n гиперплоскостей будут однозначно пересекаться в секретной точке.

Из всего сказанного можно сделать вывод, что схема Блэкли значительно менее эффективна, чем схема Шамира. Дело в том, что в схеме Шамира каждая тень того же размера, что и секрет. Размер тени имеет вполне практическое значение, ведь чем тень короче, тем проще ее запомнить. В схеме Блэкли каждая доля больше секрета в число раз, равное числу теней, необходимых для восстановления секрета. Более того, схема Блэкли не обладает такой же гибкостью, что и схема Шамира [2].

Существуют улучшения схемы Блэкли, позволяющие понизить ее ресурсоемкость и повысить гибкость, но, к сожалению, все они сильно усложняются по сравнению со схемой Шамира[5].

Схемы, основанные на решении систем уравнений. Математикам давно известно, что не существует решений систем линейных алгебраических уравнений (СЛАУ), в которых число уравнений меньше, чем число неизвестных.

Возьмем $n+1$ m-мерных векторов $V_0, V_1, \ldots V_n$ таким образом, чтобы квадратная матрица размера m, составленная из выбранных векторов, имела ранг m. Пусть выбранный вектор U имеет размер m, тогда секретом в нашей схеме будет матричное произведение транспонированного вектора U на другой выбранный ранее вектор V_i. Таким образом, долями секрета будут произведения транспонированного вектора U на V_i, где $i = 0, \ldots, n$[6].

Заметим, что по любым m долям можно составить СЛАУ размерности m, неизвестными в которой являются коэффициенты вектора U. Решив данную систему, можно найти коэффициенты U, а имея их, можно найти и секрет. При этом, естественно, СЛАУ неразрешима, если число неизвестных будет больше числа уравнений, т. е. если число долей меньше m [13].

Данный принцип реализован в схеме Карнина-Грина-Хеллмана [14].

Из-за хорошей способности к масштабируемости и отсутствия нелинейного роста алгоритмической сложности в схеме разделения секрета по Шамиру выбор авторов пал именно на нее.

Разработка программного обеспечения

Главной целью разработки программного комплекса была реализация лучшего способа разделения и восстановления ключа шифрования без усложнения алгоритма контролем законности операции восстановления и легитимности уполномоченного субъекта. Раз-

[5] Blakley G. R. Safeguarding cryptographic keys. Proceedings of the National Computer Conference, 48, 1979. P. 313-317.

[6] Karnin E. D., Greene J. W., Hellman M. E. On Secret Sharing Systems. F. Kschischang, IEEE, 1983. Vol. 29, Iss. 1. P. 35-41.

