Сергей В. Гребнев
Московский институт электроники и математики им. А.Н. Тихонова,
Таллинская ул., 34, г. Москва, 123458, Россия
e-mail: s.v.grebnev@mail.ru, https://orcid.org/0000-0002-9482-1939

О КРИПТОГРАФИЧЕСКИХ СВОЙСТВАХ СХЕМЫ ВЫРАБОТКИ ОБЩЕГО КЛЮЧА «ЛИМОННИК-3»

DOI: http://dx.doi.org/10.26583/bit.2019.2.01

*Аннотация***.** В настоящей статье исследуется протокол выработки общего ключа с аутентификацией на основе открытого ключа «Лимонник-3», вошедший в состав Рекомендаций по стандартизации Р 1323565.1.004-2017 «Схемы выработки общего ключа с аутентификацией на основе открытого ключа», утвержденных в 2017 году Росстандартом, наряду с протоколами класса «Эхинацея». Данный протокол использует стандартизированные криптографические решения, при этом не требует использования цифровой подписи как отдельного примитива, позволяет двум участникам использовать различные эллиптические кривые для выработки и сертификации открытого ключа. В статье описывается протокол «Лимонник-3», исследуются заложенные в его основу синтезные решения и криптографические и эксплуатационные требования, предъявляемые к протоколу при его разработке, исследуются вопросы криптографической стойкости и эффективности. При условии применения предлагаемых в статье параметров и алгоритмов показана стойкость протокола относительно известных классов атак, в том числе задачи определения секретного ключа, сводящегося к дискретному логарифмированию, KCI- и UKS-атак. Получено формальное доказательство стойкости протокола в модифицированной модели Канетти-Кравчика в предположении о вычислительной сложности интервальной распознавательной задачи Диффи-Хеллмана, связанной с дискретным логарифмированием в группе точек эллиптических кривых. Автоматизированная верификация протокола «Лимонник-3» также показала соответствие заданным требованиям и отсутствие возможных векторов атаки. Рассмотрены перспективы применения и модернизации протокола в условиях возможного появления квантового компьютера. Показано, что протокол «Лимонник-3» является гибким, стойким криптографическим решениям, удовлетворяющим требованиям, предъявляемым к современным протоколам выработки общего ключа.

*Ключевые слова**:** аутентификация, выработка общего ключа, криптографический протокол, модель Канетти-Кравчика, схема Диффи-Хеллмана, схема MTI/A0, эллиптические кривые.*

Для цитирования: ГРЕБНЕВ, Сергей Владимирович. О КРИПТОГРАФИЧЕСКИХ СВОЙСТВАХ СХЕМЫ ВЫРАБОТКИ ОБЩЕГО КЛЮЧА «ЛИМОННИК-3». Безопасность информационных технологий, [S.l.], v. 26, n. 2, p. 6-20, mar. 2019. ISSN 2074-7136. Доступно на: <https://bit.mephi.ru/index.php/bit/article/view/1182>. Дата доступа: 05 june 2019. doi:http://dx.doi.org/10.26583/bit.2019.2.01.

Sergey V. Grebnev
Moscow Tikhonov Institute of Electronics and Mathematics,
Tallinskaya, 34, Moscow, 123458, Russia
e-mail:s.v.grebnev@mail.ru, https://orcid.org/0000-0002-9482-1939

On the cryptographic properties of “Limonnik-3” AKE scheme

DOI: http://dx.doi.org/10.26583/bit.2019.2.01

Abstract. We study the “Limonnik-3” authenticated key exchange protocol which is a part of Standardization recommendations R 1323565.1.004-2017 “Authenticated key agreement schemes based on public keys”, officially adopted in Russia in 2017, alongside with the “Echinacea” family of protocols. The protocol uses standardized cryptographic solutions, but does not require digital signature as a primitive, allows two parties to

use distinct elliptic curves. The paper describes the protocol the “Limonnik-3”, studies its design rationale, basic requirements used at the stage of protocol design, its cryptographic properties and efficiency. Provided that proposed in the paper parameters and algorithms are used, security against known classes of attacks, including secret key recovery, reduced to the elliptic curve discrete logarithm problem, KCI- and UKS-attacks, is demonstrated. A formal security proof in a modified Canetti-Krawczyk model is deduced, provided that the gap decision Diffie-Hellman problem, connected to the discrete logarithm in the group of points of an elliptic curve, is computationally hard. Automated verification of the protocols shows its security and absence of possible vectors of attack. A brief overview of post-quantum prospectives of the protocol is given. Thus, the paper shows that “Limonnik-3” is a robust and secure cryptographic solution, which satisfies all of the requirements that apply to the modern key exchange protocols.

Keywords: authentication, Canetti-Krawczyk model, cryptographic protocol, Diffie-Hellman scheme, elliptic curves, key exchange, MTI/A0 protocol.

For citation: GREBNEV, Sergey V. On the cryptographic properties of “Limonnik-3” AKE scheme. IT Security (Russia), [S.l.], v. 26, n. 2, p. 6-20, mar. 2019. ISSN 2074-7136. Available at: <https://bit.mephi.ru/index.php/bit/article/view/1182>. Date accessed: 03 june 2019. doi:http://dx.doi.org/10.26583/bit.2019.2.01.

Введение

Протоколы выработки общего ключа предназначаются для выработки ключа, известного лишь двум абонентам, как правило, не имеющим заранее распределенной ключевой информации. Выработанный ключ может быть использован для обеспечения конфиденциальности канала связи между этими абонентами.

Хорошо известным примером такого протокола является классический протокол Диффи-Хеллмана [1]. Существенным его недостатком, отмеченным еще в оригинальной работе, является отсутствие взаимной аутентификации абонентов, что позволяет нарушителю выполнять атаки методом «человек посередине» (например, таким образом была взломана первая версия протокола SSH).

Аутентифицированная выработка общего ключа позволяет одновременно с формированием общего ключа провести взаимную аутентификацию абонентов друг перед другом, при этом обеспечивая защиту от атаки типа «человек посередине». Отметим, что аутентификация может быть выполнена и после выработки общего ключа, однако такой метод, очевидно, является менее эффективным с точки зрения коммуникационной и вычислительной нагрузки. При этом механическое добавление в протокол механизмов аутентификации, например, использование цифровой подписи [2], может создать дополнительные траектории атаки, примером которых могут являться UKS- и KCI-атаки (подробнее рассмотрим далее).

