

САМОСИНХРОНИЗИРУЮЩИЙСЯ ПОТОЧНЫЙ АЛГОРИТМ ЗАЩИТЫ ПЕРЕДАЧИ ГЕОДАННЫХ С БАЗОВОЙ ГРУППОЙ, ЯВЛЯЮЩЕЙСЯ АМАЛЬГАМИРОВАННЫМ СВОБОДНЫМ ПРОИЗВЕДЕНИЕМ

Лоссов К.И.

Московский государственный университет геодезии и картографии, 105064, Москва, Россия
e-mail: konsiv@gmail.com

Алгоритм защиты, представленный в данной работе, является самосинхронизирующимся поточным алгоритмом, и структурно схож с режимом OFB, предназначенным для защиты информационных потоков. Но, в отличие от последнего, использует другую платформу, а также иные генератор ключей и одностороннюю функцию. В качестве платформы выбираются свободные произведения групп с объединением, где объединяемые подгруппы некоммутативны, и не являются нормальными в множителях. Работа генератора ключей и односторонняя функция основаны на редукции в амальгамированных произведениях. Алгоритм предполагает, что индексы объединенной подгруппы в сомножителях должны быть не меньше исходного алфавита сообщений, при этом для его длительного бесперебойного функционирования сама подгруппа должна быть достаточно большой. Для валидации алгоритма описан пример такой схемы для конкретной свободной конструкции, в котором продемонстрирована работа генератора ключей и кодирование-декодирование передаваемого сообщения.

Ключевые слова: алгоритм защиты, поток гео данных, односторонняя функция, группа, объединенная подгруппа, редукция произведения, самосинхронизирующийся режим.

A SELF-SYNCHRONIZING STREAMING GEODATA SECURITY ALGORITHM WITH A CORE GROUP THAT IS AN AMALGAMATED FREE PRODUCT

Lossov K.I.

Moscow State University of Geodesy and Cartography, 105064, Moscow, Russia

The security algorithm presented in this paper is a self-synchronizing stream algorithm and is structurally similar to the OFB mode, designed to protect information flows. However, unlike the latter, it uses a different platform, as well as a different key generator and one-way function. The platform chosen is free products of groups with amalgamation, where the amalgamated subgroups are non-commutative and non-normal in their factors. The operation of the key generator and one-way function are based on reduction in amalgamated products. The algorithm assumes that the indices of the amalgamated subgroup in the factors must be no smaller than the original message alphabet, while for its long-term, uninterrupted operation, the subgroup itself must be sufficiently large. To validate the algorithm, an example of such a scheme for a specific free construction is described, demonstrating the operation of the key generator and the encoding and decoding of the transmitted message.

Keywords: protection algorithm, geodata stream, one-way function, group, amalgamated subgroup, product reduction, self-synchronizing mode.

Введение

В настоящее время наблюдается недооценка рисков при передаче конфиденциальных данных. Так, например, в исследовании [1] констатируется уязвимость геостационарных спутниковых систем при обеспечении связи, включая доступ к мобильной телефонии, коммерческому и правительственному сетевому трафику. Эти системы демонстрируют недостаточную степень защиты передаваемой конфиденциальной информации по открытым каналам связи. Было показано, что злоумышленник с ограниченными ресурсами, используя коммерчески доступное, недорогое оборудование, может надежно перехватывать и декодировать сотни каналов связи с одной точки обзора. Вместе с тем, предпринимаются усилия для защиты особо чувствительных информационных потоков. В работе [2] рассмотрена возможность использования криптографических алгоритмов многослойного шифрования информационных массивов, отображающих геопространственную обстановку местности. Таким образом,

явно наблюдаются импульсы, стимулирующие разработку новых, надежных алгоритмов защиты гео данных. В настоящей работе предложен самосинхронизирующийся поточный алгоритм защиты [3], основанный на редукции произведения в амальгамах групп, который ввиду блочно-поточной структуры удобен для обеспечения безопасной передачи кортежей пространственно-временных данных.

