

ОРГАНИЗАЦИЯ ИЕРАРХИЧЕСКОГО ДОСТУПА К ДАННЫМ ГИС С ПОМОЩЬЮ ЭЛЕМЕНТОВ ОБЪЕДИНЯЕМОЙ ПОДГРУППЫ АМАЛЬГАМЫ

Лоссов К.И.

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

В условиях обработки и хранения больших объемов связанных пространственно-временных данных критически возрастает потребность в бесшовных криптографических методах разграничения доступа, обладающих постквантовой стойкостью. Целью исследования является разработка иерархической схемы шифрования топологии и атрибутов графовых баз ГИС, позволяющей хранить зашифрованную информацию в виде единой монолитной строки, доступ к сегментам которой дифференцируется динамически в зависимости от роли субъекта в древовидной структуре прав. В основу предложенного подхода положен оригинальный математический синтез аппарата комбинаторной теории групп – свободного произведения групп с объединенной подгруппой и классической криптографической схемы Акла-Тейлора для частично упорядоченных множеств классов безопасности. Разграничение доступа реализуется за счет уникальных свойств приведения групповых слов к нормальной форме.

Разработана технология сериализации элементов графа с использованием алфавита представителей левых смежных классов множителей амальгамы. Показано, что суперпользователь, обладая полным набором постоянных элементов (ключей) из объединяемой подгруппы, способен дешифровать любой подграф. Для нижестоящих уровней СУБД генерируются ключи на основе однонаправленного каскадного «просачивания» элементов подгруппы сквозь блоки под действием автоморфизмов. Это обеспечивает автоматический доступ пользователя ко всем подчиненным узлам при его полной изоляции от параллельных ветвей графа.

Предложенная схема исключает избыточность хранения и претендует на постквантовую стойкость, обусловленную SQ-универсальностью амальгам. Наличие «алгебраического остатка» на конце строки обеспечивает встроенный контроль целостности данных, исключая несанкционированную подмену графа злоумышленником.

Ключевые слова: графовые базы данных, разграничение прав доступа, контроль целостности графа, свободное произведение с объединением, нормальная форма, представители смежных классов, сдвиговой множитель, аутентифицированное шифрование, постквантовая криптография.

HIERARCHICAL GIS DATA ACCESS VIA ELEMENTS OF AMALGAMATED SUBGROUP

Lossov K.I.

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

Managing and securing massive spatiotemporal datasets critically demands seamless cryptographic access control mechanisms capable of withstanding post-quantum threats. This paper introduces a hierarchical encryption framework designed specifically for the topology and attributes of graph-based GIS databases. The proposed method consolidates encrypted information into a single monolithic string, where segment-level access is dynamically adjusted according to the user's role within a tree-like privilege structure. At its core, the approach relies on a novel mathematical synthesis that blends combinatorial group theory – specifically, the free product of groups with an amalgamated subgroup – with the classical Akl-Taylor cryptographic scheme tailored for partially ordered security classes. Security enforcement hinges on the inherent algebraic properties of group words when reduced to their normal form.

We also present a novel graph serialization technique that utilizes the alphabet of left coset representatives derived from the amalgam factors. Within this framework, a superuser holding a complete set of static keys from the amalgamated subgroup can successfully decrypt any given subgraph. For lower-level database management systems, subordinate keys are derived through a unidirectional cascading «percolation» of subgroup elements across blocks, driven by automorphisms. This mechanism guarantees that users automatically inherit access to all descendant nodes while remaining strictly isolated from parallel branches of the graph topology.

By architecture, the scheme eliminates storage overhead and achieves post-quantum resilience rooted in the SQ-universality of amalgams. Furthermore, an «algebraic remainder» appended to the end of the data string functions as a built-in integrity check, effectively preventing malicious graph tampering or unauthorized payload substitution.

Keywords: graph databases; access control; graph integrity control; free product with amalgamation; normal form; coset representatives; shift multiplier; authenticated encryption; post-quantum cryptography.

Введение

Обеспечение конфиденциальности, целостности и доступности пространственно-временных данных (ПВД) является одной из наиболее сложных задач современной геоинформационной безопас-

ности. Еще в 1983 году [1] поднималась проблема контроля доступа для организаций с иерархической структурой, таких как правительство, дипломатический корпус, армия, в управлении базами данных, содержащими конфиденциальную информацию,

или в защите промышленных секретов. Там же было предложено решение проблемы контроля доступа в иерархии, основанное на криптографии, при этом иерархия представлена частично упорядоченным множеством. Однако простая реализация этой схемы требует от пользователей, занимающих высокие позиции в иерархии, хранения большого количества криптографических ключей. Ввиду этого, авторами осуществлена попытка достижения компромисса между временем и хранением для решения проблемы управления ключами. В работе [2] замечено, что всеобъемлющая структура, достаточно общая и гибкая для работы с пространственными аспектами в реальных приложениях, все еще отсутствует. Там же предложено расширение GEO-RBAC для поддержки иерархий, моделирования разрешений, наследования пользователей и активаций, а также ограничений разделения обязанностей. Продолжением работы [1] являются, в частности, современные статьи [3,4]: в [3] описывается готовый к практике инженерный инструмент, основанный на геометрии подпространств, а в [4] поверх жестких криптографических замков добавлена «умная» система поведенческого контроля. О неугасающем интересе к представленной в работе тематике свидетельствуют, например, работы [5,6,7].