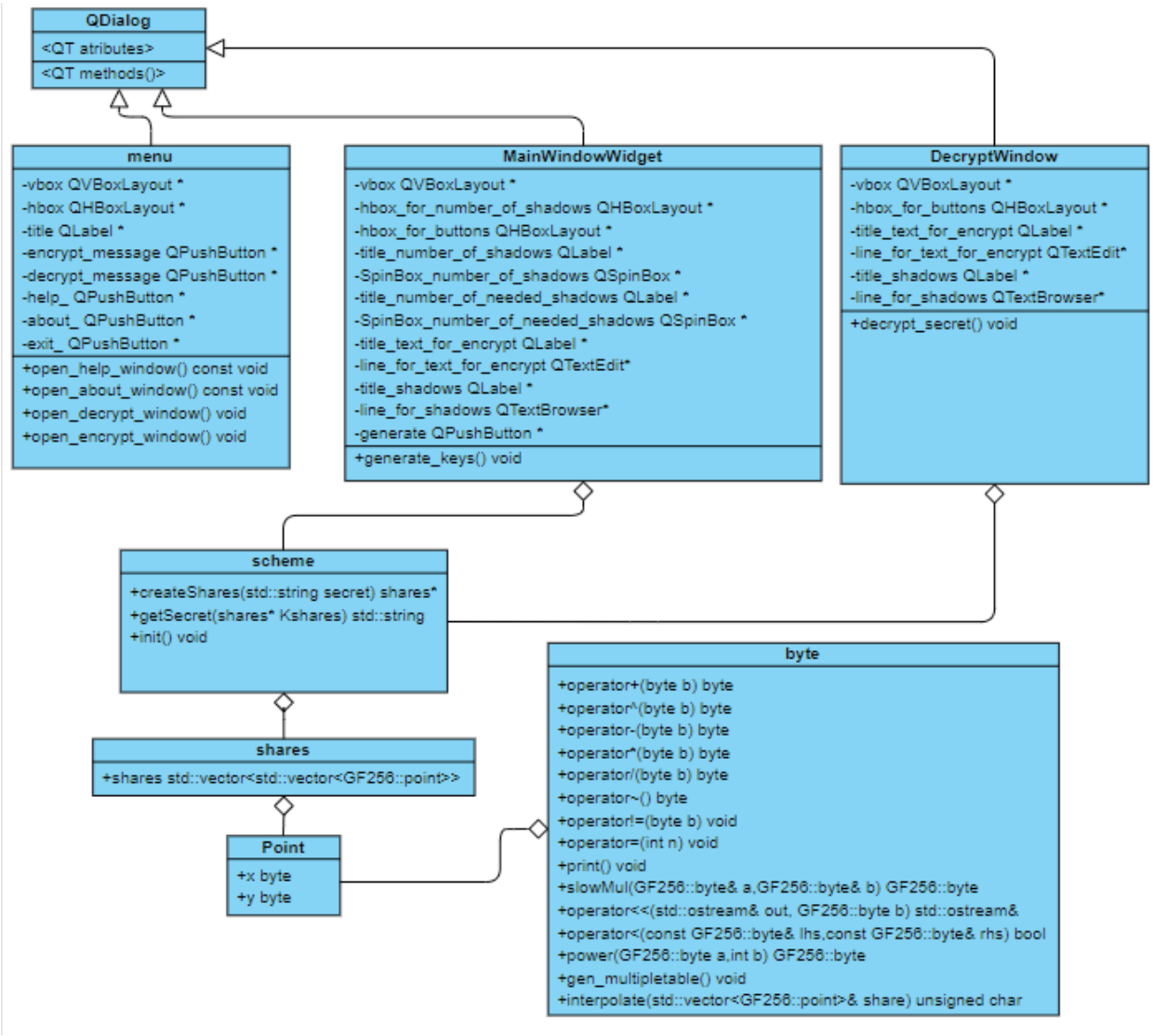

Рис. 3. Диаграмма классов программного комплекса

работанный программный комплекс использует простую пороговую схему, так что необходимо однократно потратить около двух-трех минут на разделение ключа. Ключи необходимо разделять один раз за срок их действия (около одного года). Конечно же, пороговые схемы сто́ит использовать в комплексе с уже готовыми приложениями для передачи информации. К примеру, вместе с любым известным мессенджером.

Программу, реализующую разделение секрета, можно оформить в виде небольшого кроссплатформенного модуля, который с легкостью можно будет интегрировать в любой сервис генерации ключей шифрования независимо от используемого языка программирования. Модули, реализующие разделение и восстановление секрета, должны быть оформлены раздельно, так как они выполняются на разных устройствах. При этом демонстрационную программу можно выполнить в виде единого модуля, реализующего все функции для имитационного моделирования.

Язык и инструментальные средства программирования. В качестве языка программирования для реализации программного обеспечения (ПО), был выбран язык *C++*. Это популярный компилируемый язык со статической типизацией [5]. Выбранный язык является достаточно низкоуровневым для решения с его помощью криптографических задач. Приложение было написано с использованием объектно-ориентированного подхода, что позволило придать архитектуре высокую скорость выполнения, гибкость и масштабируемость. Заметим, что в состав языка C++ включена библиотека *STL*. В данный набор шаблонов включена бóльшая часть контейнеров для работы с памятью, а также множество алгоритмов.

Программа была собрана при помощи компилятора *MinGW*; несмотря на то, что это компилятор для *Windows*, написанный код может использоваться и для других платформ, используя другие компиляторы.

В качестве инструментальных средств программирования была выбрана программная среда *QTCreator*, поддерживающая графический интерфейс.

Описание программного комплекса. Программный комплекс *ShamirSecretDepoKeys.exe* предназначен для

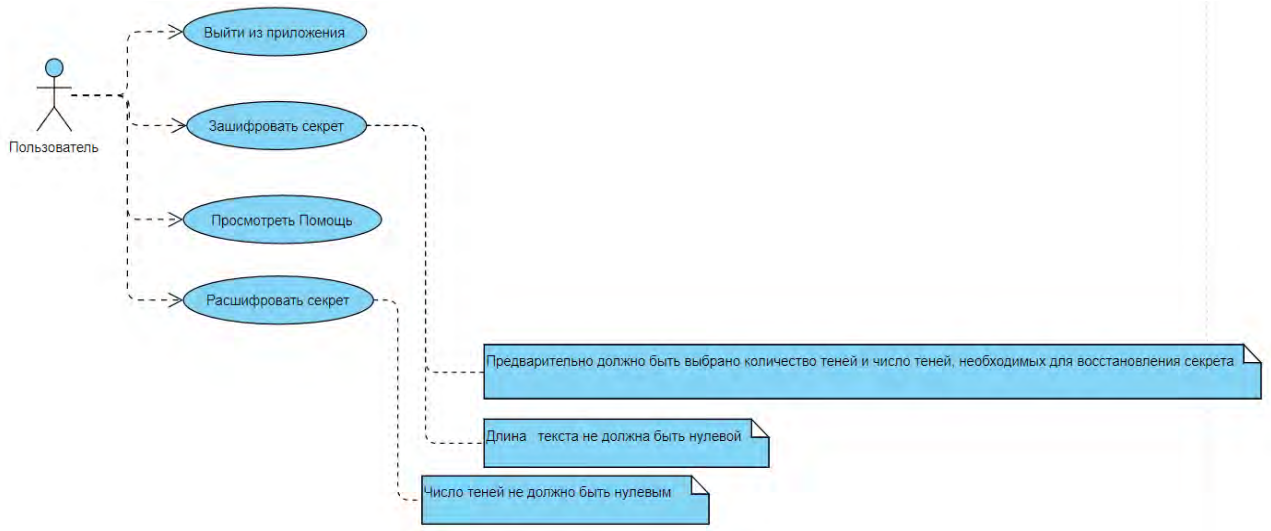

Рис. 4. Диаграмма вариантов использования программного комплекса

разделения и восстановления секрета по схеме Шамира. Данная программа написана на языке *C++* с использованием интегрированной среды разработки *QtCreator*.

