Научная статья
УДК 004.056
DOI:10.31854/1813-324X-2022-8-4-119-129

Разработка схемы контроля доступа к данным на основе иерархии ролей с использованием постквантовых математических преобразований

Анастасия Викторовна Ярмак, yarmak.av@ibks.spbstu.ru

Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, 195251, Российская Федерация

Аннотация: *В работе представлена схема CSIDH-HRBAC, основанная на постквантовых математических преобразованиях и позволяющая реализовать контроль доступа к данным, располагающимся в недоверенной облачной инфраструктуре. CSIDH-HRBAC построена на базе ролевой модели управления доступом с поддержкой системы иерархии ролей. Предлагаемая схема подразумевает наличие доверенной стороны, осуществляющей управление криптографическими ключами, ассоциированными с пользователями, ролями, файлами. Приведено описание основных процедур, связанных с получением доступа к данным, лишением прав доступа, добавлением новых сущностей. Рассмотрены типовые сценарии атак на предложенную схему, в том числе подмена роли, сговор участников с целью вскрытия ключа родительской роли, попытка доступа к данным после отзыва роли у пользователя. Для оценки быстродействия криптографических операций выполнено моделирование ее работы при различных параметрах. Обсуждаются преимущества и ограничения схемы CSIDH-HRBAC. В частности, отмечается необходимость защиты от угроз со стороны администратора, перспектива применения квантово-устойчивых примитивов на основе задач теории решеток.*

Ключевые слова: *криптографический контроль доступа, эллиптические кривые, изогении, криптография*

Источник финансирования: Исследование выполнено при финансовой поддержке Минцифры России в рамках научного проекта № 12/21-к (грант ИБ).

Ссылка для цитирования: Ярмак А.В. Разработка схемы контроля доступа к данным на основе иерархии ролей с использованием постквантовых математических преобразований // Труды учебных заведений связи. 2022. Т. 8. № 4. С. 119–129. DOI:10.31854/1813-324X-2022-8-4-119-129

Post-Quantum Cryptographic Access Control Based on Hierarchical RBAC Model

Anastasya Yarmak, yarmak.av@ibks.spbstu.ru

Peter the Great St. Petersburg Polytechnic University,
St. Petersburg, 195251, Russian Federation

Abstract: *The paper considers the isogeny-based cryptographically enforced data access control scheme CSIDH-HRBAC for untrusted cloud. CSIDH-HRBAC is based on a role-based access control model with support for a role hierarchy system. The proposed scheme implies the presence of a trusted party that manages cryptographic keys associated with users, roles, files. The basic procedures for gaining access to data, revoking access rights, adding new entities and updating parameters are given. Typical scenarios of attacks on the proposed scheme are considered, including role substitution, collusion by participants to compute the parent role key, attempt to access data after role revocation from user. To evaluate the*

performance of cryptographic operations, the simulation of the basic procedures was performed. The advantages and limitations of the CSIDH-HRBAC scheme are discussed. In particular, the need for protection against threats from the administrator, the prospect of using lattice-based post-quantum cryptographic primitives is noted.

Keywords: *cryptographic access control, elliptic curves, isogeny, cryptography*

Funding: the reported study was funded by Ministry of Digital Development, Communications and Mass Media of the Russian Federation, project number 12/21-k (grant on Information Security).

For citation: Yarmak A. Post-Quantum Cryptographic Access Control Based on Hierarchical RBAC Model. *Proc. of Telecom. Universities.* 2022;8(4):119–129. (in Russ.) DOI:10.31854/1813-324X-2022-8-4-119-129

Введение

Современные крупномасштабные системы представляют собой класс сложных систем, состоящих из взаимодействующих территориально-распределенных компонентов, построенных на базе технологий Интернета вещей, промышленного Интернета вещей, беспроводных сенсорных сетей и т. п. Большое число узлов системы, их распределенность и избыточность, разнородность вычислительных возможностей и ролей устройств, динамическая топология сети – все эти аспекты накладывают ограничения на применение традиционных способов обеспечения кибербезопасности [1–3].

Ключевыми факторами в развитии систем управления крупномасштабными объектами такого типа представляются, во-первых, переход к групповой, многоадресной передаче данных, обеспечивающей возможность одновременной рассылки информации устройствам, входящим в некоторую группу (групповое взаимодействие узлов), и во-вторых, использование принципа многоуровневого, многослойного и стратифицированного описания архитектуры [4] с введенным отношением вертикальной соподчиненности подсистем.

С распространением облачных технологий многие организации перешли на парадигму аутсорсинга хранения и обработки данных. Появление новой сущности, отвечающей, в том числе, за соблюдение политики управления доступом, ведет к возникновению новых угроз безопасности, так как провайдер облачных услуг может просматривать данные независимо от заданных владельцем прав доступа.

Таким образом, актуальной является задача защиты информации от несанкционированного доступа при ее обработке и хранении в облачном хранилище. Данный аспект стал причиной возросшего интереса к концепции криптографического контроля доступа, которая позволяет обеспечить защиту хранимых в облаке данных с помощью криптографических преобразований, а разграничение доступа к информации осуществлять путем управления соответствующими криптографическими ключами.

1. Способы организации криптографического контроля доступа

Основная проблематика, возникающая при обеспечении контроля доступа к данным в облачном хранилище, включает в себя:

– защиту от угроз, связанных с недоверенным облачным провайдером;

– обеспечение динамического управления доступом, снижение вычислительных и временных затрат при изменении политики доступа к данным;

– учет специфики крупномасштабных систем при реализации управления доступом к данным, в частности иерархической структуры, большого количества пользователей.

Как показано на рисунке 1, в качестве характеристик схем криптографического контроля доступа можно указать следующие:

– возможность интеграции с классическими моделями разграничения доступа;

– используемая криптографическая схема;

– математический аппарат, а также вычислительно сложные задачи, положенные в основу безопасности криптографической схемы;

– рассматриваемые угрозы, от которых обеспечивается защита.

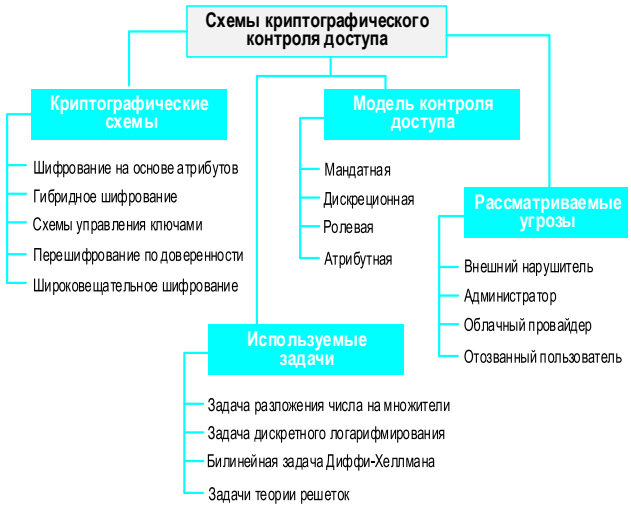


Рис. 1. Характеристики схем криптографического контроля доступа

Fig. 1. Parameters of Cryptographic Access Control Schemes

В настоящий момент известны решения, основанные на мандатной [5] и избирательной [6] моделях контроля доступа. Шифрование на основе атрибутов представляется «криптографическим аналогом» модели разграничения доступа на основе атрибутов [7]. Однако большинство схем построено на базе ролевой модели управления доступом [8–10], что обусловлено ее распространением в больших организациях и облачных платформах [11]. Поэтому представляется обоснованной разработка схемы криптографического контроля доступа, совместимая с моделью разграничения доступа на основе функционально-ролевых отношений.