В работе описана схема криптографического протокола «Лимонник-3» аутентифицированной выработки общего ключа двумя абонентами по открытому каналу связи (далее - протокол выработки общего ключа), соответствующая принятым в 2017 году рекомендациям по стандартизации [1]. Криптографическая стойкость схемы основана на предположении о сложности задачи дискретного логарифимирования в группе точек эллиптической кривой. В настоящей работе описываются схемы протокола и исследуются ее базовые криптографические свойства.

[1] Рекомендации по стандартизации Р 1323565.1.004-2017 «Информационная технология. Криптографическая защита информации. Схемы выработки общего ключа с аутентификацией на основе открытого ключа».

1. Общие определения и обозначения

В протоколе выработки общего ключа принимают участие две стороны (два абонента) – *инициатор* A (клиент) и *ответчик* B (сервер). Стороны идентифицируются при помощи *идентификаторов* (текстовых строк) $\mathrm{Id}_A, \mathrm{Id}_B$ соответственно. Как инициатор, так и ответчик могут иметь как *долговременные*, так и *сеансовые* ключи. Долговременные ключи используются в течение длительного срока времени, сеансовые ключи вырабатываются заново для каждого сеанса протокола. Долговременные секретные ключи обозначаются s_A, s_B, соответствующие открытые ключи – S_A, S_B. Сеансовые секретные ключи обозначаются k_A, k_B, соответствующие открытые ключи – K_A, K_B. Сертификаты открытых ключей абонентов обозначаются $\mathrm{Cert}_A, \mathrm{Cert}_B$. Считаем, что сертификат содержит также и идентификатор абонента: $\mathrm{Cert}_A = \mathrm{Cert}(\mathrm{Id}_A, S_A)$.

Общее секретное значение, являющееся результатом выполнения протокола, обозначим K. Будем далее называть это значение *общим ключом*. Вспомогательные значения, выработанные с участием обоих абонентов в ходе выполнения протокола и, возможно, использованные для выработки общего ключа, будем называть *общей ключевой информацией*.

Вспомогательный ключ, предназначенный только для подтверждения ключа, обозначим M. Обозначим $\mathrm{enc}_K(msg)$ – результат зашифрования сообщения msg на ключе K, $\mathrm{dec}_K(ct)$ – результат расшифрования шифртекста ct на ключе K, $\mathrm{mac}_M(msg)$ – код аутентификации сообщения msg на ключе M.

Обозначим h_i – фиксированные и обязательно различные строки, соответствующие заголовкам сообщений на i-ом шаге протокола. Обозначим OI – информацию, связанную с контекстом сеанса (метки времени, сетевые адреса и др.), известную обоим абонентам. Обозначим также K – опционально используемое заранее распределенное секретное значение.

2. Схема Диффи-Хеллмана

Основным криптографическим примитивом, используемым для построения протокола выработки общего ключа, является *схема Диффи-Хеллмана* [1]. Данная схема реализуется в произвольной конечной группе $G = \langle P \rangle_q$, в которой задача вычисления дискретного логарифма является вычислительно сложной и состоит в том, что абоненты A и B вырабатывают случайные секретные ключи $k_A, k_B \in [1, q-1]$ соответственно, обмениваются посылками $K_A = k_A P$, $K_B = k_B P$ и вычисляют общий ключ $Q = k_A(k_B P) = k_B(k_A P)$.

Секретные и открытые ключи какого-либо абонента (например, A) всегда связаны соотношением $S_A = s_A P$, $k_A = k_A P$, где P – некоторый элемент группы G простого порядка q.

Приведенную выше схему можно назвать *эфемерной* схемой Диффи-Хеллмана в отличие от *статической*, где ключи обоих абонентов фиксированы, как и общий ключ (т.о. выступают в качестве долговременных ключей).

Требование к случайности выбора секретных ключей в эфемерной схеме является принципиальным для обеспечения ее стойкости.

В рамках предлагаемого решения имеет смысл рассматривать прежде всего *схему*

Диффи-Хеллмана с кофактором, реализуемую в подгруппе простого порядка q группы точек эллиптической кривой над конечным простым полем характеристики p, заданной в одном из следующих представлений:

• (краткая) форма Вейерштрасса:

$$E_{W,a,b}(GF(p)) = \{(x, y) : x^2 \equiv x^3 + ax + b (\mathrm{mod}\ p)\}, \quad (1)$$

• (скрученная) форма Эдвардса [3]:

$$\overline{E}_{\mathrm{Edw},a,d}(GF(p)) = \{(x, y) : ax^2 + y^2 \equiv 1 + dx^2y^2 (\mathrm{mod}\ p)\}. \quad (2)$$

Обозначим $\pi(Q)$, $\pi : E(GF(p)) \to GF(p)$ – функцию, сопоставляющую точке Q на эллиптической кривой $E(GF(p))$ одну ее координату, а именно x-координату в (краткой) форме Вейерштрасса (1) или x-координату в (скрученной) форме Эдвардса (2).

Международный стандарт[2] не рекомендует непосредственное использование общего ключа, выработанного в результате выполнения схемы Диффи-Хеллмана (т.е. элемента какой-либо конечной группы), в качестве ключа для симметричного алгоритма шифрования. Вместо этого следует использовать значение, выработанное одной из *функций вычисления производного ключа* (KDF, key derivation function). Не конкретизируя данную схему, будем обозначать ее

$$\mathrm{kdf} : V^m \times V^*[\times V^l] \to V^k \times V^k;$$

где m – длина хэш-кода, S – контекст сеанса (идентификаторы, сертификаты, метки времени и пр., представленные в виде конкатенации строки октетов), l – (максимальная) длина опционально используемого и заранее распределенного между абонентами секретного ключа K, k – длина ключа блочного шифра и схемы выработки кода аутентификации сообщения.

Запись $K \,\|\, M = \mathrm{kdf}(\cdot)$ означает, что старшие k бит $2k$-битового значения kdf используются непосредственно как выработанный общий ключ, а младшие k бит – как вспомогательный ключ M.

3. Схема протокола «Лимонник-3»

Протокол «Лимонник-3» развивает классическую схему MTI/A0 из работы [4] с применением предложенных в [5] модификаций, обеспечивающих доказуемую стойкость, а также с введением возможности использования двух различных эллиптических кривых согласно [6] (независимо подобная идея предлагалась в [7]).