Самосинхронизирующиеся поточные криптосистемы представляют собой разновидность симметричных поточных шифров. В этих системах генерация ключевого потока осуществляется на основе исходного ключа и предыдущих N символов зашифрованного текста. Это позволяет каждому зашифрованному символу быть корректно расшифрованным при условии, что предшествующие символы были правильно получены. Поскольку предложенная в работе схема соединяет в себе элементы поточной и блочной защиты информации, то ее безопасность во многом зависит от качества генератора потока ключей, а также от силы алгоритма и режима работы.

Концепция блоков обеспечивает безопасность процесса и автоматическую синхронизацию между отправителем и получателем. Кроме того, с помощью блочного шифра можно проверить подлинность сообщения: второй абонент, получив сообщение, зашифровывает его на секретном ключе и сравнивает полученный результат с последним блоком шифротекста, который отправил первый абонент. Так получатель удостоверяется, что сообщение не было подделано на узле связи.

Распознавание блоков в поточном шифровании обеспечивают синхронные маркеры, которые вставляются в передаваемое сообщение, это специальные сигналы в шифротексте, которые позволяют приемнику найти местоположение данных и восстановить состояние генератора ключа после ошибки, повышая надежность потокового шифрования. При потере части шифротекста система использует эти маркеры для возобновления корректной расшифровки. Принимающая сторона может расшифровывать данные в асинхронном режиме, не требуя синхронизации генераторов ключей на передающей и принимающей сторонах. Основное преимущество такой модели шифрования заключается в том, что ошибки, возникающие при вставке, удалении или изменении нескольких символов в зашифрованном тексте, не распространяются на последующие блоки данных, т.е. получатель будет иметь возможность корректно расшифровывать последующие блоки сообщения.

В настоящее время на практике для защиты данных широко применяются методы алгебраической криптографии, в частности, шифрование, использующее в качестве платформы различные группы. Многочисленные примеры можно встретить в работах [4,5].

В представленной работе в качестве платформы алгоритма защиты используются свободные произведения групп с объединенной подгруппой, которая не является абелевой, нормальной в сомножителях, порядок которой должен быть достаточно большим по сравнению с длиной блока, используемого при самосинхронизирующемся блочно-поточном режиме.

Установка алгоритма

Установочная схема аналогична описанной в работе [6]. Таким образом, задаются группы $G_1 = \langle A * B, H = K, \varphi \rangle$, $G_2 = \langle A * B, H = K, \psi \rangle$, где изоморфизм ψ сформирован в результате деления ключа, состоящего из пары ψ и элемента $k_0 \in K$. Исходный текст записывается в виде нормальной формы в группе G_1 , затем умножается на k_0 и после редукции произведения в группе G_2 формируется зашифрованный текст.

Принципиальное описание самосинхронизирующихся поточных криптосистем можно представить следующим образом [7]:

$$\sigma_i = (c_{i-t}, c_{i-t+1}, \dots, c_{i-1}),$$

$$k_i = f(\sigma_i, k),$$

$$c_i = h(k_i, m_i),$$

где $\sigma_0 = (c_{-t}, c_{-t+1}, \dots, c_{-1})$ – начальное (открытое) состояние, k – ключ, k_i – единицы поточного ключа, f – функция, производящая текущий ключ, h – функция шифрования, m_i – единицы исходного текста, c_i – единицы зашифрованного текста.

Опишем этот алгоритм шифрования применительно к выбранной платформе.

1. Пусть начальное (открытое) состояние системы задано фиксированным групповым словом, записанным нормальной формой в группе G_1

$$\sigma_0 = (c_{-t}, c_{-t+1}, \dots, c_{-1}),$$

шифрование потока здесь сводится к шифрованию блоков длиной, не большей длины σ_0 . Первый корреспондент, начиная справа, делит исходное сообщение на блоки. Если крайний левый блок имеет меньшую длину, в него добавляются символы, не несущие смысловой нагрузки, в количестве, обеспечивающем полный блок.

Для создания элементов поточного ключа первый корреспондент умножает крайний правый элемент σ_0

$$\sigma_0 = (a_t^\sigma) b_t^\sigma \dots, b_2^\sigma, a_1^\sigma, b_1^\sigma$$

на секретный множитель k_0 справа и приводит σ_0 к нормальной форме в группе G_2

$$\text{н. ф. } (\sigma_0 \cdot k_0)_{G_2} = (h_1) k_1 \cdot (\bar{a}_t^\sigma) \bar{b}_t^\sigma \dots, \bar{b}_2^\sigma, \bar{a}_1^\sigma, \bar{b}_1^\sigma \rightarrow k_1(h_1),$$

откуда выделяет полученного слева представителя объединенной подгруппы k_1 (или h_1).