Программный комплекс имеет модульную архитектуру и содержит 13 программных модулей:

– файлы исходного кода: shamir.cpp, menu.cpp, mainwindowwidget.cpp, main.cpp, GF(256).cpp, decryptwindow.cpp.;
– заголовочные файлы: shamir.h, mainwindowwidget.h, main.h, GF(256).h, decryptwindow.h, menu.h.;
– файл проекта: ShamirSecretDepoKeys.pro.

Модули *mainwindowwidget.h, menu.h, decryptwindow.h, decryptwindow.cpp, menu.cpp, mainwindowwidget.cpp* нужны для обеспечения логики содержимого окон графического интерфейса. Модули *shamir.h, shamir.cpp, main.h, GF(256).h, GF(256).cpp* предназначены для описания логики разделения и восстановления секрета.

Архитектура ПО обладает достаточной гибкостью и масштабируемостью.

Диаграмма классов и диаграмма вариантов использования. Диаграмма классов (от англ. *class diagram*) — это диаграмма языка *UML*, показывающая структуру классов программы. Данная диаграмма описывает поля и методы классов, типы возвращаемых значений, структуру наследований и интерфейсов [3]. Применяется не только для визуализации, но и для прямого или обратного планирования архитектуры. Авторы использовали данную диаграмму именно для прямого планирования архитектуры программного комплекса (*рис.* 3). Диаграмма вариантов использования (от англ. *use case diagram*) в *UML* — это диаграмма, отражающая отношения между акторами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне [6]. Диаграмма прецедентов разработанного ПО показана на *рис.* 4.

Рис. 5. Основное окно интерфейса программного комплекса

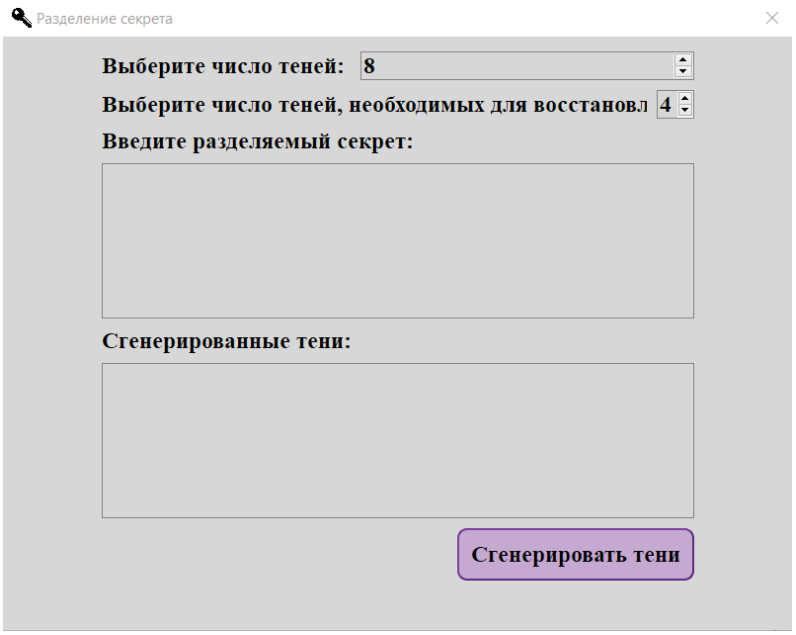

Рис. 6. Окно интерфейса «Шифрование секрета»

Интерфейс программного комплекса. Вид основного окна программного комплекса представлен на *рис*. 5.

На *рис*. 6 показан вид окна интерфейса «Разделение секрета». В этом окне пользователь может разделить секрет на тени, выбрав общее число теней и число теней, необходимых для восстановления секрета. Программа позволяет выбирать число теней в диапазоне от 5 до 10 единиц. Допустимый диапазон числа теней, необходимых для восстановления, находится в диапазоне от 2 до 5. Это сделано в интересах моделирования. Реальные схемы целесообразно сделать двух видов: 2 из 4 и 3 из 5.

Сгенерируем ключ шифрования длиной 256 *бит* и попробуем зашифровать секретную фразу «*Sapere aude*!» алгоритмом *AES* с помощью сгенерированного ключа (*рис*. 7).

Следующий пример доказывает эффективность программного комплекса разделения закрытого ключа:

"*AADRkK7s32waZoO2d/IvDejqbWEZa8L2*",

выбрав общее число теней 8, а число теней, необходимых для восстановления секрета — 4. Такая пороговая схема называется (4, 8).

Результат генерации теней показан на *рис*. 8. Рассмотрим конкретную тень: 8 1 114 198 103 46 157 156 118 227 93 65 210 140 43 122 42 14 7 58 41 206 94 82 241 99 34 242 128 155 64 139 191 101. *Первое* число означает общее количество теней в пороговой схеме, *второе* число — номер конкретной тени. Все дальнейшие числа являются, собственно, тенью.