Используем две (возможно, одинаковые) эллиптические кривые $E_A(GF(p_A)), E_B(GF(p_B)$ и соответствующие подгруппы: $\langle P_A \rangle \subseteq E_A(GF(p_A)$, $\langle P_B \rangle \subseteq E_B(GF(p_B)$, при этом кофакторы $c_A = \#E_A(GF(p_A)) / \#\langle P_A \rangle$, $c_B = \#E_B(GF(p_B)) / \#\langle P_B \rangle$ могут быть нетривиальными.

Долговременными ключами этого протокола являются ключи схемы Диффи-Хеллмана.

[2] ISO/IEC 11770-3. Information technology. Security techniques. Key management. Part 3: Mechanisms using asymmetric techniques. — International Standard. — 2008. — Second edition.

Абонент A строит долговременную ключевую пару по правилу $S_A = s_A P_A$, а сеансовую ключевую пару – по правилу $K_A = k_A P_B$. Действия абонента B симметричны: $S_B = s_B P_B$, $K_B = k_B P_A$.

Протокол Л – 3

$A \to B$	$\mathrm{Id}_A, \mathrm{Cert}_A, k_A P_B$
B:	$Q = c_A k_B S_A, R = c_B s_B k_A P_B, W = H(\pi(Q), \pi(R))$
	$KPM = \mathrm{kdf}_H(W, \mathrm{Id}_A \mathrm{PId}_B \mathrm{POI[PK]})$
	$tag_B = \mathrm{mac}_M(h_2, k_B P_A, k_A P_B, \mathrm{Id}_B, \mathrm{Id}_A)$
$B \to A$	$\mathrm{Id}_B, \mathrm{Cert}_B, k_B P_A, tag_B$
A:	$Q = c_A s_A k_B P_A, R = c_B k_A S_B, W = H(\pi(Q), \pi(R))$
	$KPM = \mathrm{kdf}_H(W, \mathrm{Id}_A \mathrm{PId}_B \mathrm{POI[PK]})$
	Если $tag_B \neq \mathrm{mac}_M(h_2, k_B P_A, k_A P_B, \mathrm{Id}_B, \mathrm{Id}_A)$,
	разрывает сеанс
	$tag_A = \mathrm{mac}_M(h_3, k_A P_B, k_B P_A, \mathrm{Id}_A, \mathrm{Id}_B)$
$A \to B$	tag_A
B:	*Если* $tag_A \neq \mathrm{mac}_M(h_3, k_A P_B, k_B P_A, \mathrm{Id}_A, \mathrm{Id}_B)$,
	разрывает сеанс

4. Строительные блоки схем

В качестве базовых примитивов – «строительных блоков» – при реализации схемы предполагается использовать стандартизированные либо предполагающиеся к стандартизации в Российской Федерации решения и основанные на них схемы:

• схема Диффи-Хеллмана реализуется на эллиптических кривых, соответствующих требованиям ГОСТ Р 34.10-2012 для 512-битового секретного ключа, см., например, [8];

• функция хэширования – алгоритм «Стрибог» с длиной хэш-кода 512 битов;

• схема шифрования – алгоритм блочного шифрования «Кузнечик» с длиной блока 128 битов;

• код аутентификации сообщения – алгоритм блочного шифрования «Кузнечик» в режиме выработки имитовставки;

• функция выработки производного ключа – схема на основе TLS-PRF с использованием функции хэширования «Стрибог» либо схема, определенная национальными рекомендациями по стандартизации.

5. Обоснование стойкости

При обосновании стойкости протокола будем считать, что все используемые криптографические примитивы (схема цифровой подписи, хэш-функция, функция выработки производного ключа, алгоритм шифрования и т.д.) являются стойкими. Ряд атак, специфичных для конкретной реализации протокола (например, временные атаки или атаки, основанные на анализе формата сообщений) мы вынуждены оставить за рамками рассмотрения.

5. Задача восстановления общего ключа

Очевидно, что задача восстановления общего ключа в протоколе «Лимонник-3» может быть легко решена противником, способным эффективно решать *задачу Диффи-Хеллмана* или вычислять *дискретный логарифм* в группах точек эллиптических кривых.

Определение 1. *Вычислительная задача Диффи-Хеллмана (CDH), или просто задача Диффи-Хеллмана, в группе* $G = \langle P \rangle$ *состоит в определении по элементам* $X = aP, Y = bP$ *элемента* $(ab)P \in G$. *Определим также функцию* $CDH(P, aP, bP) = (ab)P$.

Определение 2. *Задача дискретного логарифмирования (DLP) в группе* $G = \langle P \rangle$ *состоит в определении по элементу* $Y \in G$ *такого* y, *что* $Y = yP$.

Отметим, что задача Диффи-Хеллмана с кофактором, очевидно, эквивалентна задаче Диффи-Хеллмана, поэтому в дальнейшем будем говорить только о последней. Далее, поскольку на настоящий момент не известны общие методы, позволяющие решать задачу Диффи-Хеллмана иначе как сведением ее к задаче дискретного логарифмирования, при анализе практической стойкости ограничимся рассмотрением только последней задачей.

Наилучшим методом для решения задачи дискретного логарифмирования является параллельный метод поиска коллизий [9, 10]. Трудоемкость указанного метода оценивается в [10] как

$$T = \sqrt{\frac{\pi q}{2m}} \tag{3}$$

операций вычисления итерационной функции метода (примерно соответствующей по трудоемкости вычислению суммы двух точек на эллиптической кривой), где m – мощность группы эффективно вычислимых автоморфизмов. Для кривых, соответствующих требованиям ГОСТ Р 34.10-2012, считаем $m = 2$.

Таким образом, трудоемкость вычисления общего ключа для задачи в подгруппе мощности q, где $q \approx 2^n$, группы точек эллиптической кривой, соответствующей требованиям ГОСТ Р 34.10-2012, обеспечивает битовую стойкость $n/2$. Например, при использовании эллиптических кривых, удовлетворяющих требованиям ГОСТ Р 34.10-2012 при 512-битовом секретном ключе, битовая стойкость соответствует длине ключа блочного шифра “Кузнечик”.

6. Определение долговременных ключей

В протоколе «Лимонник-3» долговременные открытые и секретные ключи абонентов связаны между собой тем же соотношением $P_A = s_A P$, что и в схеме цифровой подписи в соотвествии с ГОСТ Р 34.10-2012, а восстановление секретного ключа по открытому ключу требует решения задачи дискретного логарифмирования.