Для шифрования первой буквы элемент k_1 (или $\varphi(h_1)$) перебрасывается через неё справа налево с изменением по правилу приведения к нормальной форме в группе G_2 , образующийся слева представитель объединенной подгруппы (k'_1) можно не вычислять

$$(b_1 \cdot k_1)_{G_2} = k'_1 \cdot \bar{b}_1 \rightarrow \bar{b}_1.$$

3. Для шифрования второй буквы σ_1 записывается с использованием полученного в предыдущем пункте результата шифрования

$$\sigma_1 = (a_t^\sigma) b_t^\sigma \dots, b_2^\sigma, a_1^\sigma, \bar{b}_1,$$

как и в п.2, правый элемент σ_1 умножается на k_0 справа и приводится к нормальной форме в группе G_2

$$\text{н. ф. } (\sigma_1 \cdot k_0)_{G_2} \rightarrow k_2,$$

откуда выделяет полученного слева представителя объединённой подгруппы k_2 (или h_2), который перебрасывается через неё справа налево с изменением по правилу приведения к нормальной форме в группе G_2

$$\text{н. ф. } (a_1 \cdot \psi^{-1}(k_2))_{G_2} = h'_2 \cdot \bar{a}_1 \rightarrow \bar{a}_1.$$

4. Для шифрования третьей буквы σ_2 записывается с использованием всех предыдущих результатов шифрования

$$\sigma_3 = (a_t^\sigma) b_t^\sigma \dots, b_2^\sigma, \bar{a}_1, \bar{b}_1,$$

и т.д.

Описанный алгоритм позволяет зашифровать каждый фрагмент исходного сообщения заданной длины независимо от остальных его частей (элементов, блоков). Поскольку для инициализации процедуры шифрования каждого блока используются одни и те же начальные условия в виде σ_0 и k_0 , то потеря или добавление единиц зашифрованного текста не повлияют на дешифровку следующего блока сообщения.

На рисунке 1 представлена схема описанного алгоритма самосинхронизирующегося режима

шифрования.

Опишем алгоритм дешифрования:

1. Второй корреспондент выделяет зашифрованный блок

$$(\bar{a}_t) \bar{b}_t, \dots, \bar{b}_2, \bar{a}_1, \bar{b}_1.$$

2. Корреспондент умножает крайний правый элемент σ_0 на k_0 справа и приводит его к нормальной форме в группе G_2

$$\text{н. ф. } (\sigma_0 \cdot k_0)_{G_2} = (h_1) k_1 \cdot (\bar{a}_t^\sigma) \bar{b}_t^\sigma \dots, \bar{b}_2^\sigma, \bar{a}_1^\sigma, \bar{b}_1^\sigma \rightarrow k_1(h_1),$$

откуда выделяет полученного слева представителя объединённой подгруппы k_1 (или h_1) и вычисляет ему обратный k_1^{-1} (или h_1^{-1}).

Для расшифровки первой буквы элемент k_1^{-1} (или $\varphi(h_1^{-1})$) перебрасывается через неё справа налево с изменением по правилу приведения к нормальной форме в группе G_2 , образующийся слева представитель объединённой подгруппы $(k'_1)^{-1}$ не вычисляется

$$(\bar{b}_1 \cdot k_1^{-1})_{G_2} = (k'_1)^{-1} \cdot b_1 \rightarrow b_1.$$

3. Для дешифрования второй буквы σ_1 записывается с использованием полученной от первого корреспондента шифровки