На *рис*. 9 показано окно интерфейса «Восстановление секрета». В данном окне пользователь вводит тени, разделяя их знаком новой строки. Если какая-то из теней повреждена или число теней меньше ранее заданного количества, восстановления секрета не произойдет.

Как видно, ключ был успешно восстановлен по четырем теням из восьми. Проверим результат восстановления ключа, расшифровав исходное сообщение (*рис*. 10).

На *рис*. 11 приведен пример окна интерфейса «Помощь».

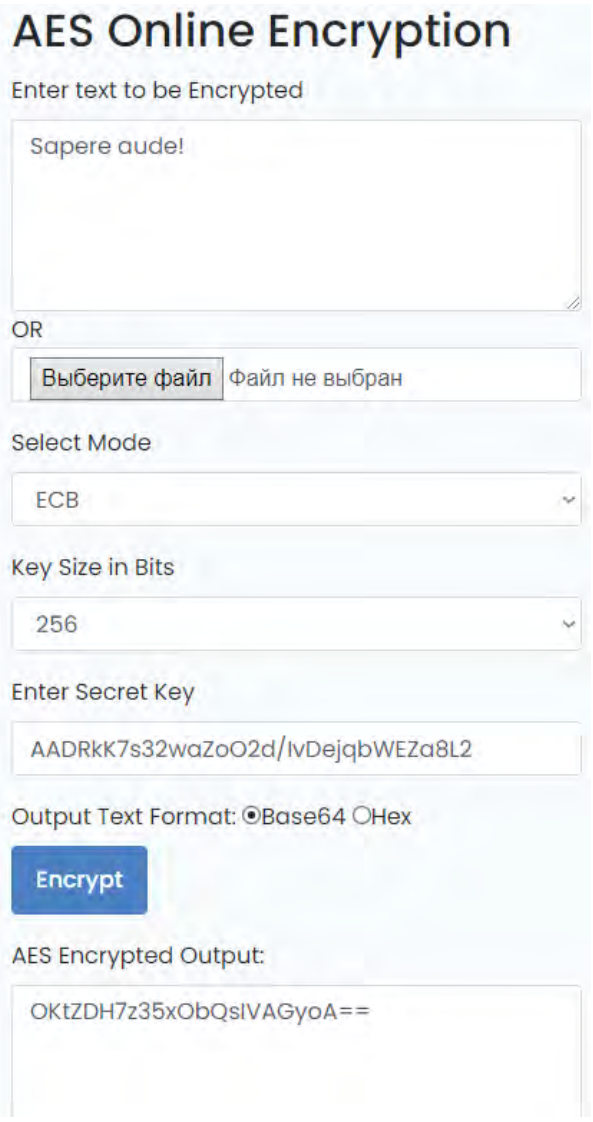

Рис. 7. Шифрование сообщения

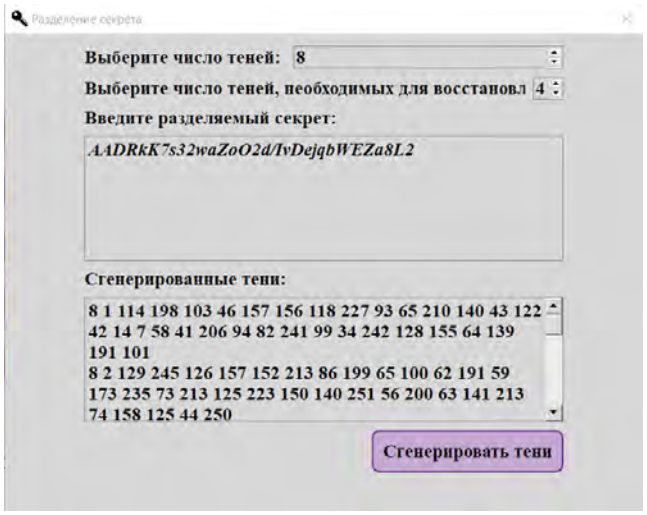

Рис. 8. Результат генерации теней

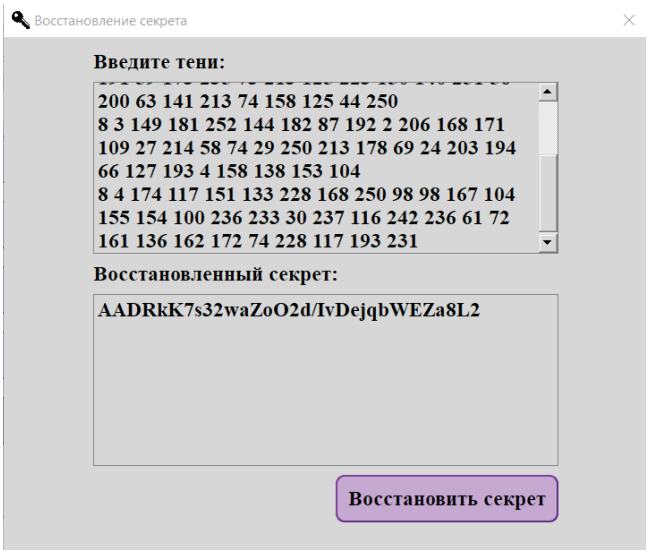

Рис. 9. Интерфейс окна «Восстановление секрета»

Как видно, восстановление секретной фразы произошло корректно.