Для определения долговременных секретных ключей при компрометации общего ключа K протокола «Лимонник-3» противнику предстоит решить следующие задачи:

1) обратить функцию выработки производного ключа;
2) обратить хэш-функцию;
3) решить задачу дискретного логарифмирования.

В сделанных предположениях о стойкости используемых примитивов данные задачи являются вычислительно сложными.

Навязывание ключа

Существенная зависимость общего ключа K, выработанного в каждом сеансе, от параметров сеанса обеспечивается вычислением K как функции от посылок абонентов K_A, K_B, сформированных псевдослучайным образом, а также от их идентификаторов и контекста протокола.

Рассмотрим возможность навязывания заданной общей ключевой информации Q', отличной от $Q = ck_Ak_BP$, одним из абонентов. Абонент A имеет единственную возможность навязать Q', выбрав $k_A \equiv 0(\mathrm{mod}\, q)$, но такая возможность исключается по правилам построения ключей и должна отслеживаться при выполнении протокола. Абонент B для навязывания заданного $Q' = k_{A'}P$, где значение $k_{A'}$ ему известно, должен подобрать свой сеансовый ключ $k_{B'}$ так, что $k_{B'}k_AP = k_{A'}P$, т.е. $k_{B'} \equiv k_A / k_{A'}(\mathrm{mod}\, q)$, для чего необходимо вычислить дискретный логарифм в $\langle P \rangle$.

При разработке реализации схемы в поле OI возможно добавить одноразовые случайные значения, метки времени и другую информацию, привязанную к конкретному сеансу.

Заметим, что стойкость конкретной реализации протокола будет обусловлена характеристиками используемого датчика случайных или псевдослучайных чисел.

Аутентификация инициатора и ответчика

Аутентификация обеспечивается привязкой долговременного открытого ключа к абоненту, подтверждаемой *сертификатом*. Таким образом, стойкость аутентификации обеспечивается стойкостью инфраструктуры открытых ключей (ИОК).

UKS-атаки

Атака класса unknown key-share (UKS, source-substitution) состоит в том, что абоненты вырабатывают общий ключ, но один из них считает его общим с третьим абонентом, навязанным противником. При этом противник не обязан знать общий ключ.

В [11–13] описаны известные методы противодействия UKS-атакам:

• проверка знания абонентом секретного ключа, соответствующего данному открытому ключу, при его сертификации (например, запрос на регистрацию сертификата ключа схемы цифровой подписи может быть подписан этим же ключом);

• подтверждение ключа, реализованное таким образом, что ключ связан с участниками сеанса;

• использование функций выработки производного ключа, зависящих от идентификаторов абонентов.

В работе предложено использование единообразных схем подтверждения ключа и выработки общего ключа, которые обеспечивают реализацию последних двух методов. Требование проверки знания секретного ключа является логичным и, как правило, выполняется, однако обеспечить его в рамках произвольной сети связи вряд ли представляется возможным.

KCI-атаки

KCI-атака (key-compromise impersonation) состоит в том, что противник, завладевший долговременным ключом абонента, пытается выдать себя перед ним за третье лицо. Легко показать, что протокол «Лимонник-3» стоек относительно KCI-атак, как и прототип – протокол MTI/A0 [4]. Действительно, если противник определил долговременный секретный ключ s_A и пытается выдать себя перед A за B, то ему необходимо определить ключевую информацию $Q = c_A s_A k_B P_A = c_A k_B S_A, R = c_B k_A S_B = c_B s_B k_A P_B$, но определить R без знания k_A либо s_B не сможет.

Подтверждение ключа

Протокол «Лимонник-3» используют для подтверждения ключа код аутентификации сообщения, где формат сообщения включает идентификаторы абонентов. Код аутентификации вырабатывается на уникальном для сеанса общем ключе M, отличающемся от общего ключа K. Сложность задачи ложного подтверждения ключа для противника определяется сложностью задачи подделки кода аутентификации. Вероятность того, что противник сможет повторно использовать переданный в одном из сеансов код аутентификации, определяется вероятностью совпадения ролей, идентификаторов и сеансовых открытых ключей абонентов. В предположении о том, что хотя бы один из абонентов использует надежный датчик ПСЧ, а функция выработки ключа удовлетворяет некоторым разумным криптографическим требованиям (однонаправленность; неотличимость выходных данных от случайной строки из $V^k \times V^k$; существенная зависимость выхода от контекста сеанса; независимость двух «половинок2 выхода друг от друга, т.е. невозможность извлечь при компрометации K информацию о M и наоборот), вероятность этого события можно считать ничтожно малой.

Чтение назад

Очевидно, что протокол «Лимонник-3» не может обеспечить защиту от чтения назад при одновременной компрометации долговременных секретных ключей обоих абонентов. Действительно, для восстановления общего ключа любого сеанса противнику достаточно вычислить $c_A s_A (k_B P_A), c_B s_B (k_A P_B)$.

7. Формальное обоснование стойкости

Рассмотрим формальное доказательство стойкости для протокола, идентичного «Лимонник-3», но использующего в качестве функции выработки производного ключа хэш-функцию, в модифицированной модели Канетти-Кравчика [7, 14]. Приведем разработанную модель.

Введем следующие определения.

Определение 3. *Функция* $p(\lambda), p: \mathrm{R} \to \mathrm{R}$ *называется пренебрежимо малой, если для любого многочлена* $p \in \mathrm{Z}[x]$, $\forall \varepsilon \in \mathrm{R}_+$ $\exists N$ *в т.ч.* $\forall \lambda > N$ $| p(\lambda) | < \varepsilon$.

Определение 4. *Распознавательная задача Диффи-Хеллмана (DDH) в группе* $G = \langle P \rangle$ *состоит в вычислении для тройки* $X = aP, Y = bP, Z$ *функции* $DDH(P, X, Y, Z)$, *которая принимает значение 1, если* $Z = (ab)P$ *или 0 в противном случае.*

Определение 5. *Интервальная задача Диффи-Хеллмана (Gap Diffie-Hellman Problem,*

GDH) в группе $G = \langle P \rangle$ *состоит в решении вычислительной задачи Диффи-Хеллмана при наличии у решающего алгоритма доступа к оракулу для распознавательной задачи Диффи-Хеллмана.*

В эксперименте принимают участие следующие стороны:

• несколько (два или более) честных абонентов;

• удостоверяющий центр (УЦ) – особенный абонент, регистрирующий сертификаты открытых ключей абонентов. Считаем, что каждый абонент может иметь единственный сертифицированный открытый ключ;

• противник – особенный абонент, полностью контролирующий коммуникационную среду.