Динамические ПВД, циркулирующие в системах мониторинга беспилотного транспорта, спутникового позиционирования (GNSS) и интернета вещей (IoT), принципиально отличаются от традиционных текстовых или числовых массивов. В практике защиты геоданных существует несколько подходов к безопасности: использование туннелирования (VPN, IPsec), шифрования транспортного уровня (TLS/SSL) и межсетевых экранов. Такой подход защищает только канал передачи («трубу»), но оставляет данные абсолютно уязвимыми внутри конечных узлов распределенной системы или при их хранении в базах данных ГИС. Кроме того, он не обеспечивает разграничения прав доступа к отдельным слоям или элементам геоданных. Другими методами являются использование алгоритмов внесения искусственного шума (при этом безвозвратно ухудшается точность позиционирования для высокоточных систем), а также сквозное криптографическое шифрование при помощи блочных (AES, «Кузнечик») или потоковых (ChaCha20, «Магма» в режиме гаммирования) шифров, применяющихся непосредственно к координатным массивам. Последний подход теоретически более надежен, однако его стандартная реализация сталкивается с жесткими ресурсными ограничениями.

Традиционные архитектурные модели контроля доступа (такие как GEO-RBAC [2]) реализуются на уровне программного обеспечения ГИС-серверов или СУБД, что оставляет данные уязвимыми при прямом компрометировании базы данных. Крипто-

графические методы защиты иерархических баз данных (например, схемы на основе HIBE или Акла-Тейлора [1]) зачастую приводят к высокой избыточности ключевой информации из-за необходимости изолированного шифрования разных уровней детализации объектов карт. Кроме того, они демонстрируют уязвимость к специфическим атакам, направленным на подмену или сдвиг пространственных координат внутри зашифрованных таблиц. В данной работе предлагается преодолеть указанные ограничения за счет внедрения алгебраических структур непосредственно в контур криптографической защиты хранилищ ГИС. Применение математического аппарата амальгамы групп [8] позволяет реализовать гибкий механизм каскадного управления доступом к векторным объектам на уровне физического хранения, обеспечивая однопроходную аутентификацию данных за счет жестких правил нормализации слов в амальгамах, которые при определенных условиях обладают свойством SQ-универсальности [9,10], что обуславливает высокую вычислительную сложность проблемы принадлежности элемента к двойному смежному классу и делает эти системы надежным фундаментом для защиты от квантового криптоанализа. Однако до сих пор потенциал указанных объектов практически не исследовался применительно к обеспечению безопасности сложных ветвящихся топологий в графовых базах данных ГИС.

Материалы и методы

Определение. Если W – групповое слово из $G=\langle A*B, H=K, \varphi \rangle$, а $W_n=c_1 \cdots c_n \cdot h$ – его нормальная форма, тогда

$$\text{унф}(W)=c_1 \cdots c_n$$

будем называть *усеченной нормальной формой* W .

Определение. Пусть h – элемент объединяемой подгруппы, W – слово, записанное в нормальной форме, причем $\text{унф}(W)=W \notin H$. Тогда элемент $h \cdot W$ после приведения к нормальной форме примет вид $\tilde{W} \cdot h_w$, где $\tilde{W}=\text{унф}(h \cdot W)$, $h_w \in H$. Элемент h_w будем называть остатком мультипликативной редукции произведения $h \cdot W$.

Для h_w выполняется равенство

$$h \cdot W = \tilde{W} \cdot h_w \quad (1)$$

Пусть ψ_h – функция, заданная на множестве слов W из определения выше, такая, что

$$\psi_h(W)=h_w$$

В дальнейшем нам будет удобно вместо $\psi_h(W)$ писать $h(W)$, таким образом

$$\psi_h(W)=h(W)=h_w$$

Организацию иерархического доступа к графовой базе данных объясним на примере.

Пусть имеется взвешенный граф, являющийся корневым деревом (см. рис.), с вершинами $\{1,1\}$, $\{2,1\}$, $\{2,2\}$, $\{2,3\}$, $\{3,1\}$, ..., $\{3,6\}$, где первый индекс означает номер уровня, а второй – номер вершины на этом уровне, и ребрами с весами W_{11} , W_{21} , W_{22} , W_{23} , W_{31} , ..., W_{36} , являющимися базами данных, представленными групповыми словами амальгамы. Метки вершин будут являться идентификаторы пользователей соответствующего уровня (например, вершине $\{1,1\}$ соответствует пользователь $]1,1[$).