С точки зрения стойкости рассматриваемых схем криптографического контроля доступа, большинство конструкций используют предположения о сложности задач разложения числа на множители, дискретного логарифмирования, билинейной задачи Диффи – Хеллмана. Данные задачи не позволяют строить квантово-устойчивые конструкции, т. е. обеспечить защиту от нарушителя, располагающего квантовой вычислительной моделью. Из работ, посвященных постквантовым схемам, известны основанные на решетках [12], однако они предполагают использование атрибутной модели контроля доступа.

Среди рассматриваемых решений на основе ролевой модели схема [10] обеспечивает защиту в том числе от угрозы со стороны администратора, что достигается за счет аппаратной технологии Intel-SGX, которая накладывает ограничения на используемую инфраструктуру. Решение Crypt-DAC [8] в этом смысле является более универсальным, однако предполагает доверие к администратору, осуществляющему управление криптографическими ключами. В качестве криптографической схемы используется гибридное шифрование, что позволяет реализовать ее с помощью существующей криптосистемы с открытым ключом. Однако в Crypt-DAC не используются постквантовые математические преобразования.

Для учета специфики крупномасштабных систем в качестве основы можно использовать модификацию ролевой политики разграничения доступа, предусматривающую иерархическую систему ролей и являющейся наиболее близкой к реальным организационно-технологическим системам [13]. Предлагаемая схема CSIDH-HRBAC криптографического контроля доступа является развитием работы [14], где в качестве математического аппарата использовались изогении эллиптических кривых. Несмотря на то, что изогении позволяют строить гибкие протоколы постквантовой криптографии, ориентированные на применение в масштабируемых и динамических средах, в работе [14] используется SIDH-конструкция, которая на сегодняшний день считается уязвимой, а также не учитывает иерархию ролей, как и Crypt-DAC [8], положенная в ее основу. Таким образом, основное отличие схемы CSIDH-HRBAC состоит в использовании квантово-устойчивой CSIDH-конструкции на изогениях, а также учета отношения иерархии на множестве ролей.

2. Предлагаемая схема криптографического контроля доступа CSIDH-HRBAC

Эллиптические кривые являются объектом изучения алгебраической геометрии, нашедшие свое прикладное применение при построении асимметричных криптографических систем, в том числе постквантовой криптографии.

Изогения $\varphi: E_1(K) \to E_2(K)$ эллиптических кривых E_1 и E_2 – это рациональное отображение, сохраняющее неподвижной бесконечно удаленную точку: $\varphi(P_\infty) = P_\infty$ [15]:

$$\varphi(x, y) = (r_1(x), r_2(x)y),$$

где $r_1(x) = \frac{p_1(x)}{q_1(x)}, r_2(x) = \frac{p_2(x)}{q_2(x)}$ – рациональные функции с взаимно простым числителем и знаменателем; $\deg\varphi = \max\{\deg p_1(x), \deg q_1(x)\}$ – степень изогении.

Задача поиска изогении $\varphi: E_1 \to E_2$ подразумевает вычисление идеала $\alpha \in Cl(O_D)$ из группы классов идеалов квадратичного порядка O_D, для которого выполняется условие: $E_2 = [\alpha] * E_1$, где E_1 и E_2 – известные эллиптические кривые с $\mathrm{End}(E_1) \cong \cong O_D \cong \mathrm{End}(E_2)$.

В общем случае, криптосистемы с открытым ключом на изогениях эллиптических кривых можно разделить на три семейства – CRS-, SIDH-, CSIDH-схемы [16]. Преимущества представителей последних (высокая скорость вычислений, т. к. используемые степени изогении фиксированы и, как правило, равны 2 и 3, небольшая по сравнению с другими постквантовыми криптосистемами длина параметров) обусловили их широкое применение при построении криптосистем на изогениях.

Однако, как показано в таблице 1, в настоящий момент такой подход не рекомендуется использовать на практике, так как атака, опубликованная в работах [17–18], позволяет решить задачу поиска изогений за полиномиальное время. В основу атаки легла теорема Кани, которую возможно применить за счет особенностей SIDH-схем, в которых между абонентами пересылаются вспомогательные точки (образы точек, являющихся образующими подгруппы группы кручения), а также известна степень секретной изогении. CRS- и CSIDH-схемы построены по другим принципам, т. е. данная атака для них не применима. Поэтому наиболее перспективным представляется подход на основе CSIDH [19], который обеспечивает субэкспоненциальный уровень сложности решения задач поиска изогений и, в то

же время, позволяет уменьшить время вычисления изогении и сократить длину параметров по сравнению с CRS-схемами.

ТАБЛИЦА 1. Сложность решения задачи поиска изогении для различных конструкций

TABLE 1. Complexity of Isogeny Problem for Different Schemes

Сложность решения задачи поиска изогении			
	CRS-схемы	SIDH-схемы	CSIDH-схемы
A	$O(p^{1/4})$	$O(l^8 \log N + \log^2 N)$	$O\left(p^{\frac{1}{4}}\right)$
B	$L_p[\frac{1}{2}, \sqrt{2}]$	—	$L_p[\frac{1}{2}, \sqrt{2}]$

Условные обозначения: A – классическая вычислительная модель; B – квантовая вычислительная модель; p – характеристика поля; N – степень изогении; l – наибольший простой делитель N.

Пусть $HRBAC = < U, R, P, C, F_{PR}, F_{UR}, F_{RR} >$ – модель контроля доступа с иерархической системой ролей, где:

U – множество пользователей;

R – множество ролей;

$P \subseteq \text{Files} \times OP$ – множество прав доступа;

Files – множество файлов;

$OP = \{\text{read}, \text{write}\}$ – операции над файлами;

C – множество сеансов работы пользователей с системой;

$F_{PR} \subseteq P \times R$ – отношение, задающее связь роли и соответствующих прав;

$F_{UR} \subseteq U \times R$ – отношение, определяющее связь пользователя и роли;

$F_{RR} = \{(r_j, r_i) | r_j, r_i \in R \text{ и } r_j \leq r_i\}$ – отношение частичного порядка на множестве R, определяющее иерархию ролей (если справедливо $r_j \leq r_i$, то роль r_i наследует все права доступа роли r_j).

Также вводятся следующие вспомогательные функции [13], необходимые для управления доступом в системе:

– $\text{f}_{\text{user}}: C \to U$, сопоставляющая сеансу работы с системой пользователя, осуществляющего данный сеанс;

– $\text{f}_{\text{roles}}: C \to R$, сопоставляющая набор ролей, доступных пользователю, осуществляющему данный сеанс, с учетом иерархически подчиненных ролей.

Также подразумевается использование следующих криптографических алгоритмов [8]:

– $\text{Enc}_k^{\text{sym}}/\text{Dec}_k^{\text{sym}}$ – алгоритмы зашифрования и расшифрования данных на ключе k с помощью симметричного шифра;

– $\text{Enc}_{ek}^{\text{pub}}/\text{Dec}_{dk}^{\text{pub}}$ – алгоритмы зашифрования данных на открытом ключе pk и расшифрования данных на закрытом ключе dk с помощью асимметричного шифра;

– $\text{Sign}_{sk}/\text{Verify}_{vk}$ – алгоритмы формирования подписи и проверки подписи с помощью ключей sk и vk соответственно.

В качестве криптосистемы с открытым ключом, необходимой для шифрования ключа dk_r роли r и ключа k_f, предлагается использовать схему шифрования Эль-Гамаля на изогениях. В качестве протокола подписи используется подпись CSI-FiSh [20].