Пусть λ - параметр стойкости.

Будем обозначать каждый сеанс протокола $(A, B, role, T)$, где A – выполняющий протокол абонент, $role \in \{инициатор, ответчик\}$ – роль абонента, T – транскрипция протокола, т.е. все переданные сообщения. Считаем, что протокол выполняется одним абонентом A, поскольку вся коммуникационная среда контролируется противником, и абонент A не имеет возможности установить, кто фактически является вторым участником сеанса. Для сеанса $(A, B, role)$, где $role = инициатор$, сеанс $(B, A, ответчик)$, назовем *дополняющим*, и наоборот.

Определение 6. *Выберем случайное значение* $b \in_R \{0,1\}$*, и определим тестовый сеанс* $(A, B, role, T; K_C)$*, где*

$$K_C := \begin{cases} \text{общий ключ сеанса (A,B,role,T)}, & \text{если } b = 0, \\ \text{случайная строка длины } \lambda, & \text{если } b = 1. \end{cases}$$

Предположим, что протокол является стойким относительно *угрозы извлечения информации об общем ключе (AKE-стойким)*, если не существует такого полиномиального вероятностного противника M, что

$$Pr[\text{M}(A, B, role, T; K_C) = b] - \frac{1}{2} = p(\lambda),$$

где функция $p(\lambda)$ является пренебрежимо малой.

При этом угроза состоит в том, что противник сможет различать случайную строку и общий ключ, выработанный в тестовом сеансе, так, что вероятность его успеха (как функция от параметра стойкости) не является пренебрежимо малой.

Для реализации определенной выше угрозы противник имеет возможность перехватывать, изменять и задерживать все сообщения всех абонентов, кроме УЦ. Также противник может захватывать контроль над участниками, т.е. выполнять шаги протокола от их имени, регистрировать от их имени открытые ключи (в частности, тем самым реализовывая KCI- и UKS-атаки), определять секретные ключи участников и общие ключи завершенных сеансов (за исключением тестового сеанса, т.е. того, против которого направлена атака, и ключей его участников).

Рассмотрим вариант протокола без подтверждения ключа:

Протокол Л – 2

$A \to B$ $\quad \mathrm{Cert}_A, k_A P_B$

B: $\quad Q = c_A k_B S_A, R = c_B s_B k_A P_B, W = H(\pi(Q), \pi(R))$

B: $\quad K = \mathrm{kdf}_H((W, \mathrm{Id}_A \,\mathrm{P}\mathrm{Id}_B \,\mathrm{POI})$

$B \to A$ $\quad \mathrm{Cert}_B, K_B$

A: $\quad Q = c_A s_A k_B P_A, R = c_B k_A S_B, W = H(\pi(Q), \pi(R))$

$\quad K\,\mathrm{P}M = \mathrm{kdf}_H(W, \mathrm{Id}_A \,\mathrm{P}\mathrm{Id}_B \,\mathrm{POI})$

Теорема 1. *Если задача GDH для семейства групп Λ является сложной, а функция выработки производного ключа моделируется при помощи случайного оракула, то протокол Л-2 является стойким относительно угрозы извлечения информации об общем ключе.*

Доказательство. Доказательство следует из методологии [5], поскольку протокол Л-2 обладает определенным сходством с протоколом KEA+ из [5], за исключением возможности использования различных эллиптических кривых.

Определим *подпись* сеанса протокола

$$\Omega = (s_A k_B P_A, s_B k_A P_B, \mathrm{Id}_A, \mathrm{Id}_B).$$

При этом общий ключ

$$H(CDH(P_A, S_A, K_B), CDH(P_B, S_B, K_A), \mathrm{Id}_A, \mathrm{Id}_B)_{\lambda, 2\lambda - 1}.$$

Поскольку H моделируется случайной функцией с пренебрежимо малой вероятностью коллизии, противник имеет следующие возможности определения общего ключа:

1. вызвать алгоритм $H(CDH(P_A, S_A, K_B), CDH(P_B, S_B, K_A), \mathrm{Id}_A, \mathrm{Id}_B)$, т.е. решить задачу дискретного логарифимирования;
2. инициировать сеанс с той же подписью и определить его ключ.

Отбрасываем вторую возможность, поскольку сеансы с одинаковыми подписями включают идентификаторы сторон, поэтому очевидно, что их сеансовые ключи идентичны, и сеансы идентичны, за исключением пренебрежимо малой доли, определяемой вероятностью коллизии H.

Теперь достаточно показать, что если полиномиальный противник M способен отличить выработанный ключ от случайной последовательности с вероятностью, не являющейся пренебрежимо малой, то можно построить полиномиальный алгоритм S для решения задачи GDH с вероятностью, не являющейся пренебрежимо малой.

Итак, пусть в алгоритме S на вход поступают $(X, Y) \in G^2$, $G = \langle P \rangle$, $(G, P) \in \Lambda$.

Предположим, что S имеет доступ к оракулу, решающему задачу DDH. В алгоритме запускается эксперимент с участием n честных абонентов, где каждый из них участвует в не более чем k сеансов, и противник M. В алгоритме S случайно выбирается абонент A, устанавливаются его групповые параметры $G_A := \langle P \rangle, P_A := P$ и долговременный ключ $S_A := X$. Остальные абоненты устанавливают свои групповые параметры соответственно протоколу, случайно выбирая их из Λ. S выбирает случайное значение $u \in_R [1, \ldots, nk]$ и устанавливает счетчик $i := 1$.

В ходе эксперимента S обрабатывает запросы M следующим образом.

1. Обращение к оракулу $H(v)$ обрабатывается определенной ниже функцией $H_1(v)$.

2. Запрос на инициализацию сеанса $(B,C,role)$, где B,C – два абонента, отличные от A, обрабатываются согласно протоколу Л-2, за исключением вычисления H, где используется H_1.

3. Запрос на инициализацию сеанса $(A,C,role)$ не может обрабатываться согласно протоколу Л-2, так как S не может определить s_A.