Для шифрования объединенной графовой базы (включающей в себя базы $W_{11}, \dots, W_{36}$), изображенной на рисунке, выбираются секретные элементы

$$h_{11}, h_{21}, h_{22}, h_{23}, h_{31}, h_{32}, h_{33}, h_{34}, h_{35}, h_{36}, \quad (2)$$

принадлежащие объединяемой подгруппе. Далее, W_{11} умножается слева на h_{11} и, после приведения произведения $h_{11} \cdot W_{11}$ к нормальной форме, получаем $\tilde{W}_{11} \cdot h_{11} (W_{11})$, где $\tilde{W}_{11}$ – зашифрованная база W_{11} , а $h_{11} (W_{11})$ – остаток мультипликативной редукции. Аналогично шифруются базы W_{21} , W_{22} , W_{23} , используя множители

$$h_{21}^o = h_{11}(W_{11}) \cdot h_{21}, h_{22}^o = h_{11}(W_{11}) \cdot h_{22}, h_{23}^o = h_{11}(W_{11}) \cdot h_{23}, \quad (3)$$

а затем базы W_{3k} , $k=1, \dots, 6$ при помощи элементов

$$\begin{aligned} h_{3j}^o &= h_{21}^o(W_{21}) \cdot h_{3j}, j=1,2,3; h_{34}^o = h_{22}^o(W_{22}) \cdot h_{34}; \\ h_{3m}^o &= h_{23}^o(W_{23}) \cdot h_{3m}, m=5,6. \end{aligned} \quad (4)$$

Применяемый алгоритм шифрования подробно описан в работе [11], при сохранении требований ненормальности и некоммутативности объединяемой подгруппы.

Пользователю $]1,1[$ первого (высшего) уровня

доступа передаются ключи шифрования-дешифрования (для дешифрования применяется умножение на элементы, обратные h_{11} , (3) и (4)). Пользователям $]2,1[$, $]2,2[$, $]2,3[$ второго уровня передаются наборы ключей $h_{21}^o, h_{31}, h_{32}, h_{33}; h_{22}^o, h_{34}; h_{23}^o, h_{35}, h_{36}$ соответственно. Ключи шифрования-дешифрования (4) передаются пользователям $]3,i[$, $i=1, \dots, 6$, причем $]3,i[$ передается только ключ h_{3i}^o .

Пользователь $]3,k[$ ($k=1, \dots, 6$) может расшифровать только базу W_{3k} ; $]2,1[$ может расшифровать базы $\tilde{W}_{21}, \tilde{W}_{31}, \tilde{W}_{32}, \tilde{W}_{33};]2,2[- \tilde{W}_{22}, \tilde{W}_{34};]2,3[- \tilde{W}_{23}, \tilde{W}_{35}, \tilde{W}_{36}$. Пользователь первого уровня $]1,1[$ может расшифровать любую из баз второго и третьего уровней и $\tilde{W}_{11}$.

Элементы (2) за исключением h_{11} являются множителями-сдвигами, препятствующими пользователям низкого уровня получать доступ к базам более высокого уровня, поскольку, зная только произведение элементов, невозможно однозначно восстановить сомножители. Так элемент h_{2i} является сдвиговым элементом, препятствующим пользователю $]2,i[$ ($i=1, \dots, 3$) получить доступ к базе W_{11} , поскольку для расшифровки $\tilde{W}_{11}$ требуется знание элемента $h_{11} (W_{11})$, но у него в распоряжении имеется только $h_{21}^o = h_{11}(W_{11}) \cdot h_{21}$. Также h_{3k} гарантирует защиту баз пользователей $]2,i[$ от несанкционированного прочтения пользователями $]3,k[$, $k=1, \dots, 6$.

Если обращения от пользователей к базам более высокого уровня блокируются, то вместо набора (2) можно взять единственный элемент h_{11} , которым шифруется W_{11} . Затем база W_{2i} шифруется элементом $h_{11} (W_{11})$, а пользователям $]2,i[$ передаются ключи шифрования $(h_{11} (W_{11}))(W_{2i})$ для создания баз $\tilde{W}_{3k}$, ключи дешифрования которых $((h_{11} (W_{11}))(W_{2i}))^{-1}$ направляются пользователям $]3,k[$.

При изменении любой из баз – W_{31}, W_{32}, W_{33} , пользователь $]2,1[$ заменяет ключи для пользователей $]3,1[$, $]3,2[$, $]3,3[$, используя (4). Аналогично