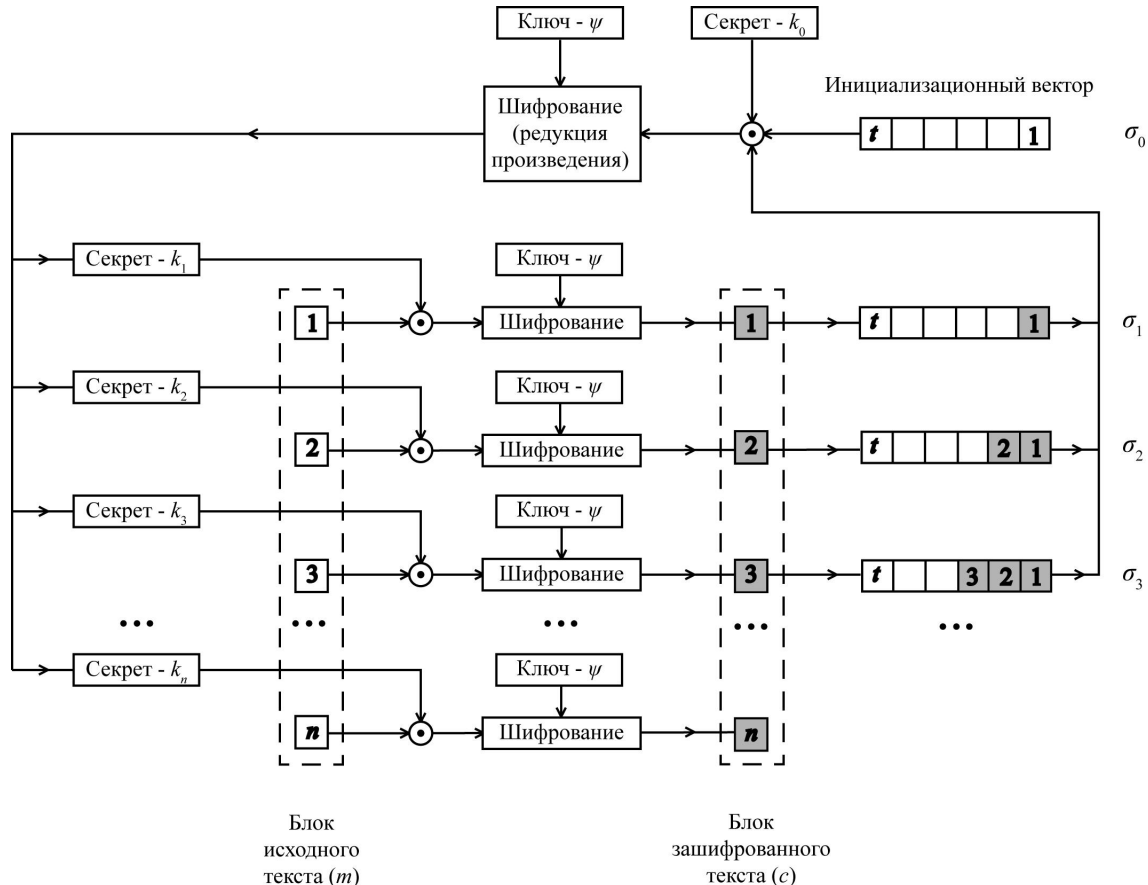


Рис. 1. Схема шифрования

$$\sigma_1 = (a_t^\sigma) b_t^\sigma \dots, b_2^\sigma, a_1^\sigma, \bar{b}_1,$$

далее правый элемент σ_1 домножается на k_0^{-1} справа и приводится к нормальной форме в группе G_2

$$\text{н. ф. } (\sigma_1 \cdot k_0^{-1})_{G_2} \rightarrow k_2^{-1},$$

откуда выделяет полученного слева представителя объединённой подгруппы k_2^{-1} (или h_2^{-1}), который перебрасывается через неё справа налево с изменением по правилу приведения к нормальной форме в группе G_2

$$\text{н. ф. } (\bar{a}_1 \cdot \psi^{-1}(k_2^{-1}))_{G_2} = (h'_2)^{-1} \cdot a_1 \rightarrow a_1,$$

4. Для дешифрования третьей буквы σ_2 записывается с использованием полученной от первого корреспондента шифровки

$$\sigma_3 = (a_t^\sigma) b_t^\sigma \dots, b_2^\sigma, \bar{a}_1, \bar{b}_1,$$

и т.д.

Рассмотрим пример использования алгоритма самосинхронизирующегося режима шифрования в случае групп подстановок.