Криптографическими ключами управляет администратор организации в соответствии с ролевой политикой доступа. Криптографический контроль доступа осуществляется путем назначения сущностям ролевой политики доступа соответствующих криптографических ключей, а также генерации кортежей, определяющих связь между этими сущностями. Их описание представлено в таблице 2. Облачный провайдер отвечает за хранение файлов пользователей. Предполагается, что перед загрузкой данных в облако все файлы шифруются. Пользователи могут осуществлять загрузку данных из облака, а также вносить изменения в файлы согласно заданным для их роли правам.

ТАБЛИЦА 2. Криптографические ключи сущностей в схеме CSIDH-HRBAC

TABLE 2. Cryptographic Keys for CSIDH-HRBAC

Сущность	Криптографические ключи	Описание
Пользователь $u \in U$	1) Пара ключей (ek_u, dk_u) для чтения файлов 2) Пара ключей (sk_u, vk_u) для записи в файлы	Необходимы для чтения и записи в файлы, хранящиеся у облачного провайдера
Роль $r \in R$	Пара ключей (ek_r, dk_r) для доступа к файлам	Определяют доступ к файлам для заданной роли
Файл $f \in F$	Каждому файлу f соответствует ключ k_f симметричного алгоритма	Используется для шифрования файлов, хранящихся в облаке
Кортеж $fk_{rf} \in FK$ (связь ключа роли и прав доступа)	Если роли r разрешена операция op над файлом f, то существует кортеж: $fk_{rf} = \left(r, (f, op), \text{Enc}_{ek_r}^{\text{pub}}(k_f)\right)$	Ключ роли и права доступа связаны через кортежи из FK. В каждом кортеже хранится симметричный ключ k_f, зашифрованный на открытом ключе ek_r роли r
Кортеж $rk_{ur} \in RK$ (связь ключа пользователя и ключа роли)	Если пользователю u была выдана роль r, то существует кортеж: $rk_{ur} = \left(u, r, \text{Enc}_{ek_u}^{\text{pub}}(dk_r)\right)$	Ключи пользователя и роли связаны через кортежи из RK. В таком кортеже хранится закрытый ключ dk_r роли r, зашифрованный на открытом ключе ek_u пользователя u
Кортеж $rr_{ij} \in RR$ (связь ключей ролей)	Если r_j – прямой потомок роли r_i, то существует кортеж: $rr_{ij} = \left(E_{rj}^{id}, y_{ij}\right)$	Открытая информация, необходимая для вычисления ключа подчиненной роли

В рамках предлагаемой схемы криптографического контроля доступа поддерживаются операции генерации ключей, получения доступа на чтение и запись, добавления новых сущностей, смены ключей, лишения прав доступа.

2.1. Инициализация параметров схемы криптографического контроля доступа CSIDH-HRBAC

Для генерации параметров схемы администратору необходимо выполнить следующие действия.

1) Сгенерировать характеристику поля: $p = 4l_1l_2 \dots l_m \pm 1$, $l_1, l_2, \dots, l_n$ – малые простые.

2) Сгенерировать суперсингулярную эллиптическую кривую $E_0(\mathbb{F}_p): y^2 = x^3 + x$ с числом точек $\#E_0(\mathbb{F}_p) = p + 1 = 4l_1l_2 \dots l_n$ и кольцом эндоморфизмов $\mathrm{End}(E_0) \cong O_D$.

Группа классов $Cl(O_D) =< \mathfrak{g} >$ – циклическая, $\#Cl(O_D) = N$.

3) Сгенерировать случайное значение $\alpha_{pub} \leftarrow \mathbb{Z}_N$ и вычислить идеал $[\alpha_{pub}] = \mathfrak{g}^{\alpha_{pub}}$. Определить изогению $\varphi_0^{pub}: E_0 \to E_0^{pub}, E_0^{pub} = [\alpha_{pub}] * E_0$.

4) Сгенерировать случайное значение $\alpha_{sign} \leftarrow \mathbb{Z}_N$ и вычислить идеал $[\alpha_{sign}] = \mathfrak{g}^{\alpha_{sign}}$. Определить изогению $\varphi_0^{sign}: E_0 \to E_0^{sign}, E_0^{sign} = [\alpha_{sign}] * E_0$.

5) Выбрать параметры S, t для протокола цифровой подписи CSI-FiSh на изогениях.

6) Сгенерировать случайное значение $\alpha_{roles} \leftarrow \mathbb{Z}_N$ и вычислить идеал $[\alpha_{roles}] = \mathfrak{g}^{\alpha_{roles}}$. Определить изогению $\varphi_0^{roles}: E_0 \to E_0^{roles}, E_0^{roles} = [\alpha_{roles}] * E_0$.

7) Сгенерировать случайное значение $\alpha_{id} \leftarrow \mathbb{Z}_N$ и вычислить идеал $[\alpha_{id}] = \mathfrak{g}^{\alpha_{id}}$. Определить изогению $\varphi_0^{id}: E_0 \to E_0^{id}, E_0^{id} = [\alpha_{id}] * E_0$. Для каждой роли $r_i \in R$ вычислить кривую-идентификатор роли: $\varphi_{r_i}^{id}: E_0^{id} \to E_{r_i}^{pub}, E_{r_i}^{id} = [\alpha_{r_i}^{id}] * E_0^{id}$ путем задания случайного идеала $\alpha_{r_i}^{id}$.

8) Определить криптографическую хэш-функцию $H: \{0,1\}^* \to \{0,1\}^\rho$.

Открытыми параметрами схемы CSIDH-HRBAC являются: $p, \{l_i\}, E_0, E_0^{pub}, E_0^{sign}, E_0^{id}, \{E_{r_j}^{id}\}, S, t, E_0^{roles}, H$.

2.2. Генерация ключей

Генерация ключей k_f, используемых для шифрования файлов, выполняется согласно требованиям, заданным для выбранного симметричного алгоритма шифрования данных $\mathrm{Enc}_k^{\mathrm{sym}}/\mathrm{Dec}_k^{\mathrm{sym}}$.

Генерация ключей пользователей для чтения файлов

Для генерации ключей (ek_u, dk_u) пользователя $u \in U$ администратору необходимо выполнить следующие этапы.

Этап 1. Генерация секретной изогении. Выбрать случайное значение $g_u \leftarrow \mathbb{Z}_N$ и вычислить секретную изогению $\varphi_u^{pub}: E_0^{pub} \to E_u^{pub}$, $E_u^{pub} = [g_u] * E_0^{pub}$.

Этап 2. Назначение ключа пользователю. Установить для u пару (ek_u, dk_u) открытый ключ/закрытый ключ следующим образом:

$$ek_u = E_u^{pub}, dk_u = g_u.$$

Генерация ключей пользователей для записи в файлы

Для генерации ключей (sk_u, vk_u) пользователя $u \in U$ администратору необходимо выполнить процедуру согласно [20].

1) Для $i = 1, \dots, S-1$:

– сгенерировать случайный идеал $[g_u^{(i)}]$;

– вычислить изогенную кривую:

$$E_{u,i}^{sign} = [g_u^{(i)}] * E_0^{sign}.$$

2) Установить для u пару (sk_u, vk_u) ключ подписи и ключ проверки подписи следующим образом:

$$sk_u = ([g_u^{(1)}], \dots, [g_u^{(S-1)}]), vk_u = (E_{u,1}^{sign}, \dots, E_{u,S-1}^{sign}).$$

Генерация ключей роли

Иерархия ролей предполагает возможность вычисления закрытого ключа dk_{r_j} роли r_j пользователями с ролью r_i, если $r_j \le r_i$.