Примеры применения разработанного программного комплекса

Депонирование ключей с целью обеспечения безопасности граждан. В настоящее время распространение получили криптографические алгоритмы, позволяющие обеспечить такой уровень криптостойкости, который не под силу взломать спецслужбам разных стран. Это создает определенные проблемы в таких сферах, как борьба с мировым терроризмом. Решение данных проблем — этически сложная задача, поскольку необходимо сохранять секретность переписки, давая разрешение спецслужбам на ее чтение лишь при наличии веских улик. Но как заставить пользователей системы добровольно передать тени ключей в центр хранения теней? Данная задача должна решаться с помощью механизма запрета использования определенной информационной системы пользователям, не передавшим тени центру хранения теней [12].

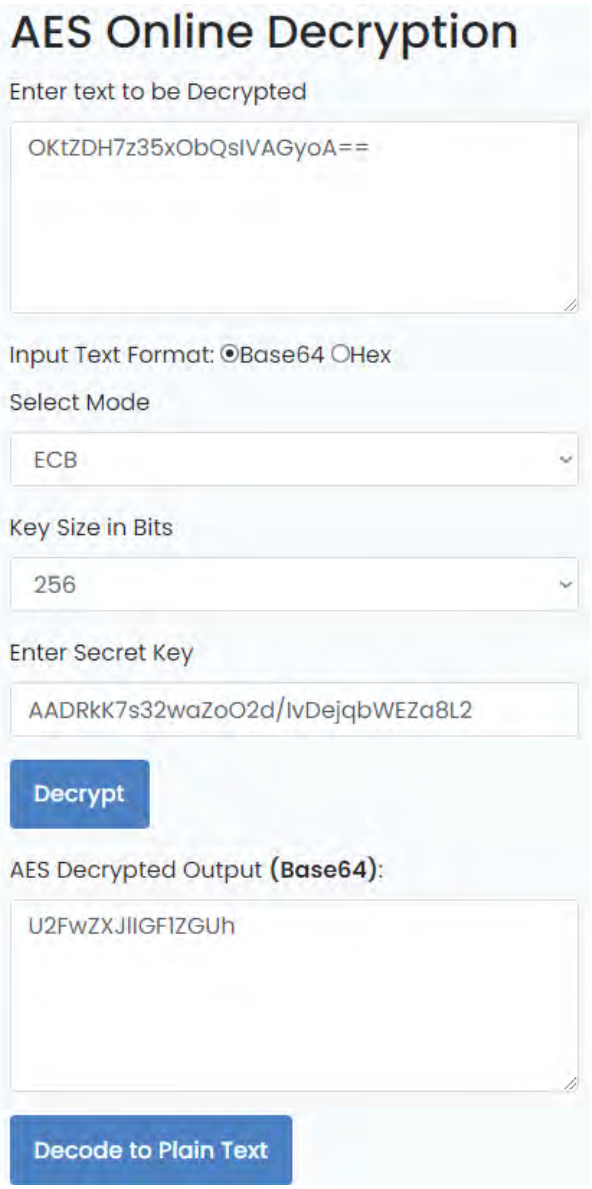

Рис. 10. Восстановление секретной фразы

К примеру, определенный пользователь хочет использовать информационную систему (в частном случае — мобильное приложение) со стойким шифрованием. При регистрации пользователь дает свое согласие на депонирование теней ключа шифрования и их распределение по центрам хранения теней. В противном случае пользователь не сможет воспользоваться информационной системой.

Пользователя необходимо также уведомить, что за попытку передачи ложных теней ключа его ждет ответственность, а также уверить в том, что ключ не будет восстановлен без соответствующего решения суда. Увеличение числа теней ключа увеличивает стойкость системы в целом, так как условному нарушителю будет необходимо украсть большее число теней, проникнув в соответственно большее число информационных систем. Определенная часть пользователей информационных систем может радикально выступать против передачи теней в государственные центры хранения.