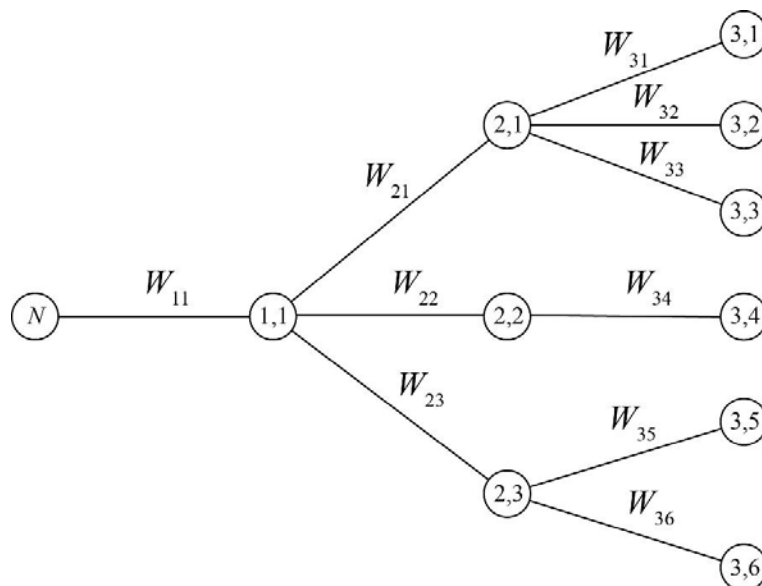


Рис. 1. Схема базы данных

поступают пользователи $[2,2[$ и $[2,3[$, заменяя ключи для $[3,4[$ и $[3,5[$, $[3,6[$ соответственно при изменении баз W_{34} или W_{35} , W_{36} . При изменении базы W_{2i} , $i=1,2,3$, пользователь $[1,1[$ формирует для $[2,i[$ новый ключ, применяя равенство (3), и уже пользователь $[2,i[$ создает новые ключи для шифрования соответствующих баз W_{3k} (например, при $i=2$ создается ключ для W_{34}). В случае модернизации W_{11} заменяются все ключи согласно равенствам (1), (3), (4).

Замечание. Выбранные элементы (2) не требуют замены при трансформации баз, при этом модернизация какой-либо базы более высокого уровня требует замены ключей у пользователей уровней ниже.

При оцифровке баз $W_{11}, \dots, W_{36}$ левыми смежными классами потребуются, чтобы получающиеся при этом слова начинались с элемента, принадлежащего одной группе, например, A , и оканчивались элементом, принадлежащим другому множеству (здесь это B). В случае, если какое-то слово из $W_{11}-W_{36}$ оканчивается на $a \in A$, то дописывается представитель левого смежного класса $B:K$, соответствующий выбранному техническому символу, например, пробелу. Таким образом, объединенная база данных представляется в виде группового слова $W_{11}W_{21}W_{22}W_{23}W_{31} \dots W_{36}$, записанного в нормальной форме.

Пользователям, кроме $\{1,1\}$, необходимо предоставить позиции в слове, соответствующие их уровню доступа.

Пример

Рассмотрим базу данных, сформированную по схеме, изображенной на рисунке, на примере модельной группировки спутников, предназначенных для мониторинга земной поверхности:

– W_{11} содержит информацию о всей группировке: количество спутников, общее состояние работоспособности системы и структуры содержащихся в ней данных – это информация первого уровня;

– каждое из слов W_{21} , W_{22} , W_{23} содержит информацию о конкретном спутнике: идентификатор в системе, дату и время записи данных, координаты и составляющие скорости – это информация второго уровня;

– каждое из слов W_{3k} ($k=1, \dots, 6$) содержит матрицу данных, соответствующих местности съемки; при $k=1, 2, 3$ рассмотрены данные разных каналов мультиспектральной съемки, при $k=4$ приведена матрица результатов радиолокационной съемки с комплексными элементами, при $k=5, 6$ приведены данные высоты поверхности моря (SSH) и высоты значительных волн (SWH), полученные из обработки альтиметрических измерений – это информация третьего уровня.

Рассмотрим результаты шифрования-дешифрования данных по пути $\{1,1\} - \{2,3\} - \{3,5\}$. В

качестве платформы выбрано свободное произведение групп $A=A_{14}$ и $B=A_{14}$, с подгруппами $H=A_{12}$ и $K=Stab_{A_{12}}$ (6,7), объединенными посредством изоморфизма $\varphi(h)=k_0 \cdot h \cdot k_0^{-1}$, где $k_0=(6,13)(7,14)$. Также абоненты открыто выбирают алфавит из представителей левых смежных классов, содержащий в себе строчные и прописные буквы русского, латинского и греческого алфавитов, числа и специальные символы. Кроме того, заранее устанавливается, что каждое групповое слово W должно начинаться с представителя группы A и заканчиваться представителем группы B , в случае необходимости (нечетного количества элементов в слове) оно дополняется заранее выбранным нейтральным элементом, например, нижним подчеркиванием « $_$ » (в заданном алфавите символизирует пробел), принадлежащим множеству B .

Пусть W_{11} содержит следующую информацию (без пробелов):

Группировка *Аестофθαλμος*; Всего_5; Активные_А1, А3, А4; Запасные_А2, А5; Состояние=хор; 2у=3; 3у=6_1=3, 2=1, 3=2.