Самый простой способ решения данной задачи – хранение для каждого пользователя u_k с ролью r_i помимо кортежа:

$$rk_{u_k r_i} = \left(u_k, r_i, \mathrm{Enc}_{ek_{u_k}}^{\mathrm{pub}}(dk_{r_i})\right)$$

кортежей:

$$rk_{u_k r_j} = \left(u_k, r_j, \mathrm{Enc}_{ek_{u_k}}^{\mathrm{pub}}\left(dk_{r_j}\right)\right) \text{ для } \forall r_j \le r_i.$$

Однако это требует значительного объема памяти и осложняет отзыв ключа роли, имеющей много подчиненных ролей.

Для этого предлагается модифицировать схему управления ключами [21], что позволит пользователям роли r_i на основе своего ключа dk_{r_i} и вспомогательной информации вычислять ключ dk_{r_j} подчиненной роли r_j. Для реализации данного механизма можно использовать связь секретной изогении родительской роли с кривой-идентификатором подчиненной роли. Тогда для генерации ключей (ek_{r_i}, dk_{r_i}) роли $r_i \in R$ администратору необходимо реализовать этапы.

Этап 1. Генерация ключа роли. Для роли r_i выбрать случайное значение $g_{r_i} \leftarrow \mathbb{Z}_N$ и положить $(ek_{r_i}, dk_{r_i}) \leftarrow (E_{r_i}, g_{r_i})$, где $\varphi_{r_i}: E_0^{roles} \to E_{r_i}$, $E_{r_i} = [g_{r_i}] * E_0^{roles}$.

Этап 2. Задание связи с ключами подчиненных ролей. Для каждой роли r_j: $r_j \le r_i$, являющейся прямым потомком роли r_i, вычислить открытое значение: $y_{ij} = dk_{r_j} \oplus H\left(j(\varphi_{r_i}(E_{r_j}^{id}))\right)$, где $j(E)$ – это j-инвариант кривой E; φ_{r_i} – изогения, связывающая открытый ключ роли r_j с кривой-идентификатором роли r_i, т. е. $\varphi_{r_i}\left(E_{r_j}^{id}\right) = [g_{r_i}] * E_{r_j}^{id}$, как показано на рисунке 2.

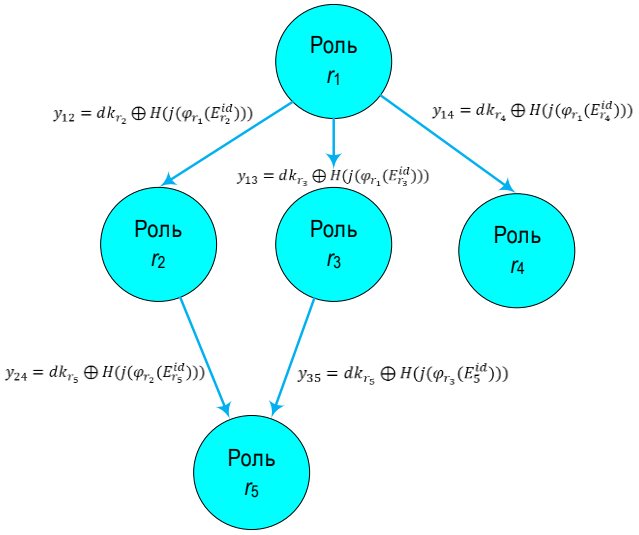


Рис. 2. Иерархия ролей и связь криптографических ключей

Fig. 2. Role-Key Hierarchy in CSIDH-HRBAC

2.3. Получение доступа на чтение

Если пользователь u с ролью r_i хочет получить доступ к файлу f, ему необходимо выполнить следующие шаги.

Шаг 1. Загрузить $rk_{ur_i} = \left(u, r_i, \mathrm{Enc}_{ek_u}^{\mathrm{pub}}\left(dk_{r_i}\right)\right)$ кортеж, содержащий зашифрованный на ключе ek_u пользователя u закрытый ключ dk_{r_i} роли r_i.

Шаг 2. Вычислить $dk_{r_i} \leftarrow \mathrm{Dec}_{dk_u}^{\mathrm{pub}}\left(dk_{r_i}\right)$.

Шаг 3. Если $\left((f, \mathrm{read}), r_i\right) \in F_{PR}$, то загрузить $fk_{r_i f} = \left(r_i,\ (f, \mathrm{read}), \mathrm{Enc}_{ek_{r_i}}^{\mathrm{pub}}\left(k_f\right)\right)$.

Шаг 4. Вычислить $k_f \leftarrow \mathrm{Dec}_{dk_{r_i}}^{\mathrm{pub}}\left(k_f\right)$ и перейти на шаг 8.

Шаг 5. Если $\exists r_j: \left((f, \mathrm{read}), r_j\right) \in F_{PR} \wedge r_j \le r_i$, то, пока $r_i \ne r_j$, выполнять:

– загрузить $rr_{ik} = \left(E_{r_k}^{id}, y_{ik}\right)$, где r_k – прямой потомок r_i;

– вычислить $dk_{r_k} = y_{ik} \oplus H\left(j(\varphi_{r_i}(E_{r_k}^{id}))\right)$;

– положить $r_i \leftarrow r_k$.

Шаг 6. Загрузить $fk_{r_j f} = \left(r_j,\ (f, \mathrm{read}), \mathrm{Enc}_{ek_{r_j}}^{\mathrm{pub}}\left(k_f\right)\right)$.

Шаг 7. Расшифровать ключ k_f, используя закрытый ключ роли dk_{r_j}.

Шаг 8. Загрузить зашифрованный файл:

$$F = \mathrm{Enc}_{k_f}^{\mathrm{sym}}(f).$$

Шаг 9. Расшифровать файл $f \leftarrow \mathrm{Dec}_{k_f}^{\mathrm{sym}}(F)$.

Рисунок 3 иллюстрирует последовательность действий, необходимых для получения доступа на чтение.

2.4. Получение доступа на запись

Если пользователь u с ролью r_i хочет осуществить запись в файл f, ему необходимо выполнить следующие действия.

1) Выполнить шаги 1–9 алгоритма получения доступа на чтение (п. 2.3) относительно прав (f, write).

2) Загрузить в облачное хранилище зашифрованный файл $F' = \mathrm{Enc}_{k_f}^{\mathrm{sym}}(f')$, где f' – это новое содержимое файла f.

3) Сформировать подпись $\sigma = (b_1, \ldots, b_t, c_1, \ldots, c_t)$ файла F' согласно алгоритму Sign_{sk_u} протокола CSi-FiSh.

4) Отправить значение σ администратору.

Для верификации действий пользователя администратор проверяет, что для:

$$u = \mathrm{f}_{\mathrm{user}}(c), r = \mathrm{f}_{\mathrm{roles}}(c): \exists (u, r) \in F_{UR} \wedge (r, p = (f, \mathrm{write})) \in F_{PR};$$

В случае успешного выполнения администратор запускает алгоритм Verify_{vk_u} проверки подписи файла F' согласно протоколу CSi-FiSh. Если подпись верна, то файл F заменяется файлом F'. В противном случае файл F' удаляется.

2.5. Лишение прав доступа

При отзыве у пользователя u роли r необходимо выполнить генерацию новых ключей роли и шифрования файлов, а также обновить кортежи *RK* и *FK*. Для этого требуется выполнить следующие шаги.

Шаг 1. Сгенерировать новую ключевую пару для роли (ek'_r, dk'_r).

Шаг 2. Удалить кортеж $rk_{ur} = \left(u, r, \mathrm{Enc}_{ek_u}^{\mathrm{pub}}(dk_r)\right)$, связывающий пользователя u и роль r.