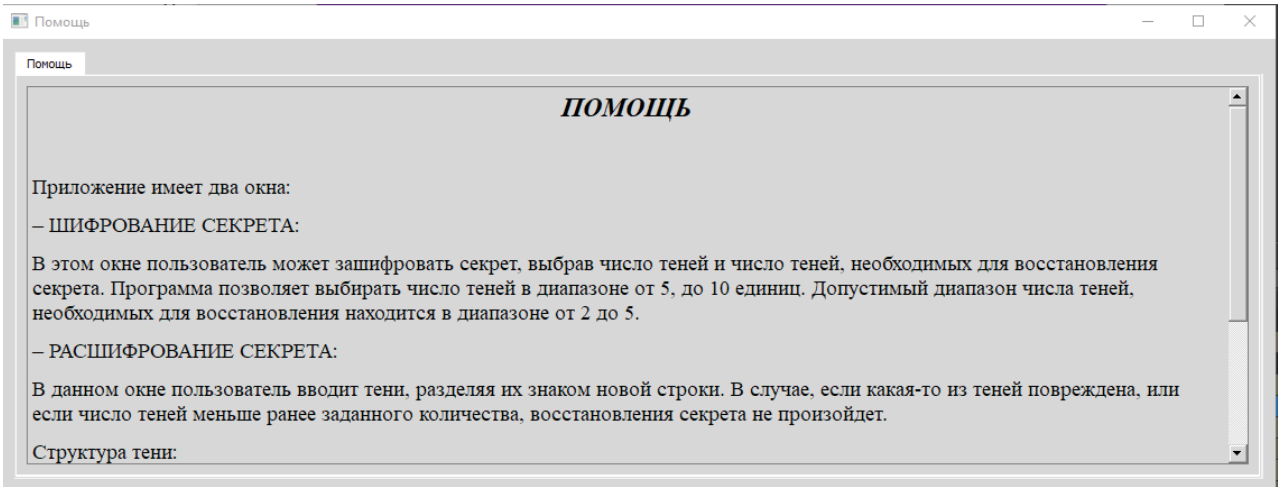

Рис. 11. Окно интерфейса «Помощь»

Для купирования данного эффекта стоит передавать тени «скрытно», обязательно указав все условия в лицензионном соглашении[7] [17].

Депонирование с целью восстановления ключа при его утрате. В последние годы все большую популярность набирают различные криптовалюты, основанные на системе блокчейна [8, 9]. Криптовалюты по сути своей являются децентрализованными. В случае потери доступа к банковскому счету пользователь банка пишет запрос в филиал, подтверждает свою личность, после чего банк восстанавливает доступ.

В случае криптовалют при потере доступа к кошельку средства, содержащиеся на нем, будут утеряны навсегда. Таким образом, пользователь сам заинтересован в раздельном сохранении теней секрета, чтобы не потерять доступ к криптовалюте, например, при повреждении оборудования, на котором хранится ключ. Пороговая схема разделения секрета Шамира позволит обеспечить безопасное раздельное хранение [16]. Если какой-либо центр хранения потеряет ключ или будет взломан, это не приведет к негативным последствиям для пользователей.

Данная идея может позволить вывести криптовалюты из теневой части рынка, введя их в правовое поле. Необходимо лишь создать соответствующие *правовые регламенты* [7, 8], которые не позволят использовать криптовалюты без выдачи теней государственным центрам хранения.

Эта схема может также использоваться для «холодного» хранения теней на внешних носителях данных, таких, как оптические диски. Это позволит восстановить секрет даже при повреждении или утере части накопителей.

Заключение

Решение проблемы депонирования ключей — этически сложная задача, поскольку необходимо сохранять секретность переписки, давая разрешение спецслужбам на ее чтение лишь при наличии веских причин, подтвержденных решением суда. Тем не менее депонирование может использоваться не только для обеспечения возможности чтения личной переписки спецслужбами, но и обеспечения безопасности криптовалют.

Авторы статьи выступают за свободу Интернета, но свобода возможна лишь в правовых рамках, иначе эта свобода превратится в криптоанархию.

Еще в 1997 г. известный криптограф Б. Шнайер четко указал на то, что государственные системы шифрования не получат широкого распространения в будущем [14]. Тем не менее депонирование ключей возможно осуществить, если оно будет выгодно самому пользователю. Более того, весь мир движется в сторону контроля Интернета, и Россия не станет исключением ввиду вступления в силу «закона Яровой» 2016 г. На текущий момент контроль информационного поля — решенный вопрос, осталось лишь выбрать методы. Одним из вариантов данного контроля является расширение «закона Яровой», обязующее владельцев информационных систем обмена сообщениями передавать тени ключей в государственные центры хранения. В случае невыполнения данных требований в законе сто́ит четко прописать размер санкций.

Сто́ит добавить, что весь информационный обмен между субъектами взаимоотношений, приведенных в примерах использования депонирования ключей, может быть реализован в электронном виде, что неизбежно потребует принятия мер по обеспечению безопасности информации, передаваемой по каналам связи [1, 10].

Актуальным остается вопрос защиты *целостности* и *корректности* депонируемых теней. Необходимо предусмотреть защиту депонируемых теней от умышленного или случайного искажения, а также возможность обнаружить любого недобросовестного участника схемы. Ведь любой из участников схемы может попытаться помешать успешному восстановлению ключа. Например, пользователь может после депонирования ключа шифрования использовать совсем другой ключ. И когда после соблюдения всех предусмотренных фор-

[7] B. McCullough, "The NSA tried this before — what the 90s debate over the clipper chip can teach us about digital privacy". Internet History Podcast, URL: http://www.internethistorypodcast.com/2014/08/the-nsa-tried-this-before-what-the-90s-debate-over-the-clipper-chip-can-teach-us-about-digital-privacy-debates/ (дата обращения: 09.05.2021).

мальностей ключ не восстановится, заявить, что он не знает причину, по которой это могло произойти.

Необходимо также предусмотреть электронное взаимодействие всех участников схемы, исключающее необходимость личных встреч и долгого ожидания. Получение ключа шифрования пользователем, депонирование теней, обращение в суд за разрешением на восстановление ключа и само восстановление ключа шифрования сотрудником спецслужб должны происходить путем сетевого электронного взаимодействия.