Пример

Установка:

Абоненты выбрали группы $A=A_7$ и $B=A_7$, с подгруппами $H=A_6$ и $K=A_6$ ($H \subset A$ и $K \subset B$), объединёнными посредством изоморфизма

$$\varphi(h) = k_1 \cdot h \cdot k_1^{-1},$$

где $k_1=(1,2,6,5,3)$. Таким образом, задана группа

$$G_1=\langle A_7 * A_7, A_6=A_6, \varphi \rangle.$$

Известно, что группа A_7 является простой, а A_6 – некоммутативной. В соответствии с описанным алгоритмом, для формирования алфавита должны быть выбраны представители правых смежных классов в A и B , из которых формируется алфавит (см. табл. 1). Кроме того, устанавливают, что крайний справа элемент сообщения принадлежит группе B .

Далее абоненты надёжным образом обмениваются секретными элементами $k_0=(1,5,6,3,4)$, $k_\psi=(2,3,5,4,6)$, принадлежащими объединённой подгруппе, и формируют изоморфизм $\psi(h)=k_\psi \cdot \varphi(h) \cdot k_\psi^{-1}$ ($\psi: H \rightarrow K$). Таким образом, корреспонденты разделили группу

$$G_2=\langle A_7 * A_7, A_6=A_6, \psi \rangle.$$

Кроме того, абоненты открыто выбирают начальное состояние системы, пусть

$$\sigma_0=a_1, b_2, a_3, b_4, a_5, b_6.$$

Передача сообщения:

1. Первый абонент хочет передать сообщение следующего содержания: «Алим, Малик и Алики там камлала, клала и макала кита». Для этого он делит исходное сообщение на блоки длины 6, начиная справа (таб. 2).

Поскольку 7-ой блок оказался неполным, отправитель снабжает его дополнительными символами, соответствующими, например, последней шифруемой букве.

При шифровании одного блока сообщения, например, m_1 , выполняя п.1-2 алгоритма, получим $\bar{b}_1=b_6$, тогда $\sigma_1=a_1, b_2, a_3, b_4, a_5, \bar{b}_1=a_1, b_2, a_3, b_4, a_5, b_6$.

В соответствии с указаниями п.3, для второй буквы получим $\bar{a}_6=a_2$, тогда $\sigma_2=a_1, b_2, a_3, b_4, \bar{a}_6, \bar{b}_1=a_1, b_2, a_3, b_4, a_2, b_6$, и т.д.

Таблица 1

Используемый алфавит

№ п/п	Буква	Группа А		Группа В	
		Представитель	Обозначение	Представитель	Обозначение
1	А	(1,3,7,6)(2,4)	a_1	(1,6,2,4,5,3,7)	b_1
2	И	(1,7,3,5,6)	a_2	(1,4)(2,6,3)(5,7)	b_2
3	К	(1,5,3,2,6,7,4)	a_3	(1,4,2)(5,7,6)	b_3
4	Л	(1,7,5)(3,4,6)	a_4	(1,3,6,4,7,2,5)	b_4
5	М	(1,4,3,6,2,5,7)	a_5	(1,5,6,7,4)	b_5
6	Т	(2,3,5,6,7)	a_6	(1,2,4,7,3,5,6)	b_6

Таблица 2

А	А	А	Л	И	М	М	А	Л	И	К	И	А	Л	И	К	Т	А	М	К	А
a_1	b_1	a_1	b_4	a_2	b_5	a_3	b_1	a_4	b_2	a_3	b_2	a_1	b_4	a_2	b_3	a_6	b_1	a_5	b_3	a_1
m_7						m_6						m_5								
М	Л	А	Л	И	К	Л	А	Л	И	И	М	А	К	А	Л	И	К	И	Т	А
b_5	a_4	b_1	a_4	b_2	a_3	b_4	a_1	b_4	a_2	b_2	a_5	b_1	a_3	b_1	a_4	b_2	a_3	b_2	a_6	b_1
m_4			m_3					m_2						m_1						

Шифрованные блоки имеют вид (таб. 3).

2. Второй абонент получает шифрованные блоки $c_7, c_6, c_5, c_4, c_3, c_2, c_1$. При расшифровке одного блока сообщения, например, c_1 , выполняя п.1-2 алгоритма, получим $\bar{b}_4=b_1$, а $\sigma_1=a_1, b_2, a_3, b_4, a_5, \bar{b}_4=a_1, b_2, a_3, b_4, a_5, b_1$.