Шаг 3. Для каждого пользователя $u' \ne u$ с ролью r выполнить замену значений в кортеже rk_{ur} с $\left(u', r, \mathrm{Enc}_{ek_u}^{\mathrm{pub}}(dk_r)\right)$ на $\left(u', r, \mathrm{Enc}_{ek_{u'}}^{\mathrm{pub}}(dk'_r)\right)$.

Шаг 4. Для каждого файла f, доступного для r:

– расшифровать файл f и зашифровать его с помощью нового ключа k'_f;

– выполнить замену значений в кортеже fk_{rf} с $\left(r,\ (f, \mathrm{op}), \mathrm{Enc}_{ek_r}^{\mathrm{pub}}\left(k_f\right)\right)$ на $\left(r,\ (f, \mathrm{op}), \mathrm{Enc}_{ek'_r}^{\mathrm{pub}}\left(k'_f\right)\right)$.

Шаг 5. Выполнить замену значений в таблице *RR* для всех ролей, связанных отношением подчиненности или доминирования, используя новую кривую-идентификатор $\tilde{E}_r^{id}$ и обновленные значения $dk_{r_j} \oplus H\left(j(\varphi'_r(E_{r_j}^{id}))\right)$ для всех r_j, являющихся потомками r, и $dk'_r \oplus H\left(j(\varphi_{r_i}(\tilde{E}_r^{id}))\right)$ для всех r_i, являющихся родительскими для роли r.

Процедура отзыва ключа роли выполняется аналогичным образом, за исключением шага 2, связанного с удалением кортежа, ассоциированного с конкретным пользователем. Добавление новых сущностей и обновление ключей сводятся к операциям генерации новых криптографических ключей и замене или добавлению соответствующих значений в таблицы *RK*, *FK*, *RR*.

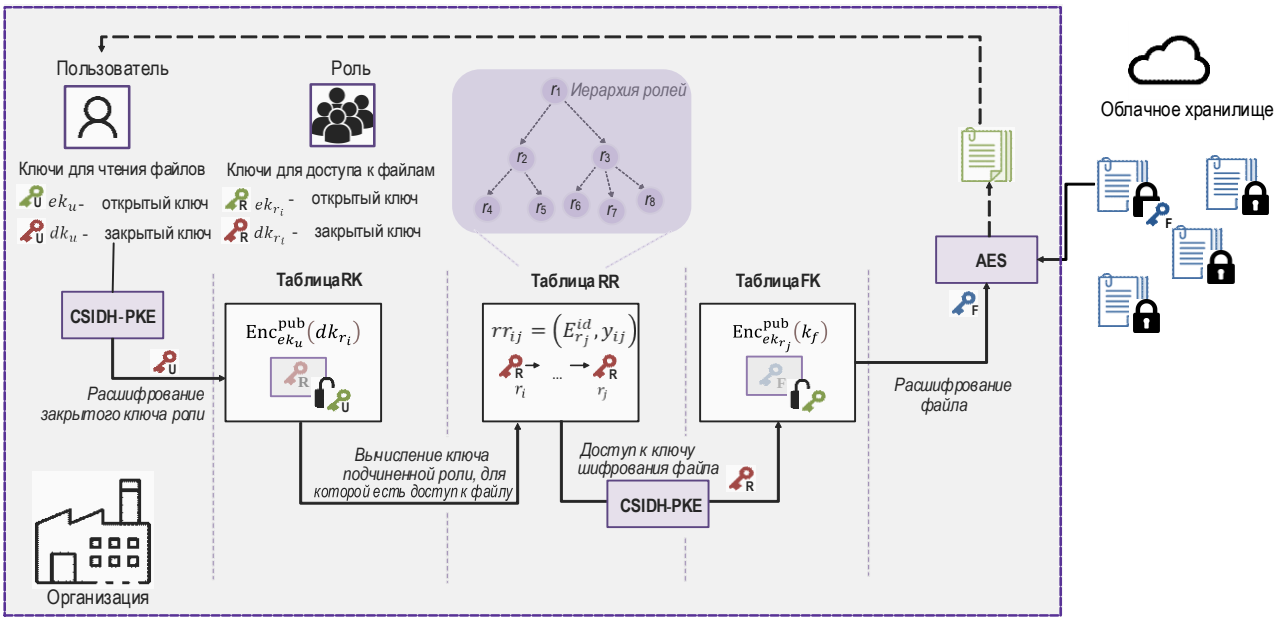


Рис. 3. Получение доступа на чтение

Fig. 3. Read Access in CSIDH-HRBAC

3. Анализ схемы CSIDH-HRBAC

Анализ безопасности включал в себя рассмотрение возможных сценариев атак на предлагаемую схему криптографического контроля доступа.

Сценарий 1. Утечка данных из облака. При реализации данного сценария нарушителю становятся доступны файлы, расположенные в облачном хранилище. В этом случае задача получения доступа к содержимому некоторого файла F сводится к задаче дешифрования шифртекста, т. е. взлома симметричного алгоритма шифрования. Если нарушитель имеет доступ к таблицам *RK*, *FK*, *RR*, то есть знает шифртекст $\mathrm{Enc}_{ek_r}^{\mathrm{pub}}(k_f)$ из кортежа fk_{rf}, то для получения ключа шифрования файла ему необходимо располагать закрытым ключом dk_r роли r или закрытым ключом dk_u пользователя u, имеющего эту роль. При отсутствии знания о ключах нарушителю необходимо решить задачу поиска изогении φ_u^{pub} или φ_r, которая считается вычислительно сложной.

Сценарий 2. Подмена роли. В данном сценарии нарушителем является пользователь системы с ролью r_j, его цель – выполнить операцию чтения файла или записи в файл, которая доступна для пользователей некоторой роли r_i. Если роли связаны отношением $r_j \le r_i$, то данное действие подразумевает вычисление закрытого ключа родительской роли. Зная ключи своей роли $\left(ek_{r_j}, dk_{r_j}\right)$ и значение $y_{ij} = dk_{r_j} \oplus H\left(j(\varphi_{r_i}(E_{r_j}^{id}))\right)$, нарушитель может найти значение $H\left(j(\varphi_{r_i}(E_{r_j}^{id}))\right)$, однако для нахождения секретной изогении φ_{r_i} (которая соответствует закрытому ключу роли r_i) ему необходимо решить задачу обращения криптографической хэш-функции H, т. е. найти кривую $\varphi_{r_i}\left(E_{r_j}^{id}\right)$, после чего вскрытие закрытого ключа роли r_i сводится к задаче поиска изогении между кривыми $\varphi_{r_i}\left(E_{r_j}^{id}\right)$ и $E_{r_j}^{id}$, что, с учетом предположений, имеет субэкспоненциальную сложность для квантового компьютера.

Сценарий 3. Повышение привилегий. Данный сценарий подразумевает попытку пользователя системы выполнить запись в файл, доступный ему только для чтения. В таком случае данное действие будет обнаружено администратором путем анализа политики разграничения прав доступа. Поэтому для прохождения проверки нарушителю необходимо модифицировать соответствующий кортеж в таблице *FK*.

Сценарий 4. Сговор участников. Пусть имеется сговор пользователей с ролями r_2, r_3, подчиняющимися одной родительской роли r_1, с целью восстановления ключа dk_{r_1} этой роли. Обладая общеизвестной информацией $rr_{12} = \left(E_{r_2}^{id}, y_{12}\right)$, $rr_{13} = \left(E_{r_3}^{id}, y_{13}\right)$ из таблицы *RR*, они могут восстановить значения $H\left(j(\varphi_{r_1}(E_{r_2}^{id}))\right)$ и $H\left(j(\varphi_{r_1}(E_{r_3}^{id}))\right)$. Тогда вскрытие ключа сводится к реализации сценария 2.