Литература

1. Алексеев В.В., Стрельцов А.А., Емельянов Е.В. Правовой подход к построению защиты информации в организации // Правовая информатика. 2020. № 2. С. 54—61. DOI: 10.21681/1994-1404-2020-2-54-61 .
2. Болотов А.А., Гашков С.Б., Фролов А.Б., Часовских А.А. Элементарное введение в эллиптическую криптографию. Алгебраические и алгоритмические основы. М. : КомКнига, 2006. 328 с.
3. Буч Г., Рамбо Дж., Джекобсон А. Язык UML. Руководство пользователя. 2-е изд. СПб. : ДМК Пресс, 2004. 432 с.
4. Гриднев В.А., Володин И.С., Желудкова А.М. Организационно-правовые вопросы внедрения стойких частных криптосистем // Правовая информатика. 2021. № 1. С. 51—60. DOI: 10.21681/1994-1404-2021-1-51-60 .
5. Керниган Б.У., Ритчи Д.М. Язык программирования С. М. : Вильямс, 2009. 304 с.
6. Ларман К. Применение UML 2.0 и шаблонов проектирования. 3-е изд. М. : Вильямс, 2006. 736 с.
7. Ловцов Д.А. Системология правового регулирования информационных отношений в инфосфере : монография. М.: РГУП, 2016. 316 с. ISBN 978-5-93916-505-1.
8. Ловцов Д.А. Информационно-правовые основы правоприменения в цифровой сфере // Мониторинг правоприменения. 2020. № 2(35). С. 44—52. DOI: 10.21681/2226-0692-2020-2-44-52 .
9. Ловцов Д.А. Информационная безопасность автоматизированных блокчейн-систем: угрозы и способы повышения // Тр. II Междунар. науч.-прак. конф. «Трансформация национальной социально-экономической системы России» (22 ноября 2019 г.) / РГУП. М. : РГУП, 2020. С. 464—473. ISBN 978-5-93916-823-6.
10. Ловцов Д.А. Проблема гарантированного обеспечения информационной безопасности крупномасштабных автоматизированных систем // Правовая информатика. 2017. № 3. С. 66—74. DOI: 10.21681/2226-0692-2017-3-66-74 .
11. Ловцов Д.А., Кабелев Д.Б. Технология и проблемы рационального управления ключевой информацией в АСУ // Труды XXIX Всеросс. науч.-техн. конф. «Проблемы эффективности и безопасности функционирования сложных технических и информационных систем» (24—25 июня 2010 г.) в 5-ми тт. Т. 4 / РАО. Серпухов : Серп. воен. ин-т, 2010. С. 144—148.
12. Сунаид Х.А.С., Яковлев А.В. Состояние и перспективы развития государственной информационно-телекоммуникационной системы Йемена // Труды Науч.-прак. конф. «Информационные системы и процессы» (15 июня 2018 г.). Тамбов : Межд. инф. Нобелевский центр, 2018. С. 76—89.
13. Шенец Н.Н. Об идеальных модулярных схемах разделения секрета в кольцах многочленов от нескольких переменных // Труды Междунар. конгресса по информатике: «Информационные системы и технологии» (31 октября 2011 г.) / БГУ. Т. 1. Минск : БГУ, 2011. С. 169—173.
14. Шнайер Б. Разделение секрета // Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. М. : Триумф, 2002. С. 93—96.
15. Шнайер Б. Алгоритмы разделения секрета // Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. М. : Триумф, 2002. С. 588—591.
16. Хучиров А.Г., Яковлев Ал.В., Яковлев Ан.В. Реализация метода кластерного анализа в математической модели процессов взаимообмена информационными ресурсами в АСУ специального назначения // Перспективы науки. № 1. Тамбов : Фонд развития науки и культуры, 2017. С. 75—79. ISSN 2077-6810.
17. Чмора А.Л. Современная прикладная криптография. 2-е изд., стер. М. : Гелиос АРВ, 2002. С. 123—124.

Рецензент: ***Алексеев Владимир Витальевич,*** *доктор технических наук, профессор, заведующий кафедрой информационных систем и защиты информации Тамбовского государственного технического университета, г. Тамбов.*
E-mail: vvalex1961@mail.ru

SOFTWARE IMPLEMENTING THE SHAMIR ALGORITHM IN STRONG PRIVATE CRYPTOSYSTEMS

Viktor Gridnev, *Ph.D. (Technology), Associate Professor at the Information Systems and Information Protection Department of the Tambov State Technical University, Tambov, Russian Federation.*
E-mail: vikadres@yandex.ru

Aleksandr Selivanov, *5th year student at the Tambov State Technical University, Tambov, Russian Federation.*
E-mail: istrebion@yandex.ru

Keywords: *software, cryptosystem, encryption key, depositing, secret sharing threshold scheme, algorithm, shadow authentication control, security, information system, user.*

Abstract.

Purposes of the paper: facilitating the user's work in sharing a secret based on application software implementing a system for depositing encryption keys in strong private cryptosystems using the Shamir threshold scheme.

Methods used: object-oriented programming, simulation modelling.

Results obtained: a justification was given for the need to use strong private cryptosystems with key depositing. Topical cases of using such systems were presented. A software package for key depositing based on the Shamir threshold scheme was developed. Legal issues of depositing keys in private cryptosystems were studied.