В соответствии с указаниями п.3, для второй буквы получим $\bar{a}_3=a_6$, а $\sigma_2=a_1, b_2, a_3, b_4, \bar{a}_3, \bar{b}_4=a_1, b_2, a_3, b_4, a_6, b_1$, и т.д. Тогда дешифрованный первый блок имеет вид

$$m_1=a_4, b_2, a_3, b_2, a_6, b_1,$$

аналогичная процедура выполняется для оставшихся 6-и блоков шифрованного сообщения.

В таблице 4 проанализирован результат шифрования по изменению количества отдельных букв алфавита по отношению к исходному тексту, и самих букв при шифровании.

По данным приведенной таблицы можно заключить: количество отдельных букв в исходном и шифрованном сообщениях не совпадают, без изменения («на месте») остались одна буква («Л») в 4-м блоке, одна буква («К») в 6-м блоке, одна буква («М») в 7-м блоке из общего числа 42 задействованных в сообщении.

Таблица 3

И	И	М	М	М	М	Т	И	Т	М	К	К	М	Т	М	Т	И	Т	К	Т	И
a_2	b_2	a_5	b_5	a_5	b_5	a_6	b_2	a_6	b_5	a_3	b_3	a_5	b_6	a_5	b_6	a_2	b_6	a_3	b_6	a_2
c_7						c_6						c_5								
А	Л	Т	Т	М	И	А	К	И	А	М	И	И	А	Т	М	Л	А	К	И	Т
b_1	a_4	b_6	a_6	b_5	a_2	b_1	a_3	b_2	a_1	b_5	a_2	b_2	a_1	b_6	a_5	b_4	a_1	b_3	a_2	b_6
c_4			c_3						c_2						c_1					

Таблица 4

Элементарный анализ результата по схеме самосинхронизирующегося шифрования

№ п/п	Буква	Количество в тексте		Кол-во не измен. место
		Исходный	Шифровка	
1	А	12	5	0
2	И	9	10	0
3	К	6	5	1
4	Л	8	2	1
5	М	5	10	1
6	Т	2	10	0

Заключение

Для повышения эффективности рассматриваемого алгоритма защиты следует использовать блоки большей длины. В этом случае большее число букв будет исправлено набором создаваемых при самосинхронизирующемся режиме секретных элементов объединенной подгруппы – поточным ключом для каждого блока, который получается при редукции произведения $(\sigma_i \cdot k_0)$. При реализации описанной процедуры защиты передачи данных использован классический вариант редукции произведения (см. [8-10]), применяющий правые смежные классы для записи нормальной формы, в этом случае работа алгоритма начинается с крайней правой буквы в блоке. При защите потоков, возможно, удобнее для задания нормальной формы элементов использовать левые смежные классы и соответствующую

такому заданию редукцию произведения, т.к. работа алгоритма при этом начинается с изменения крайней левой буквы блока.

В рассмотренном примере, ввиду ограниченности вычислительной мощности, использовался короткий алфавит, состоящий из 6-и букв, поэтому для использования традиционных алфавитов потребуется удлинение сообщений ввиду необходимости компенсации недостающих букв комбинациями элементов «квазиалфавита». Поскольку индекс $|A_n : A_{n-1}| = n$, а он отвечает за размер алфавита, то его возможная длина ограничивается только производительностью доступных вычислительных мощностей. Прогресс в области вычислительной техники, в частности появление квантовых компьютеров, в перспективе позволит использовать общеупотребительные алфавиты.

Благодарности

Автор благодарит преподавателя кафедры Высшей математики МИИГАиК Дергилёву А.Э. за ценные советы при написании статьи, а также за помощь в проведении экспериментов и обработке данных.