Сценарий 5. Попытка доступа к данным после отзыва участника. Пусть пользователь u пытается получить доступ к файлу f после отзыва у него роли r. Реализация данного сценария аналогична сценарию 1, так как в процессе отзыва участника осуществляется перегенерация ключей роли и ключей шифрования данных.

Для моделирования работы схемы использовалась библиотека [22], реализующая базовые операции вычисления изогений, а также позволяющая вычислить подпись на изогениях. В качестве кривых используются кривые в форме Монтгомери, поэтому для задания эллиптической кривой достаточно указать коэффициент A в уравнении кривой $E_0(\mathbb{F}_p): y^2 = x^3 + Ax^2 + x$. Это позволяет избавиться от процедуры вычисления j-инварианта кривой при подсчете элементов из таблицы *RR*.

Характеристика поля имеет длину 512 бит и задается следующим образом: $p = 4 \cdot 3 \cdot 5 * 7 \cdot \ldots \cdot 587 - 1$. Длина ключей (ek_u, dk_u) для чтения составляет 64 и 32 байта, соответственно, ключи (sk_u, vk_u) для записи в файлы имеют длину 16 байт и 16 Кб. В качестве хэш-функции использовался алгоритм Keccak-256, симметричный шифр – AES-256. В таблице 3 представлены результаты тестировании процедуры инициализации параметров схемы. Время генерации параметров зависит от числа ролей, определяемых ролевой политикой доступа, так как для каждой роли необходимо сгенерировать кривую-идентификатор роли. Данный этап может быть распараллелен путем использования нескольких вычислителей, каждый из которых будет взаимодействовать с своим пулом ролей.

Для добавления пользователя в систему необходимо выполнить генерацию ключей для чтения и записи в файл. При этом самым трудоемким этапом является генерация пары ключей для протокола подписи на изогениях. Время, необходимое для данных вычислений, определяется параметром S: при малом значении S процедура генерации параметров осуществляется быстрее, длина открытого ключа минимальна (128 байт), однако время формирования и проверки подписи значительно увеличивается.

При большом значении $S = 2^{15}$ время генерации может достигать нескольких минут, размер открытого ключа равен 2 Мб, но процедуры подписи сообщения и проверки подписи выполняются быстрее. При моделировании было выбрано значение $S = 2^8$, при котором общее время добавления пользователя в систему в среднем равно 9,97 с.

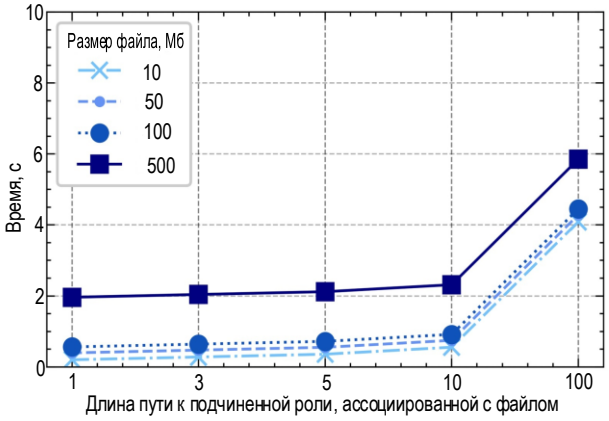
Операции назначения пользователю роли и предоставления доступа к файлу подразумевают зашифрование закрытого ключа роли и ключа шифрования файла с помощью схемы на изогениях. Среднее время выполнения этих операций равно 81 мс.

ТАБЛИЦА 3. Время инициализация параметров схемы

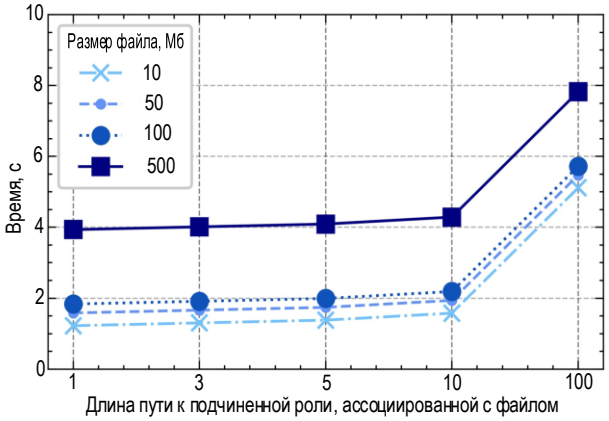
TABLE 3. Scheme's Parameters Initialization Time

Число ролей	Время, с
10	0, 53
100	3, 91
500	18, 43
1000	39, 24

На рисунках 4 и 5 представлены результаты тестирования, позволяющие оценить время, затрачиваемое только на криптографические операции. В частности, использовалось предположение о том, что каждой роли соответствует 10 пользователей, при этом число связей с другими ролями равно 3.



a)



b)

Рис. 4. Результаты тестирования процедур: a) получение доступа на чтение; b) получение доступа на запись в файл

Fig 4. Time for Read Access (a) and Write Access Operations (b)

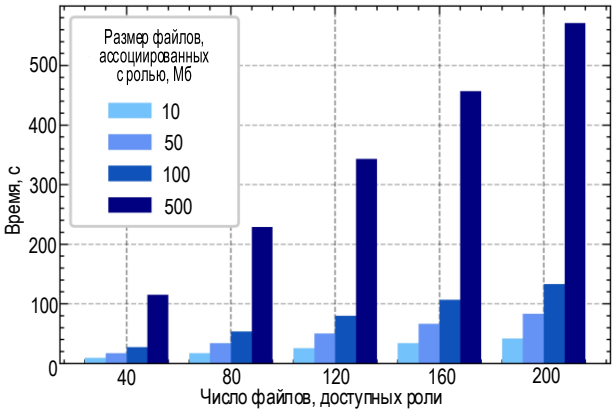


Рис. 5. Результаты тестирования процедуры отзыва роли у пользователя

Fig 5. Time for Revoking Role from User

Как видно из рисунка 4, время выполнения процедуры записи в файл отличается от получения доступа на чтение, что объясняется необходимостью формирования и проверки подписи для верификации вносимых изменений в файл. При доступе к файлу, ассоциированному с подчиненной ролью, пользователю необходимо вычислить закрытый ключ роли. Можно заметить, что данная операция незначительно влияет на скорость выполнения, так как для нахождения ключа роли необходимо рассчитать коэффициент изогенной кривой, найти значение хэш-функции и выполнить операцию XOR.

При лишении пользователя роли самой затратной, с точки зрения вычислений, является операция обновления ключей шифрования файлов, ассоциированных с данной ролью (т. е. тех, к которым разрешен доступ пользователям, имеющим эту роль). Одним из способов оптимизации является использование повторного шифрования на новом ключе (добавление нового слоя шифрования) вместо операции расшифрования файла и его зашифрования на новом ключе, но в таком случае пользователям необходимо хранить последовательность ключей.

Заключение

В работе представлена схема криптографического контроля доступа CSIDH-HRBAC, основанная на конструкции Crypt-DAC и применимая для систем, использующих ролевую политику управления доступом к данным. Предложенная схема отличается применением постквантовых математических преобразований для защиты информации, размещенной в недоверенном облачном хранилище, а также с учетом иерархии ролей, который актуален для крупномасштабных систем, имеющих ярко выраженную иерархическую структуру узлов (например, промышленный Интернет вещей, Интернет вещей, интеллектуальные энергосети и др.). Контроль доступа к данным осуществляется за счет использования криптографических алгоритмов и назначения соответствующих ключей, что обеспечивает легкость интеграции.