Если присвоить элементам алфавита (представителям смежных классов) числовые индексы в списке и поставить им в соответствие стандартный алфавит, то W_{11} можно записать следующим образом:

a[35], b[16], a[19], b[15], a[15], b[9], a[16], b[14], a[3], b[10], a[1], b[173], a[149], b[129], a[143], b[139], a[145], b[132], a[125], b[135], a[136], b[139], a[142], b[176], a[34], b[17], a[6], b[4], a[14], b[173], a[120], b[176], a[32], b[10], a[18], b[9], a[3], b[13], a[27], b[6], a[173], b[89], a[116], b[175], a[89], b[118], a[175], b[89], a[119], b[176], a[39], b[1], a[15], b[1], a[17], b[13], a[27], b[6], a[173], b[89], a[117], b[175], a[89], b[120], a[176], b[48], a[14], b[17], a[18], b[14], a[31], b[13], a[9], b[6], a[181], b[21], a[14], b[16], a[176], b[117], a[19], b[181], a[118], b[176], a[118], b[19], a[181], b[121], a[173], b[116], a[181], b[118], a[175], b[117], a[181], b[116], a[175], b[118], a[181], b[117]

Аналогично записывается W_{23} :

А3; 15082026; 02.15.30; $X=1425390.124$; $Y=5214080.453$; $Z=4589120.782$; $V_x=-5120.34$; $V_y=-1450.56$; $V_z=4850.56$;

или

a[89], b[118], a[176], b[116], a[120], b[115], a[123], b[117], a[115], b[117], a[121], b[176], a[115], b[117], a[174], b[116], a[120], b[174], a[118], b[115], a[176], b[112], a[181], b[116], a[119], b[117], a[120], b[118], a[124], b[115], a[174], b[116], a[117], b[119], a[176], b[113], a[181], b[120], a[117], b[116], a[119], b[115], a[123], b[115], a[174], b[119], a[120], b[118], a[176], b[114], a[181], b[119], a[120], b[123], a[124], b[116], a[117], b[115], a[174], b[122], a[123], b[117], a[176], b[110], a[86], b[181], a[179], b[120], a[116], b[117], a[115], b[174], a[118], b[119], a[176], b[110], a[87], b[181], a[179], b[116], a[119], b[120], a[115],

b[174], a[120], b[121], a[176], b[110], a[88], b[181], a[119], b[123], a[120], b[115], a[174], b[120], a[121], b[176]

Для W_{35} – матрицы значений измеренных величин, запятая выступает разделителем между элементами в строке, точка с запятой обозначает переход на новую строку:

-27.15, -27.18, -27.20, -27.22; -27.17, -27.20, -27.23, -27.25; -27.20, -27.22, -27.24, -27.27 _

или

a[179], b[117], a[122], b[174], a[116], b[120], a[175], b[179], a[117], b[122], a[174], b[116], a[123], b[175], a[179], b[117], a[122], b[174], a[117], b[115], a[175], b[179], a[117], b[122], a[174], b[117], a[117], b[176], a[179], b[117], a[122], b[174], a[116], b[122], a[175], b[179], a[117], b[122], a[174], b[117], a[115], b[175], a[179], b[117], a[122], b[174], a[117], b[118], a[175], b[179], a[117], b[122], a[174], b[117], a[120], b[176], a[179], b[117], a[122], b[174], a[117], b[115], a[175], b[179], a[117], b[122], a[174], b[117], a[117], b[175], a[179], b[117], a[122], b[174], a[117], b[119], a[175], b[179], a[117], b[122], a[174], b[117], a[122], b[173]

Для шифрования этого пути выбраны секретные элементы:

$h_{11}=(2, 6, 7, 11, 5) (3, 12, 10, 9, 4), h_{23}=(1, 10, 7, 9, 11)(2, 12, 3, 5, 6, 8, 4), h_{35}=(1, 4, 5, 2, 6, 10, 12) (3, 9, 7, 8, 11).$

Шифрование базы:

1. Пользователь]1,1[вычисляет $h_{11} W_{11} = \tilde{W}_{11} \cdot k_{w11}$, где $\tilde{W}_{11}$ равняется

a[175], b[158], a[64], b[65], a[104], b[80], a[121], b[8], a[8], b[124], a[158], b[165], a[105], b[49], a[135], b[155], a[84], b[40], a[151], b[5], a[82], b[139], a[90], b[106], a[87], b[37], a[27], b[65], a[14], b[46], a[50], b[126], a[175], b[35], a[179], b[73], a[7], b[68], a[2], b[86], a[146], b[94], a[59], b[62], a[173], b[24], a[146], b[181], a[16], b[126], a[48], b[65], a[3], b[121], a[166], b[154], a[71], b[76], a[90], b[117], a[28], b[41], a[173], b[5], a[148], b[46], a[88], b[17], a[23], b[174], a[31], b[115], a[166], b[59], a[159], b[35], a[85], b[125], a[176], b[94], a[116], b[86], a[165], b[42], a[55], b[7], a[145], b[40], a[79], b[173], a[181], b[179], a[79], b[128], a[13], b[72], a[134], b[116], a[161], b[97],