The software package includes a programme providing a simple interface for separating and restoring secrets as well as a programme embedded in third-party software in the form of a header library. The main feature of the software package is simplicity, using it doesn't require special qualifications.

References

1. Alekseev V.V., Strel'tsov A.A., Emel'ianov E.V. Pravovoi podkhod k postroeniiu zashchity informatsii v organizatsii. Pravovaia informatika, 2020, No. 2, pp. 54-61. DOI: 10.21681/1994-1404-2020-2-54-61 .
2. Bolotov A.A., Gashkov S.B., Frolov A.B., Chasovskikh A.A. Elementarnoe vvedenie v ellipticheskuiu kriptografiiu. Algebraicheskie i algoritmicheskie osnovy. M. : KomKniga, 2006. 328 pp.
3. Buch G., Rambo Dzh., Dzhekobson A. Iazyk UML. Rukovodstvo pol'zovatelia. 2-e izd. SPb. : DMK Press, 2004. 432 pp.
4. Gridnev V.A., Volodin I.S., Zheludkova A.M. Organizatsionno-pravovye voprosy vnedreniia stoikikh chastnykh kriptosistem. Pravovaia informatika, 2021, No. 1, pp. 51-60. DOI: 10.21681/1994-1404-2021-1-51-60 .
5. Kernigan B.U., Ritchi D.M. Iazyk programmirovaniia C. M. : Vil'iams, 2009. 304 c.
6. Larman K. Primenenie UML 2.0 i shablonov proektirovaniia. 3-e izd. M. : Vil'iams, 2006. 736 pp.
7. Lovtsov D.A. Sistemologiia pravovogo regulirovaniia informatsionnykh otnoshenii v infosfere : monografiia. M.: RGUP, 2016. 316 pp. ISBN 978-5-93916-505-1.
8. Lovtsov D.A. Informatsionno-pravovye osnovy pravoprimeneniia v tsifrovoi sfere. Monitoring pravoprimeneniia, 2020, No. 2(35), pp. 44-52. DOI: 10.21681/2226-0692-2020-2-44-52 .
9. Lovtsov D.A. Informatsionnaia bezopasnost' avtomatizirovannykh blokchein-sistem: ugrozy i sposoby povysheniia. Tr. II Mezhdunar. nauch.-prak. konf. "Transformatsiia natsional'noi sotsial'no-ekonomicheskoi sistemy Rossii" (22 noiabria 2019 g.), RGUP. M. : RGUP, 2020, pp. 464-473. ISBN 978-5-93916-823-6.
10. Lovtsov D.A. Problema garantirovannogo obespecheniia informatsionnoi bezopasnosti krupnomasshtabnykh avtomatizirovannykh sistem. Pravovaia informatika, 2017, No. 3, pp. 66-74. DOI: 10.21681/2226-0692-2017-3-66-74 .
11. Lovtsov D.A., Kabelev D.B. Tekhnologiia i problemy ratsional'nogo upravleniia kliuchevoi informatsiei v ASU. Trudy XXIX Vseross. nauch.-tekhn. konf. "Problemy effektivnosti i bezopasnosti funktsionirovaniia slozhnykh tekhnicheskikh i informatsionnykh sistem" (24-25 iiunia 2010 g.) v 5-mi tt., t. 4, RAO. Serpukhov : Serp. voen. in-t, 2010, pp. 144-148.
12. Sunaid Kh.A.S., Iakovlev A.V. Sostoianie i perspektivy razvitiia gosudarstvennoi informatsionno-telekommunikatsionnoi sistemy Iemena. Trudy Nauch.-prak. konf. "Informatsionnye sistemy i protsessy" (15 iiunia 2018 g.). Tambov : Mezhd. inf. Nobelevskii tsentr, 2018, pp. 76-89.
13. Shenets N.N. Ob ideal'nykh moduliarnykh skhemakh razdeleniia sekreta v kol'tsakh mnogochlenov ot neskol'kikh peremennykh. Trudy Mezhdunar. kongressa po informatike: "Informatsionnye sistemy i tekhnologii" (31 oktiabria 2011 g.), BGU, t. 1. Minsk : BGU, 2011, pp. 169-173.
14. Shnaier B. Razdelenie sekreta. Prikladnaia kriptografiia. Protokoly, algoritmy, iskhodnye teksty na iazyke Si. M. : Triumf, 2002, pp. 93-96.
15. Shnaier B. Algoritmy razdeleniia sekreta. Prikladnaia kriptografiia. Protokoly, algoritmy, iskhodnye teksty na iazyke Si. M. : Triumf, 2002, pp. 588-591.
16. Khuchirov A.G., Iakovlev Al.V., Iakovlev An.V. Realizatsiia metoda klasternogo analiza v matematicheskoi modeli protsessov vzaimoobmena informatsionnymi resursami v ASU spetsial'nogo naznacheniia. Perspektivy nauki, No. 1. Tambov : Fond razvitiia nauki i kul'tury, 2017, pp. 75-79. ISSN 2077-6810.
17. Chmora A.L. Sovremennaia prikladnaia kriptografiia. 2-e izd., ster. M. : Gelios ARV, 2002, pp. 123-124.