Таким образом, в зависимости от роли A', $role$, алгоритм S выполняет следующее. Если A – инициатор, то он выбирает $x \in_R Z$, отправляет C элемент xP_C и, получив ответ ω, вычисляет общий ключ $H_2(1,\omega,xP_C,\mathrm{Id}_A,\mathrm{Id}_C)$, где функция $H_2(i,\omega,k_AP_C,\mathrm{Id}_A,\mathrm{Id}_C)$ определена ниже. Если A – ответчик, S ждет запроса α, выбирает $k_A \in_R Z$ и вычисляет общий ключ $H_2(2,\alpha,k_AP_C,\mathrm{Id}_C,\mathrm{Id}_A)$.

4. При обработке запроса на инициализацию сеанса $(B,A,role)$ с абонентом B S проверяет счетчик i. Если $i=u$, то сеанс объявляется «особым», абонент B отправляет Y – второй параметр нашей задачи CDH – в качестве своего открытого сеансового ключа и не вычисляет общий ключ. Если $i \neq u$, то S увеличивает значение счетчика $i := i+1$ и продолжает, как в случае 2.

5. Обрабатывая запрос на определение общего ключа или на определение сеансового секретного ключа из какого-либо сеанса, отличного от особого, S передает M соответствующее значение, которое уже вычислено при обработке запросов типа 2, 3 или 4.

Для особого сеанса S останавливает эксперимент.

6. Обрабатывая запрос на установление контроля над C, где C – отличный от A и B абонент, S передает M значение долговременного сеансового ключа s_C, а также значения всех общих ключей, выработанных с участием C в ходе эксперимента. При попытке установления контроля над A или B S возвращает отрицательный результат (ошибку).

Когда противник M завершает работу, S просматривает все сделанные M запросы к оракулу $H()$ и проверяет, присутствует ли среди них значение $CDH(X,Y)$. Обнаружив такое значение (при помощи вызова оракула DDH), S возвращает его в качестве решения задачи GDH. Если этого значения не найдено, то S завершается с отрицательным результатом (ошибкой).

Определим функции H_1 и H_2.

Функция $H_1(Z_1,Z_2,B,C)$ симулирует H относительно корректных подписей Л-2:

• если функция H_1 для заданного входа уже определена, то вернуть это значение;

• если функция H_1 не определена, то просмотреть транскрипцию предыдущих вызовов $H_2(\cdot)$ и для каждого вызова $H_2(i,\omega,Z,B',C')=v$ проверить, что

$$B=B', C=C', Z=Z_{3-i} \text{ и } DDH(X,\omega,Z_i)=1.$$

Если все условия выполнены, то вернуть v.

• если такой e найден, выбрать случайное значение w, $w \in V^{\lambda}$, запомнить, что

$H_1(Z_1, Z_2, B, C) := w$, и вернуть w.

Функция $H_2(i,\ldots)$ реализует действие оракула над подписями, неизвестными S: по условиям эксперимента он принимает такие входные параметры (Z_1, Z_2, B, C), что $Z_i = CDH(X, \omega)$ и $Z_{3-i} = Z$.

• Если H_2 для заданного аргумента уже определено, то вернуть его;

• Если нет, то просмотреть транскрипции предыдущих вызовов $H_1(\cdot)$ и для каждого вызова $H_1(Z_1, Z_2, B', C') = v$ проверить, что

$$B = B', C = C', Z = Z_{3-i} \text{ и } DDH(X, \omega, Z_i) = 1.$$

Если все условия выполнены, вернуть v.

• Если такого v не найдено, выбрать случайное значение $w \in V^{\lambda}$, запомнить, что $H_2(i, \omega, Z, B, C) := w$ и вернуть w.

Покажем, что транскрипция симулированного эксперимента распределена так же, как и транскрипция эксперимента, выполненного $\mathcal{M}$, за исключением пренебрежимо малой доли значений.

Действительно, транскрипция $\mathcal{M}$ состоит из всех долговременных открытых ключей, долговременных секретных ключей некоторых (контролируемых $\mathcal{M}$) участников, сеансовых ключей и выработанных общих ключей контролируемых участников, а также ответов оракула.

Просто проверить, что все секретные и соответствующие открытые ключи, кроме неизвестных $\mathcal{M}$ s_A, распределены идентично сессии $\mathcal{M}$. Вероятность того, что $\mathcal{M}$ выберет сеанс, выполняемый с участником A, в качестве соответствующего сеанса, не менее $1/n^2t$.

В этом случае $\mathcal{M}$ определяет общий ключ, предоставляя S подпись тестового сеанса, содержащую $CDH(X, Y)$. Итак, S действительно решает вычислительную задачу Диффи-Хеллмана.

Время работы S определяется выполнением алгоритма $\mathcal{M}$ (что требует полиномиального (от λ) времени t) и обработкой $O(t^2)$ запросов к оракулам, таким образом, общее время работы $\mathcal{M}$ также полиномиально.

Таким образом разработан полиномиальный алгоритм решения задачи GDH с вероятностью

$$\frac{1}{n^2 k} Pr[Success(\mathcal{M})],$$

где $Pr[Success(\mathcal{M})]$ – вероятность того, что $\mathcal{M}$ нарушает AKE-стойкость протокола Л-2.

Замечание 1. *Используя Теорему 1, можно показать, что протокол «Лимонник-3» является стойким в модели Universally Composable [14]. То есть если задача GDH для семейства* Λ *является сложной, функция* mac *стойкая относительно подделки, а хэш-функция моделируется случайным оракулом, то протокол «Лимонник-3» является AKE-стойким.*

8. Автоматизированная верификация

Автоматизированная верификация свойств схемы «Лимонник-3» выполнена

А.М. Семеновым в [15] с использованием ряда средств автоматизированной верификации, таких как AVISPA, ProVerif, Scyther. Показано, что протокол удовлетворяет требованиям стойкости, эксплуатируемых уязвимостей не обнаружено.

9. Практическая эффективность

Эксплуатационные характеристики схемы «Лимонник-3» при необходимой стойкости укладываются в требования, заданные в [6]. Схема требует 3 пересылки (это минимальное количество пересылок, при котором возможно подтверждение ключа), использует стандартизированные криптографические примитивы, превосходит по эффективности, измеренной в количестве операций вычисления кратной точки в группе точек эллиптической кривой, схемы типа STS, так как не требует проверки цифровых подписей, и соответствует по эффективности схемам типа MQV. Возможность использования абонентами различных эллиптических кривых повышает гибкость схемы. Предусмотренная возможность использования классов эллиптических кривых, обладающих эффективными законами сложения (например, скрученных кривых Эдвардса, описанных в рекомендациях по стандартизации Р 50.1.114-2016[3], и в [8]), позволит дополнительно повысить эффективность конкретных реализаций, выраженную в количестве элементарных операций процессора [3].

