КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 2 (34), 2026
ISSN 2663 – 4023
DOI 10.28925/2663-4023.2026.34.1381
УДК 004.056.5

Балацька Валерія Сергіївна
доктор філософії, доцент,
старший викладач кафедри управління інформаційною безпекою,
Львівський державний університет безпеки життєдіяльності, Львів, Україна
ORCID:0000-0002-6262-6792
v.balatska@ldubgd.edu.ua

МЕТОД ФОРМУВАННЯ ПРОТОКОЛУ МІЖРЕЄСТРОВОЇ ВЗАЄМОДІЇ З МІНІМАЛЬНИМ РОЗКРИТТЯМ ПЕРСОНАЛЬНИХ ДАНИХ

Анотація. У статті розглянуто науково-практичну проблему надлишкового розкриття персональних даних під час взаємодії державних електронних реєстрів. У багатьох випадках для прийняття управлінського або правового рішення достатньо підтвердити виконання визначеної умови, однак система-запитувач отримує значення первинного атрибута чи фрагмент реєстрового запису. Такий підхід збільшує обсяг даних, доступних стороні-перевіряльнику, ускладнює контроль їх подальшого використання та створює додаткові ризики кореляції відомостей між інформаційними системами. Метою дослідження є розроблення методу формування залежного від політики протоколу міжреєстрової взаємодії, який перевіряє допустимість запиту, підтверджує заданий предикат без розкриття вихідного атрибута та створює перевірюваний аудиторський запис без збереження персональних або сталих кореляційних даних у permissioned blockchain. Метод поєднує версійний профіль мінімального розкриття, перевірку мети, правової підстави, аудиторії, строку дії та параметрів предиката, зв'язування ZK-доказу з контекстом конкретної операції й формування ключового аудиторського зобов'язання. У блокчейні фіксуються лише мінімальні відомості, необхідні для подальшої перевірки цілісності операції, тоді як повні доказові дані зберігаються у захищеному позамережевому сховищі. Формальну модель подано для діапазонного предиката та доповнено умовами прийняття рішення, контролю одноразовості запиту і перевірки актуальності політики. Запропонований порядок передбачає перевірку доменних умов до виконання криптографічних операцій, що унеможливлює підміну дозволеної мети позитивним результатом перевірки доказу. Окремо визначено склад мережевих і позамережевих даних, механізм повторного обчислення аудиторського зобов'язання та сценарії виявлення модифікації доказу, повторного використання одноразового значення і звернення за застарілою версією політики або протермінованим статусом відповідного реєстрового твердження. Референсний прототип реалізовано на кривій secp256k1 із використанням зобов'язань Педерсена, бітової декомпозиції, неінтерактивних OR-доказів, Ed25519 і HMAC-SHA-256. Після 15 прогрівальних запусків виконано 150 вимірювань часу та 2000 функціональних перевірок десяти класів. Медіана формування доказу становила 30,587 мс, перевірки – 34,663 мс; 95-й перцентиль – відповідно 32,939 і 37,003 мс. Розмір серіалізованого доказу становив 7185 байтів. Усі валідні запити прийнято, а порушення мети, параметрів, аудиторії, строку дії, версії політики, актуальності статусу, одноразовості та цілісності виявлено. Для дослідженого предиката первинний атрибут стороні-перевіряльнику не передавався. Наукова новизна полягає у формалізації наскрізного зв'язку між дозволеною метою, мінімальним предикатом, контекстним доказом і приватним аудиторським якорем. Практичне значення результатів полягає у можливості використання запропонованого методу як контрольного рівня між сервісом політик, реєстром-джерелом і permissioned blockchain.

Ключові слова: державні електронні реєстри; міжреєстрова взаємодія; персональні дані; мінімізація даних; zero-knowledge proof; permissioned blockchain; протокол; політика доступу; аудиторський якір.

ВСТУП

Цифрова публічна послуга рідко формується в межах одного інформаційного ресурсу. Для прийняття рішення реєстр-запитувач звертається до одного або кількох реєстрів-джерел, зіставляє

одержані відомості, перевіряє встановлені умови, фіксує підставу операції та зберігає її результат. Така взаємодія є необхідною для надання адміністративних і соціальних послуг, підтвердження правового статусу особи, призначення виплат, перевірки повноважень та актуальності реєстрових записів. Водночас кожне нове інтеграційне з'єднання утворює додатковий маршрут передавання персональних даних і розширює коло інформаційних систем, у яких можуть зберігатися їх копії. Закон України «Про публічні електронні реєстри» встановлює правову й організаційну основу створення, ведення та взаємодії реєстрів [2], а Закон України «Про захист персональних даних» пов'язує склад і зміст персональних даних із визначеною метою їх обробки [1]. Аналогічний вектор задає принцип data minimisation, закріплений у ст. 5 Регламенту (ЄС) 2016/679: дані мають бути адекватними, релевантними та обмеженими тим, що необхідно для конкретної мети, а володілець повинен мати змогу продемонструвати дотримання цього принципу [3]. Отже, міжреєстровий протокол має одночасно забезпечувати достатність відомостей для прийняття рішення, ненадмірність їх розкриття, контроль правової підстави запиту та доказовість правомірності виконаної операції.