К преимуществам схемы CSIDH-HRBAC можно отнести отсутствие необходимости в создании и хранении дополнительных кортежей в таблицах *FK*, *RK* в случае наличия иерархии ролей (по сравнению с Crypt-DAC), учет специфики иерархических систем, а также использование квантово-устойчивых примитивов. В качестве ограничений можно отметить следующее: набор прав доступа старших ролей включает только те права, которые не вошли в набор подчиненных ролей; пользователь может принадлежать только одной роли; требуется хранить таблицу *RR*; необходимо доверие к администратору, осуществляющему управление криптографическими ключами пользователей.

Дальнейшее направление исследования связано с оптимизацией схемы, в частности, процедур, связанных с лишением прав доступа, а также обеспечением защиты от угрозы со стороны администратора. Также в качестве отдельного направления работы можно отметить использование других квантово-устойчивых примитивов, например, использующих задачи теории решеток, где для задания связи между ключами ролей можно использовать механизм делегирования базиса решетки [23]. В частности, в качестве шифрования можно использовать схему CRYSTALS-Kyber, в качестве цифровой подписи – CRYSTALS-Dilithium, которые являются финалистами конкурса Национального института стандартов и технологий на постквантовые стандарты.

Список источников

1. Krundyshev V., Kalinin M. The Security Risk Analysis Methodology for Smart Network Environments // Proceedings of the International Russian Automation Conference (RusAutoCon, Sochi, Russia, 06–12 September 2020). IEEE, 2020. PP. 437–442. DOI:10.1109/RusAutoCon49822.2020.9208116

2. Ovasapyan T., Moskvin D., Tsvetkov A. Detection of attacks on the Internet of Things based on intelligent analysis of devices functioning indicators // Proceedings of the 13th International Conference on Security of Information and Networks (SIN, Merkez Turkey, 4–7 November 2020). New York: Association for Computing Machinery, 2020. P. 3. DOI: 10.1145/3433174.3433611

3. Александрова Е.Б., Облогина А.Ю., Шкоркина Е.Н. Аутентификация управляющих устройств в сети Интернета вещей с архитектурой граничных вычислений // Проблемы информационной безопасности. Компьютерные системы. 2021. № 2. С. 82–88.

4. Мако Д., Месарович М., Такахара И. Теория иерархических многоуровневых систем. М.: Мир. 1973.

5. Горковенко Е.В. Применение нетрадиционных криптографических преобразований в системах с мандатной политикой управления доступом к информации // Известия Южного федерального университета. Технические науки. 2008. № 8(85). С. 135–141.
6. Di Vimercati S.D.C., Foresti S., Jajodia S., Paraboschi S., Samarati P. Over-encryption: Management of Access Control Evolution on Outsourced Data // Proceedings of the 33rd International Conference on Very Large Data Bases (VLDB, Vienna Austria, 23–27 September 2007). VLDB Endowment Inc., 2007. PP. 123–134.
7. Epishkina A., Zapechnikov S. On Attribute-Based Encryption for Access Control to Multidimensional Data Structures // Proceedings of the First International Early Research Career Enhancement School on Biologically Inspired Cognitive Architectures (BICA) for Young Scientist and Cybersecurity (FIERCES 2017, Moscow, Russia, 1–3 August 2017). Advances in Intelligent Systems and Computing. Vol. 636. Cham: Springer, 2017. PP. 251–256. DOI:10.1007/978-3-319-63940-6_36
8. Qi S., Zheng Y. Crypt-DAC: Cryptographically Enforced Dynamic Access Control in the Cloud // IEEE Transactions on Dependable and Secure Computing. 2019. Vol. 18. Iss. 2. PP. 765–779. DOI:10.1109/TDSC.2019.2908164
9. Chinnasamy P., Deepalakshmi P. HCAC-EHR: hybrid cryptographic access control for secure EHR retrieval in healthcare cloud // Journal of Ambient Intelligence and Humanized Computing. 2022. Vol. 13. Iss. 2. PP. 1001–1019. DOI:10.1007/s12652-021-02942-2
10. Contiu S., Pires R., Vaucher S., Pasin M., Felber P., Réveillère L. IBBE-SGX: Cryptographic Group Access Control Using Trusted Execution Environments // Proceedings of the 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN, Luxembourg, Luxembourg, 25–28 June 2018). IEEE, 2018. PP. 207–218. DOI:10.1109/DSN.2018.00032
11. Punithasurya K., Priya S.J. Analysis of Different Access Control Mechanism in Cloud // International Journal of Applied Information Systems. 2012. Vol. 4. Iss. 2. PP. 34–39. DOI:10.5120/IJAIS12-450660
12. Jemihin Z.B., Tan S.F., Chung G.C. Attribute-Based Encryption in Securing Big Data from Post-Quantum Perspective: A Survey // Cryptography. 2022. Vol. 6. Iss. 3. PP. 40. DOI:10.3390/cryptography6030040
13. Гайдамакин Н.А. Теоретические основы компьютерной безопасности: учебно-методический комплекс. Екатеринбург: Уральский государственный университет им. А.М. Горького, 2008. 2012 с.
14. Крашенинников Э.А., Ярмак А.В., Александрова Е.Б. Контроль доступа к данным облачного хранилища на основе изогений // Методы и технические средства обеспечения безопасности информации. 2022. № 31. С. 139–141.
15. Ростовцев А.Г. Эллиптические кривые в криптографии. Теория и вычислительные алгоритмы. СПб.: НПО «Профессионал, 2010. 364 с.
16. Chenu-de la Morinerie M. Supersingular Group Actions and Post-quantum Key-exchange. DSc Thesis. Paris: Polytechnic Institute of Paris, 2021.
17. Castryck W., Decru T. An efficient key recovery attack on SIDH (preliminary version) // Cryptology ePrint Archive. 2022. P. 2022/975. URL: https://eprint.iacr.org/2022/975 (Accessed 12th December 2022)
18. Robert D. Breaking SIDH in polynomial time // Cryptology ePrint Archive. 2022. P. 2022/1038. URL: https://eprint.iacr.org/2022/1038.pdf (Accessed 12th December 2022)
19. Castryck W., Lange T., Martindale C., Panny L., Renes J. CSIDH: an Efficient Post-Quantum Commutative Group Action // Proceedings of the 24th International Conference on the Theory and Application of Cryptology and Information Security (Brisbane, Australia, 2–6 December 2018). Lecture Notes in Computer Science. Vol. 11274. Cham: Springer, 2018. PP. 395–427. DOI:10.1007/978-3-030-03332-3_15
20. Beullens W., Kleinjung T., Vercauteren F. CSI-FiSh: Efficient Isogeny-Based Signatures Through Class Group Computations // Proceedings of the 25th International Conference on the Theory and Application of Cryptology and Information Security (Kobe, Japan, 8–12 December 2019). Lecture Notes in Computer Science. Vol. 11921. Cham: Springer, 2019. PP. 227–247. DOI:10.1007/978-3-030-34578-5_9
21. Atallah M.J., Blanton M., Fazio N., Frikken K.B. Dynamic and Efficient Key Management for Access Hierarchies // ACM Transactions on Information and System Security. 2009. Vol. 12. Iss. 3. PP. 1–43. DOI:10.1145/1455526.1455531
22. Beullens W. CSI-FiSh // Github repository. 2019. URL: https://github.com/KULeuven-COSIC/CSI-FiSh (Accessed 12th December 2022)
23. Agrawal S., Boneh D., Boyen X. Lattice Basis Delegation in Fixed Dimension and Shorter-Ciphertext Hierarchical IBE // Proceedings of the 30th Annual cryptology conference (CRYPTO 2010, Santa Barbara, USA, 15–19 August 2010). Lecture Notes in Computer Science. Vol. 6223. Berlin, Heidelberg: Springer, 2010. PP. 98–115. DOI: 10.1007/978-3-642-14623-7_6