при этом $k_{w11} = k_{11}(W_{11}) = (1, 12, 3, 13, 11, 9)(2, 4).$

При переходе к стандартному алфавиту шифрованное сообщение примет вид:

"," K, b, c, P, r, 6, 'з', 'з', 9, K, P, Q, 'Т', λ, Н, v, 'И', Г, 'д', t, o, B, R, y, 'Е', 'ы', c, 'о', П, 'У', β, ", 'Г', "-, k, 'ж', f, 'б', x, χ, F, 'Б', 'Я', _ , 'ш', χ, "=", 'р', β, 'С', c, v, 6, Σ, Z, i, n, B, 2, Ъ, K, _ , 'д', ω, П, z, 'с', 'ч', ", 'я', 0, Σ, 'Б', Λ, 'Г', w, α, ";, F, 1, x, P, 'Л', 'Ш', 'ж', φ, 'И', q, _ , "=", "-, q, δ, 'н', j, κ, 1, N, I

2. Далее пользователь]1,1[вычисляет

$h_{23}^o = \varphi(k_{11}(W_{11})) \cdot h_{23} = (1, 3, 8, 4, 12, 5, 6)(7, 9, 10),$ и передает пользователю]2,3[ключ расшифрования $(h_{23}^o)^{-1}$.

3. После чего происходит шифрование базы W_{23} по правилу $h_{23}^o W_{23} = \tilde{W}_{23} \cdot k(W_{23})^o$, где $\tilde{W}_{23}$

a[89], b[170], a[87], b[24], a[167], b[70], a[69], b[87], a[45], b[83], a[92], b[37], a[179], b[116], a[139], b[87], a[44], b[8], a[126], b[32], a[139], b[96], a[161], b[95], a[93], b[144], a[45], b[137], a[91], b[32], a[34], b[97], a[46], b[11], a[109], b[161], a[181], b[51], a[64], b[75], a[25], b[68], a[10], b[13], a[36], b[15], a[6], b[20], a[109], b[136], a[31], b[127], a[92], b[149], a[7], b[90], a[26], b[70], a[176], b[46], a[27], b[148], a[87], b[157], a[161], b[8], a[123], b[53], a[157], b[21], a[158], b[94], a[11], b[134], a[176], b[110], a[87], b[25], a[158], b[52], a[74], b[136], a[39], b[164], a[140], b[18], a[22], b[105], a[138], b[178], a[121], b[173], a[6], b[152], a[34], b[34], a[106], b[17],

при этом $k(W_{23})^o = k_{23}^o(W_{23}) = (1, 10, 14, 11, 2, 5, 13, 3, 12, 4, 9).$

В символах примет вид:

A, X, y, 'ш', T, h, g, y, O, u, D, 'Е', "-", 1, o, y, 'Н', 'з', β, 'А', o, H, N, G, E, v, O, v, C, 'А', 'В', I, П, 'л', U, N, "=", 'Ф', b, m, 'ш', f, κ, 'н', 'Д', 'п', e, 'ф', U, μ, 'я', γ, D, Λ, 'ж', B, ъ, h, ";, П, 'ы', ω, y, I, N, 'з', 8, 'Ц', I, 'x', K, F, 'л', κ, ";, V, y, 'ш', K, 'X', I, μ, 'З', П, π, 'т', 'ц', Q, ξ, "?" , 6, _ , e, Δ, 'В', 'В', R, 'с'

4. Пользователи]1,1[и]2,3[могут вычислить

$h_{35}^o = \varphi(k_{23}^o(W_{23})) \cdot h_{35} = (1, 12, 5, 10, 8, 11, 6, 9, 4, 7, 3),$

и]2,3[передает]3,5[ключ $(h_{35}^o)^{-1}$.

5. Осуществляется шифрование базы W_{35} по правилу $h_{35}^o W_{35} = \tilde{W}_{35} \cdot k(W_{35})^o$, где $\tilde{W}_{35}$

a[157], b[4], a[57], b[122], a[102], b[47], a[175], b[57], a[73], b[75], a[176], b[109], a[113], b[155], a[95], b[88], a[48], b[171], a[37], b[68], a[79], b[33], a[74], b[45], a[109], b[18], a[11], b[126], a[1], b[97], a[156], b[28], a[115], b[127], a[89], b[140], a[124], b[132], a[87], b[78], a[110], b[62], a[11], b[33], a[42], b[52], a[73], b[145], a[134], b[48], a[97], b[95], a[174], b[178], a[179], b[162], a[91], b[5], a[44], b[143], a[18], b[147], a[173], b[132], a[158], b[170], a[22], b[137], a[65], b[104], a[95], b[103], a[94], b[50], a[64], b[38], a[146], b[77], a[136], b[3], a[34], b[109], a[58], b[102],

при этом $k(W_{35})^o = k_{35}^o(W_{35}) = (1, 14, 4, 3)(2, 9, 8, 12, 10, 11, 13, 5).$