10. Перспективы

В настоящее время ведутся работы по внедрению разработанных автором протоколов, входящих в рекомендации по стандартизации Р 1323565.1.004-2017, в том числе «Лимонник-3», в протокол TLS 1.3 [16]. Исследуются возможности создания на основе протокола «Лимонник-3» постквантовой версии с заменой примитивов Диффи-Хеллмана на примитивы, основанные на аппарате изогений суперсингулярных эллиптических кривых, такие как SIDH, описанный в [17]. Возможности создания подобных протоколов показаны, например, в [18]. На настоящий момент для протоколов выработки общего ключа, построенных на основе аппарата изогений суперсингулярных эллиптических кривых, наилучший квантовый алгоритм анализа [19] имеет трудоемкость $O(p^{1/3})$, если суперсингулярные эллиптические кривые строятся над полем $GF(p^2)$. При этом квантовый алгоритм решения задачи дискретного логарифмирования на эллиптической кривой имеет полиномиальную трудоемкость, так что протокол «Лимонник-3» не является стойким относительно квантового криптоанализа.

Заключение

В настоящей работе приведено обоснование криптографических характеристик стандартизированной схемы выработки общего ключа «Лимонник-3», в том числе получено доказательство утверждения, обосновывающего стойкость протокола в формальной модели. Приводятся результаты исследований, показывающие, что «Лимонник-3» при реализации его с рекомендуемыми в настоящей статье параметрами и криптографическими примитивами по своим криптографическим качествам удовлетворяет требованиям, предъявляемым к современным криптографическим протоколам.

[3] Рекомендации по стандартизации Р 50.1.114-2016. Информационная технология. Криптографическая защита информации. Параметры эллиптических кривых для криптографических алгоритмов и протоколов.

СПИСОК ЛИТЕРАТУРЫ:

1. Diffie W., Hellman M. New directions in cryptography//IEEE Trans. Inform. Theory. 1976. V. IT-22. No 6. P. 644–654. URL: https://ee.stanford.edu/~hellman/publications/24.pdf (дата обращения: 10.02.2019).
2. Bader C., Hofheinz D., Jager T., Kiltz E., Li Y. Tightly-secure authenticated key exchange// Theory of Cryptography Conference. – LNCS 9014. – P. 629–658. – 2015. URL: https://link.springer.com/chapter/10.1007/978-3-662-46494-6_26 (дата обращения: 10.02.2019).
3. Dygin D., Grebnev S. Efficient implementation of the GOST R 34.10 digital signature scheme using modern approaches to elliptic curve scalar multiplication. Математические вопросы криптографии. 2013. Том 4. № 2. С. 47–58. URL: http://mi.mathnet.ru/mvk82 (дата обращения: 10.02.2019).
4. Matsumoto T., Takashima Y., Imai H. On seeking smart public-key distribution systems. Trans. IECE of Japan. February 1986. E69(2). P. 99–106.
5. Lauter K., Mityagin A. Security analysis of KEA authenticated key exchange protocol. In PKC 2006, volume 3958 of LNCS. 2006. 378–394. URL: https://link.springer.com/chapter/10.1007/11745853_25 (дата обращения: 10.02.2019).
6. Матюхин Д.В. О некоторых свойствах схем выработки общего ключа, использующих инфраструктуру открытых ключей, в контексте разработки стандартизированных криптографических решений// Обозрение прикладной и промышленной математики. 8. 2011. P. 793–794 URL: https://tvp.ru/conferen/vsppm12/kazad038.pdf. (дата обращения: 10.02.2019).
7. Chatterjee S., Menezes A., Ustaoglu B. A generic variant of NIST's KAS2 key agreement scheme. ACISP. 2011. P. 353–370. https://link.springer.com/chapter/10.1007/978-3-642-22497-3_23.
8. Alekseev E.K., Nikolaev V.D., Smyshlyaev S.V. On the security properties of Russian (дата обращения: 10.02.2019).standardized elliptic curves. Математические вопросы криптографии. 2018. 9. № 3. P. 5–32. URL: http://mi.mathnet.ru/mvk260 (дата обращения: 10.02.2019).
9. Oorschot van P. C., Wiener M. J. Parallel collision search with cryptanalytic applications. J. Cryptology. 1999. 12. 1. P. 1–28. URL: http://cr.yp.to/bib/1999/vanoorschot.pdf (дата обращения: 10.02.2019).
10. Bos J. W., Costello C., Miele A. Elliptic and hyperelliptic curves: A practical security analysis// Krawczyk Hugo, editor, Proc. PKC'2014 LNCS 8383, Springer. 2014. P. 203–220. URL: https://link.springer.com/chapter/10.1007/978-3-642-54631-0_12 (дата обращения: 10.02.2019).
11. Baek J., Kim K. Remarks on the unknown key-share attacks. IEICE Trans. 2000. E83-A. 12.
12. Blake-Wilson S., Menezes A. Authenticated Diffie-Hellman key-exchange protocols. Proc. 5th Workshop on Selected Areas in Cryptography. LNCS 1556. 1999. P. 339–361.
13. Blake-Wilson S., Menezes A. Unknown key-share attacks on the station-to-station (STS) protocol. Public Key Cryptography – PKC'1999 LNCS 1560. 1999. P. 156–170. URL: https://link.springer.com/chapter/10.1007%2F3-540-49162-7_12 (дата обращения: 10.02.2019).
14. Canetti R., Krawczyk H. Analysis of key-exchange protocols and their use for building secure channels. EUROCRYPT 2001, LNCS 2045. N. Y.: Springer-Verlag. 2001. P. 453–474. URL: https://www.iacr.org/archive/eurocrypt2001/20450451.pdf (дата обращения: 10.02.2019).
15. Semyonov A.M. Analysis of Russian key-agreement protocols using automated verification tools. Математические вопросы криптографии. 2017. 8. № 2. P. 131–142. URL: https://mi.mathnet.ru/mvk229 (дата обращения: 10.02.2019).
16. Гребнев С.В., Лазарева Е.В., Лебедев П.А., Нестеренко А.Ю., Семенов А.М. Интеграция отечественных протоколов выработки общего ключа в протокол TLS 1.3. ПДМ. Приложение, 2018, № 11. С. 62–65. URL: http://mi.mathnet.ru/pdma382 (дата обращения: 10.02.2019).
17. De Feo L., Jao D., Plût J. Towards Quantum-Resistant Cryptosystems From Supersingular Elliptic Curve Isogenies., J. Mathematical Cryptology, 8(3) (2014), P. 209–247. URL: https://ia.cr/2011/506. (дата обращения: 10.02.2019).
18. Galbraith S. Authenticated key exchange for SIDH, Cryptology ePrint Archive: Report 2018/266, 2018. URL: https://ia.cr/2018/266 (дата обращения: 10.02.2019).
19. Biasse J.-F., Jao D., Sankar A. A quantum algorithm for computing isogenies between supersingular elliptic curves. Progress in Cryptology – INDOCRYPT 2014, LNCS 8885. 2015. P. 428–442. URL: https://link.springer.com/chapter/10.1007/978-3-319-13039-2_25 (дата обращения: 10.02.2019).