Список литературы

1. Wenyi Morty Zhang, Annie Dai, Keegan Ryan, Dave Levin, Nadia Heninger, and Aaron Schulman. 2025. Don't Look Up: There Are Sensitive Internal Links in the Clear on GEO Satellites. In Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security (CCS '25), October 13–17. 2025. Taipei. Taiwan. ACM. New York. NY. USA. 19 pages. <https://doi.org/10.1145/3719027.3765198>.
2. Иванова К.В., Сальников А.Ф., Мормуль Р.В. Искусственный интеллект для контроля передачи данных в тактическом звене управления с использованием многослойного и многопоточного шифрования геопространственной обстановки // Вестник Пермского университета. Математика. Механика. Информатика. 2023. Вып. 2(61). С. 65-71. DOI: 10.17072/1993-0550-2023-2-65-71.
3. Панкратов И.В. О поточных и автоматных шифрсистемах с симметричным ключом // ПДМ. 2009. №3(5). С. 59-68.
4. Романьков В.А. Алгебраическая криптология: монография. Омск: ОмГУ. 2020. 262 с.
5. Дурнев В.Г., Зеткина О.В. Методы комбинаторной теории групп в современной криптографии. Ярославль: ЯрГУ. 2017. 52 с.
6. Лоссов К.И., Дергилёва А.Э. Синхронные криптосистемы для защиты пространственно-временных данных, использующие мультипликативную редукцию в свободных произведениях групп с объединением // Мониторинг. Наука и технологии. 2025. №4(66). С. 70-75.
7. Романьков В.А. Введение в криптографию. М.: ФОРУМ. 2012. 240 с.
8. Курош А.Г. Теория групп. М.: Наука. 1967. 648 с.
9. Магнус В., Каррас А., Солитэр Д. Комбинаторная теория групп. М.: Наука. 1974. 456 с.
10. Холл М. Теория групп. М.: Изд-во ин. л-ры. 1962. 467 с.

References

1. Wenyi Morty Zhang, Annie Dai, Keegan Ryan, Dave Levin, Nadia Heninger, and Aaron Schulman. 2025. Don't Look Up: There Are Sensitive Internal Links in the Clear on GEO Satellites. In Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security (CCS '25), October 13–17. 2025. Taipei. Taiwan. ACM. New York. NY. USA. 19 pages. <https://doi.org/10.1145/3719027.3765198>.
2. Ivanova K.V., Sal'nikov A.F., Mormul' R.V. Artificial intelligence for data transmission control at the tactical command level using multi-layer and multi-stream encryption of the geospatial environment. *Vestnik Permskogo universiteta. Matematika. Mekhanika. Informatika*. 2023. No. 2(61). Pp. 65-71. DOI: 10.17072/1993-0550-2023-2-65-71.
3. Pankratov I.V. On stream and automaton cipher systems with a symmetric key. *PDM*. 2009. No. 3(5). Pp. 59-68.
4. Romankov V.A. *Algebraicheskaya kriptologiya: monografiya* [Algebraic Cryptology: A Monograph]. Omsk: Omsk State University. 2020. 262 p.
5. Durnev V.G., Zetkina O.V. *Metody kombinatornoy teorii grupp v sovremennoy kriptografii* [Methods of combinatorial group theory in modern cryptography]. Yaroslavl: Yaroslavl State University. 2017. 52 p.
6. Lossov K.I., Dergileva A.E. Synchronous cryptosystems for the protection of spatio-temporal data using multiplicative reduction in free products of groups with amalgamation. *Monitoring. Nauka i tekhnologii*. 2025. No. 4(66). Pp. 70-75.
7. Romankov V.A. *Vvedeniye v kriptografiyu* [Introduction to Cryptography]. Moscow: FORUM. 2012. 240 p.
8. Kurosh A.G. *Teoriya grupp* [Group theory]. Moscow: Nauka. 1967. 648 p.
9. Magnus V., Karras A., Solitaire D. *Kombinatornaya teoriya grupp* [Combinatorial group theory]. Moscow: Nauka. 1974. 456 p.
10. Holl M. *Teoriya grupp* [Group theory]. Moscow: Publishing house of foreign books. 1962. 467 p.

Сведения об авторах Принадлежность к организации

Лоссов Константин Иванович

кандидат физико-математических наук, заведующий кафедрой Высшей математики, Московский государственный университет геодезии и картографии, 105064, Москва, Россия

Information about authors Affiliations

Lossov Konstantin Ivanovich

Candidate of Physical and Mathematical Sciences, the Chief of the Chair of Higher Mathematics, Moscow State University of Geodesy and Cartography, 105064, Moscow, Russia

Поступила в редакцию 06.02.2026 г.