В символах примет вид:

I, r, Ъ, 7, N, 'Р', ", " , Ъ, k, m, ";, U, Y, H, G, z, 'С', Ψ, 'Е', f, q, 'Б', I, O, U, 'т', 'л', β, 'а', I, Θ, ъ, 0, γ, A, π, 9, 0, y, p, V, 'Я', 'л', 'Б', 'Л', 'X', k, φ, κ, 'С', I, G, ", " , "?" , "-, Ξ, C, 'д', 'Н', τ, 'т', ψ, _ , θ, K, X, 'ц', v, c, P, G, O, F, 'У', b, 'Ж', χ, o, μ, v, 'В', U, 'Б', N

Процедура дешифрования каждой из баз $\tilde{W}_{11}$, $\tilde{W}_{23}$ и $\tilde{W}_{35}$ осуществляется аналогичным образом, при этом в качестве множителя используются $(h_{11})^{-1}$, $(h_{23}^o)^{-1}$ и $(h_{35}^o)^{-1}$ соответственно.

При изменении одного из символов базы W_{35} , например, третьего слева (a[122]), $\tilde{W}_{35}$ отличается от результатов пункта 5 на 92.86% (всего символов 84, количество точных совпадений на тех же позициях в слове – 6), при этом $k_{w35}^o = k_{35}^o(W_{35}) = (1, 10, 12, 3, 2, 5, 13)$

$(8,14)(9,11)$, и $(k^o_{w35})^{-1}$ не равняется $(k^o_{w35})^{-1}$. Таким образом, формируемый справа остаток мультипликативной редукции может быть использован для контроля целостности данных базы. Любая подмена или потеря символов неизбежно искажает как само содержимое базы, так и контрольную сумму, вычисляемую в виде некоторой хэш-функции от полученного остатка.

Заключение

В ходе проведенного исследования решена актуальная научно-техническая задача разработки и математического обоснования криптографического метода организации многоуровневого иерархического доступа к пространственно-временным данным в графовых базах. Предложенный подход основывается на интеграции классической схемы иерархического распределения экспонент Акла-Тейлора и аппарата комбинаторной теории групп – свободных произведений с объединенной подгруппой, развивая криптографические принципы шифрования геоданных, заложенные в работе [11]. При этом разработана технология сериализации элементов графа ГИС (вершин, типов связей и их атрибутов) в единую монолитную строку («алгебраическое слово») с использованием замкнутого алфавита, сформированного из представителей левых смежных классов по объединяемой подгруппе. Предложенный способ исключает избыточность

хранения и необходимость множественной дублирующей рассылки пакетов данных пользователям с различными полномочиями. Кроме того, построена математическая модель динамического разграничения доступа, реализующая концепцию направленного «просачивания» (каскадного переноса) элементов объединенной подгруппы сквозь блоки данных под действием автоморфизмов при их приведении к нормальной форме. Доказано, что пользователь, обладающий набором из постоянных элементов объединяемой подгруппы, способен дешифровать любой целевой подграф, в то время как для пользователей промежуточных уровней СУБД генерирует контекстно-зависимые ключи, открывающие нижеуровневые базы при строгой изоляции от параллельных ветвей структуры прав. Описан механизм контроля целостности графа, основанный на определении остатка мультипликативной редукции на конце расшифрованной строки. Зафиксирован ярко выраженный лавинный эффект при модификации одного символа исходного текста, подтверждающий стойкость схемы к статистическому криптоанализу, что было продемонстрировано на конкретном примере, в котором в качестве множителей выбраны знакопеременные группы A_{14} с объединяемой подгруппой A_{12} и алфавитом, состоящим из 181 символа (т.к. $A_{14} : A_{12} = 182$, при этом тривиальный смежный класс исключается).

Благодарности

Автор благодарит ведущего инженера отдела геодезии ППК Роскадастр Дергилёву А.Э. за ценные советы при написании статьи, а также за помощь в проведении экспериментов и обработке данных.