REFERENCES:

[1] Diffie W., Hellman M. New directions in cryptography//IEEE Trans. Inform. Theory. 1976. V. IT-22. No 6. P. 644–654. URL: https://ee.stanford.edu/~hellman/publications/24.pdf (accessed: 10.02.2019).

[2] Bader C., Hofheinz D., Jager T., Kiltz E., Li Y. Tightly-secure authenticated key exchange. Theory of Cryptography Conference. – LNCS 9014. – P. 629–658. – 2015. URL: https://link.springer.com/chapter/10.1007/978-3-662-46494-6_26 (accessed: 10.02.2019).

[3] Dygin D., Grebnev S. Efficient implementation of the GOST R 34.10 digital signature scheme using modern approaches to elliptic curve scalar multiplication. Mathematical problems of cryptography. 2013. 4. № 2. P. 47–58. URL: http://mi.mathnet.ru/mvk82 (accessed: 10.02.2019).

[4] Matsumoto T., Takashima Y., Imai H. On seeking smart public-key distribution systems. Trans. IECE of Japan. February 1986. E69(2). P. 99–106.

[5] Lauter K., Mityagin A. Security analysis of KEA authenticated key exchange protocol. In PKC 2006, volume 3958 of LNCS. 2006. 378–394. URL: https://link.springer.com/chapter/10.1007/11745853_25 (accessed: 10.02.2019).

[6] Matyukhin D.V. On some properties of the key exchange schemes, using PKI, in the context of standardization (in Russian). Obozrenie prikladnoy i promyshlennoi matematiki. 18. 2011. P. 793–794. URL: http://tvp.ru/conferen/vsppm12/kazad038.pdf (accessed: 10.02.2019).

[7] Chatterjee S., Menezes A., Ustaoglu B. A generic variant of NIST's KAS2 key agreement scheme. ACISP. 2011. P. 353–370. URL: https://link.springer.com/chapter/10.1007/978-3-642-22497-3_23. (accessed: 10.02.2019).

[8] Alekseev E.K., Nikolaev V.D., Smyshlyaev S.V. On the security properties of Russian standardized elliptic curves. Mat. Vopr. Crypt. 2018. 9. № 3. P. 5–32. URL: http://mi.mathnet.ru/mvk260 (accessed: 10.02.2019).

[9] Oorschot van P. C., Wiener M. J. Parallel collision search with cryptanalytic applications. Cryptology. 1999. 12. 1. P. 1–28. URL: http://cr.yp.to/bib/1999/vanoorschot.pdf (accessed: 10.02.2019).

[10] Bos J. W., Costello C., Miele A. Elliptic and hyperelliptic curves: A practical security analysis. Krawczyk Hugo, editor, Proc. PKC'2014 LNCS 8383, Springer. 2014. P. 203–220. URL: https://link.springer.com/chapter/10.1007/978-3-642-54631-0_12 (accessed: 10.02.2019).

[11] Baek J., Kim K. Remarks on the unknown key-share attacks. IEICE Trans. 2000. E83-A. 12.

[12] Blake-Wilson S., Menezes A. Authenticated Diffie-Hellman key-exchange protocols. Proc. 5th Workshop on Selected Areas in Cryptography LNCS 1556. 1999. P. 339–361 (accessed: 10.02.2019).

[13] Blake-Wilson S., Menezes A. Unknown key-share attacks on the station-to-station (STS) protocol. Public Key Cryptography – PKC'1999 LNCS 1560. 1999. P. 156–170. URL: https://link.springer.com/chapter/10.1007%2F3-540-49162-7_12 (accessed: 10.02.2019).

[14] Canetti R., Krawczyk H. Analysis of key-exchange protocols and their use for building secure channels. EUROCRYPT 2001, LNCS 2045. N. Y.: Springer-Verlag. 2001. P. 453–474. URL: https://www.iacr.org/archive/eurocrypt2001/20450451.pdf (accessed: 10.02.2019).

[15] Semyonov A.M. Analysis of Russian key-agreement protocols using automated verification tools. Mat. Vopr. Crypt. 2017. 8. № 2. P. 131–142. URL: http://mi.mathnet.ru/mvk229 (accessed: 10.02.2019).

[16] Grebnev S.V., Lazareva E.V., Lebedev P.A., Nesterenko A.Yu., Semyonov A.M. Integration of russian key-agreement protocols into the TLS 1.3. PDM. 2018, № 11. P. 62–65. URL: http://mi.mathnet.ru/pdma382 (accessed: 10.02.2019).

[17] De Feo L., Jao D., Plût J. Towards Quantum-Resistant Cryptosystems From Supersingular Elliptic Curve Isogenies., J. Mathematical Cryptology, 8(3) (2014), P. 209–247. URL: http://ia.cr/2011/506. (accessed: 10.02.2019).

[18] Galbraith S. Authenticated key exchange for SIDH, Cryptology ePrint Archive: Report 2018/266, 2018. URL: http://ia.cr/2018/266 (accessed: 10.02.2019).

[19] Biasse J.-F., Jao D., Sankar A. A quantum algorithm for computing isogenies between supersingular elliptic curves. Progress in Cryptology – INDOCRYPT 2014, LNCS 8885. 2015. P. 428–442. URL: https://link.springer.com/chapter/10.1007/978-3-319-13039-2_25 (accessed: 10.02.2019).

Поступила в редакцию – 12 февраля 2019 г. Окончательный вариант – 20 мая 2019 г.
Received – February 12, 2019. The final version – May 20, 2019.