References

1. Krundyshev V., Kalinin M. The Security Risk Analysis Methodology for Smart Network Environments. *Proceedings of the International Russian Automation Conference, RusAutoCon, 06–12 September 2020, Sochi, Russia*. IEEE; 2020. p.437–442. DOI:10.1109/RusAutoCon49822.2020.9208116
2. Ovasapyan T., Moskvin D., Tsvetkov A. Detection of attacks on the Internet of Things based on intelligent analysis of devices functioning indicators. *Proceedings of the 13th International Conference on Security of Information and Networks, SIN, 4–7 November 2020, Merkez Turkey*. New York: Association for Computing Machinery; 2020. p. 3. DOI: 10.1145/3433174.3433611
3. Aleksandrova E.B., Oblogina A.Yu., Shkorkina E.N. Authentication of Intelligent Electronic Devices in IoT Network with the Edge Computing Architecture. *Information Security Problems. Computer Systems*. 2021;2:82–88. (in Russ.)
4. Mesarovic M., Mako D., Takahara Y. Theory of Hierarchical Multilevel Systems. New York, London: Academic Press; 1970. 294 p. (in Italian)
5. Gorkovenko Ye.V. Using of Non-Traditional Cryptographic Transformations in Informational Systems with Mandate Policy of Control Access. *Izvestiya SFedU. Engineering Sciences*. 2008;8(85):135–141. (in Russ.)

6. Di Vimercati S.D.C., Foresti S., Jajodia S., Paraboschi S., Samarati P. Over-encryption: Management of Access Control Evolution on Outsourced Data. *Proceedings of the 33rd International Conference on Very Large Data Bases, VLDB, 23–27 September 2007, Vienna Austria*. VLDB Endowment Inc.; 2007. p.123–134.

7. Epishkina A., Zapechnikov S. On Attribute-Based Encryption for Access Control to Multidimensional Data Structures. *Proceedings of the First International Early Research Career Enhancement School on Biologically Inspired Cognitive Architectures, BICA, for Young Scientist and Cybersecurity, FIERCES 2017, 1–3 August 2017, Moscow, Russia. Advances in Intelligent Systems and Computing, vol. 636*. Cham: Springer; 2017. p.251–256. DOI:10.1007/978-3-319-63940-6_36

8. Qi S., Zheng Y. Crypt-DAC: Cryptographically Enforced Dynamic Access Control in the Cloud. *IEEE Transactions on Dependable and Secure Computing*. 2019;18(2):765–779. DOI:10.1109/TDSC.2019.2908164

9. Chinnasamy P., Deepalakshmi P. HCAC-EHR: hybrid cryptographic access control for secure EHR retrieval in healthcare cloud. *Journal of Ambient Intelligence and Humanized Computing*. 2022;13(2):1001–1019. DOI: 10.1007/s12652-021-02942-2

10. Contiu S., Pires R., Vaucher S., Pasin M., Felber P., Réveillère L. IBBE-SGX: Cryptographic Group Access Control Using Trusted Execution Environments. *Proceedings of the 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN, 25–28 June 2018, Luxembourg, Luxembourg*. IEEE; 2018. p.207–218. DOI:10.1109/DSN.2018.00032

11. Punithasurya K., Priya S.J. Analysis of Different Access Control Mechanism in Cloud. *International Journal of Applied Information Systems*. 2012;4(2):34–39. DOI:10.5120/IJAIS12-450660

12. Jemihin Z. B., Tan S. F., Chung G. C. Attribute-Based Encryption in Securing Big Data from Post-Quantum Perspective: A Survey. *Cryptography*. 2022;6(3):40. DOI:10.3390/cryptography6030040

13. Gaydamakin N. *Theoretical Foundations of Computer Security*. Ekaterinburg: Ural State University A.M. Gorky Publ.; 2008. 2012 p. (in Russ.)

14. Krasheninnikov E.A., Yarmak A.V., Aleksandrova E.B. Isogeny-Based Cloud Storage Data Access Control. *Metody i tekhnicheskiye sredstva obespecheniya bezopasnosti informatsii*. 2022;31:139–141. (in Russ.)

15. Rostovtsev A. *Elliptic Curves in Cryptography. Theory and Computational Algorithms*. St. Petersburg: Professional Publ.; 2010. 364 p. (in Russ.)

16. Chenu-de la Morinerie M. *Supersingular Group Actions and Post-quantum Key-exchange*. DSc Thesis. Paris: Polytechnic Institute of Paris; 2021.

17. Castryck W., Decru T. An efficient key recovery attack on SIDH (preliminary version). *Cryptology ePrint Archive*. P. 2022/975. URL: https://eprint.iacr.org/2022/975 [Accessed 12th December 2022]

18. Robert D. Breaking SIDH in polynomial time. *Cryptology ePrint Archive*. 2022. P. 2022/1038. URL: https://eprint.iacr.org/2022/1038.pdf [Accessed 12th December 2022]

19. Castryck W., Lange T., Martindale C., Panny L., Renes J. CSIDH: an Efficient Post-Quantum Commutative Group Action. *Proceedings of the 24th International Conference on the Theory and Application of Cryptology and Information Security, 2–6 December 2018, Brisbane, Australia. Lecture Notes in Computer Science, vol. 11274*. Cham: Springer; 2018. p.395–427. DOI:10.1007/978-3-030-03332-3_15

20. Beullens W., Kleinjung T., Vercauteren F. CSI-FiSh: Efficient Isogeny-Based Signatures Through Class Group Computations. *Proceedings of the 25th International Conference on the Theory and Application of Cryptology and Information, 8–12 December 2019 Security, Kobe, Japan. Lecture Notes in Computer Science, vol.11921*. Cham: Springer; 2019. p.227–247. DOI:10.1007/978-3-030-34578-5_9

21. Atallah M.J., Blanton M., Fazio N., Frikken K.B. Dynamic and Efficient Key Management for Access Hierarchies. *ACM Transactions on Information and System Security.* 2009;12(3):1–43. DOI:10.1145/1455526.1455531

22. Beullens W. CSI-FiSh. *Github repository*. 2019. URL: https://github.com/KULeuven-COSIC/CSI-FiSh [Accessed 12th December 2022]

23. Agrawal S., Boneh D., Boyen X. Lattice Basis Delegation in Fixed Dimension and Shorter-Ciphertext Hierarchical IBE. *Proceedings of the 30th Annual cryptology conference (CRYPTO 2010, Santa Barbara, USA, 15–19 August 2010). Lecture Notes in Computer Science, vol.6223*. Berlin, Heidelberg: Springer; 2010. p.98–115. DOI: 10.1007/978-3-642-14623-7_6

Статья поступила в редакцию 14.11.2022; одобрена после рецензирования 01.12.2022; принята к публикации 06.12.2022.
The article was submitted 14.11.2022; approved after reviewing 01.12.2022; accepted for publication 06.12.2022.

Информация об авторе:

ЯРМАК Анастасия Викторовна
ассистент Института кибербезопасности и защиты информации Санкт-Петербургского политехнического университета Петра Великого
https://orcid.org/0000-0002-7121-6031