Список литературы

1. Akl S.G., Taylor P.D. Cryptographic Solution to a Problem of Access Control in a Hierarchy. ACM Transactions on Computer Systems. 1983. Vol. 1. No. 3. Pp. 239-248. DOI: 10.1145/357369.357372.
2. Damiani M.L., Bertino E., Catania B., Perlasca P. GEO-RBAC: A spatially aware role-based access control model. ACM Transactions on Information and System Security (TISSEC). 2007. Vol. 10. No. 1. Art. 2. DOI: 10.1145/1210263.1210265.
3. Rodríguez-Aragón J.F., Zato C., Pinto-Santos F., Sánchez-Pravos L. KATENA: a verifiable governance architecture for encrypted cloud storage systems. Frontiers in Big Data. 2026. Vol. 9. Art. 1829960. DOI: 10.3389/fdata.2026.1829960.
4. Saxena U.R., Khatri Y., Kadel R., Shailendra S. A Service Recommendation Model in Cloud Environment Based on Trusted Graph-Based Collaborative Filtering Recommender System. Network. 2025. Vol. 5. No. 3. Art. 30. DOI: 10.3390/network5030030.
5. Celikbilek I., Celiktas B., Ozdemir E. A Hierarchical Key Assignment Scheme: A Unified Approach for Scalability and Efficiency. IEEE Access. 2024. Vol. 12. Pp. 70568-70580. DOI: 10.1109/ACCESS.2024.3401844.
6. Lee T. F., Wang J. G., Chen Y. C. Review of hierarchical database access control for E-medicine systems. Tzu Chi Medical Journal. 2023. Vol. 35. No. 2. Pp. 143-147. DOI: 10.4103/tcmj.tcmj_124_22.
7. Lal S., Kumar N. An efficient flexible hierarchical access control scheme enabling real-life exceptions. Sādhana. 2022. Vol. 47. Art. 20. DOI: 10.1007/s12046-021-01776-0.
8. Магнус В., Каррас А., Солигэр Д. Комбинаторная теория групп / Пер. с англ. Д.И. Молдаванского, А.А. Фридмана, Ю.И. Хмелевского; Под ред. М.Д. Гриндлингера. М.: Наука. 1974. 456 с.
9. Neumann P.M. (1973). The SQ-universality of some finitely presented groups. Journal of the Australian Mathematical Society, 16 (1). 1-6. DOI:10.1017/S1446788700013859.
10. Лоссов К.И. SQ-универсальность свободных произведений с конечными объединенными подгруппами // Сибирский математический журнал. 1986. Т. 27. №6. С. 128-139.
11. Лоссов К.И., Дергилёва А.Э. Защита потоков геоданных с использованием амальгам групп в качестве платформы шифрования // Геодезия и картография. 2026. №6. С. 24-32. DOI: 10.22389/0016-7126-2026-1032-6-24-32.

References

1. Akl S.G., Taylor P.D. Cryptographic Solution to a Problem of Access Control in a Hierarchy. ACM Transactions on Computer Systems. 1983. Vol. 1. No. 3. Pp. 239-248. DOI: 10.1145/357369.357372.
2. Damiani M.L., Bertino E., Catania B., Perlasca P. GEO-RBAC: A spatially aware role-based access control model. ACM Transactions on Information and System Security (TISSEC). 2007. Vol. 10. No. 1. Art. 2. DOI: 10.1145/1210263.1210265.
3. Rodríguez-Aragón J.F., Zato C., Pinto-Santos F., Sánchez-Pravos L. KATENA: a verifiable governance architecture for encrypted cloud storage systems. Frontiers in Big Data. 2026. Vol. 9. Art. 1829960. DOI: 10.3389/fdata.2026.1829960.

4. Saxena U.R., Khatri Y., Kadel R., Shailendra S. A Service Recommendation Model in Cloud Environment Based on Trusted Graph-Based Collaborative Filtering Recommender System. *Network*. 2025. Vol. 5. No. 3. Art. 30. DOI: 10.3390/network5030030.
5. Celikbilek I., Celikbas B., Ozdemir E. A Hierarchical Key Assignment Scheme: A Unified Approach for Scalability and Efficiency. *IEEE Access*. 2024. Vol. 12. Pp. 70568-70580. DOI: 10.1109/ACCESS.2024.3401844.
6. Lee T. F., Wang J. G., Chen Y. C. Review of hierarchical database access control for E-medicine systems. *Tzu Chi Medical Journal*. 2023. Vol. 35. No. 2. Pp. 143-147. DOI: 10.4103/tcmj.tcmj_124_22.
7. Lal S., Kumar N. An efficient flexible hierarchical access control scheme enabling real-life exceptions. *Sādhana*. 2022. Vol. 47. Art. 20. DOI: 10.1007/s12046-021-01776-0.
8. Magnus V., Karras A., Soliter D. *Kombinatornaya teoriya grupp* [Combinatorial group theory]. Translated from English by D.I. Moldavansky, A.A. Fridman, and Yu.I. Khmelevsky; edited by M.D. Grindlinger. Moscow: Nauka. 1974. 456 p.
9. Neumann P.M. (1973). The SQ-universality of some finitely presented groups. *Journal of the Australian Mathematical Society*, 16 (1). 1-6. DOI:10.1017/S1446788700013859.
10. Lossov K.I. SQ-universality of free products with finite amalgamated subgroups. *Sibirskii matematicheskii zhurnal*. 1986. V. 27. No. 6. Pp. 128-139.
11. Lossov K.I., Dergileva A.E. Protection of Geodata Streams Using Group Amalgams as an Encryption Platform. *Geodeziya i kartografiya*. 2026. No. 6. Pp. 24-32. DOI: 10.22389/0016-7126-2026-1032-6-24-32.

Сведения об авторах
Принадлежность к организации

Лоссов Константин Иванович

кандидат физико-математических наук, заведующий кафедрой Высшей математики, ФГБОУ ВО 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 26.08.2026 г.