На прикладному рівні ця вимога реалізується неповно. Якщо сервісу необхідно встановити лише факт досягнення визначеного віку, чинності запису, належності до відповідної категорії або наявності права на послугу, інтерфейс часто повертає дату народження, ідентифікатор особи чи фрагмент повного реєстрового запису. Система-одержувач самостійно обчислює потрібний результат, хоча для виконання її функції достатньо відповіді про істинність визначеної умови. Навіть вибіркове передавання одного атрибута залишає його відомим одержувачу та збільшує обсяг персональних даних у журналах, кешах, резервних копіях, інтеграційних шлюзах і системах моніторингу. Надлишкове розкриття має накопичувальний характер. Окреме передавання одного атрибута може здаватися прийнятним, проте повторення таких операцій між різними установами створює можливість зіставлення відомостей, побудови профілю особи та встановлення зв'язків між її зверненнями. Додаткові ризики виникають у разі використання сталих ідентифікаторів, відкритих хешів низькоентропійних значень або однакових доказових записів у різних інформаційних системах. Тому захист транспортного каналу не усуває проблеми надмірного змісту повідомлення: зашифровані під час передавання дані після розшифрування все одно стають доступними стороні-одержувачу.

Перспективним напрямом розв'язання цієї проблеми є перехід від передавання значень атрибутів до підтвердження предикатів над ними. Застосування доказів із нульовим розголошенням дає змогу підтвердити виконання заданої умови без розкриття початкового значення. Однак самого криптографічного доказу недостатньо: протокол повинен перевіряти мету звернення, правову підставу, повноваження перевіряльника, параметри предиката, строк дії запиту, актуальність політики та одноразовість операції. Водночас аудит не має створювати нове незмінне сховище персональних або кореляційних даних. Це зумовлює необхідність поєднання контекстно зв'язаного ZK-доказу з мінімальним приватним аудиторським записом, який може бути зафіксований у permissioned blockchain і перевірений уповноваженою стороною без розкриття первинного атрибута.

Постановка проблеми. ISO/IEC 27565:2026 розглядає zero-knowledge proof (ZKP) як засіб зменшення ризиків передавання персональних даних [4]. Проте коректний доказ умови $x \geq 18$ не підтверджує, що запитувач мав право перевіряти саме цю умову, діяв із належною метою та правовою підставою, використовував чинний статус і одноразовий запит. Такі параметри мають входити до протокольного контексту й перевірятися разом із доказом. Під час аудиту необхідно підтвердити мету, версію політики та результат операції. Permissioned blockchain придатний для узгодження незмінного міжорганізаційного журналу, але запис сталих ідентифікаторів, повних доказів або відкритих хешів низькоентропійних полів створює ризик кореляції. Тому склад on-chain запису також повинен підпорядковуватися принципу мінімізації.

Отже, науково-практична проблема полягає у відсутності виконуваного зв'язку між нормативно визначеною метою операції, мінімально необхідним предикатом, його криптографічним доказом та аудиторським записом, який можна перевірити без перетворення блокчейну на сховище персональних або кореляційних даних.

Об'єктом дослідження є процес захищеної інформаційної взаємодії державних електронних реєстрів. Предметом дослідження є моделі, методи й протокольні механізми мінімального розкриття персональних даних, контекстної перевірки ZKP та приватного блокчейн-аудиту міжреєстрових операцій.

Аналіз останніх досліджень і публікацій. Нормативну основу дослідження утворюють вимоги до мети, мінімізації, простежуваності й контрольованості обробки [1-5]. NIST Privacy Framework пов'язує ці вимоги з керуванням ризиками приватності [20]. W3C Verifiable Credentials Data Model 2.0 розмежовує вибіркове та незв'язуване розкриття [6], а OpenID for Verifiable Presentations 1.0 визначає механізм

формування запитів щодо цифрових посвідчень [7]. Водночас ці документи не встановлюють доменної відповідності між правовою метою державного органу та єдиним дозволеним предикатом.

У W3C Data Integrity BBS Cryptosuites описано вибіркове розкриття й незв'язувані похідні докази, а також ризики витоку через метадані та структуру посвідчення [8]. Порівняння криптографічних механізмів у [9] показало відмінності в підтримці предикатів, незв'язуваності, розмірі доказу та часі обробки. У роботі [10] обґрунтовано вплив ZKP на мінімізацію даних у цифрових гаманцях, але звернуто увагу на можливість кореляції за допоміжними артефактами. Питання машинно виконуваного контролю мети міжреєстрового запиту залишається відкритим.

Криптографічну основу рішення утворюють зобов'язання Педерсена [11], OR-докази часткового знання [12] і перетворення Fiat-Shamir [13]. Для однозначного кодування контексту використано JSON Canonicalization Scheme [14], для засвідчення зобов'язання реєстром-джерелом – Ed25519 [15], для приховування низькоентропійних полів аудиту – HMAC [16]. TLS 1.3 захищає транспортний канал [17], але не замінює перевірки мети, версії політики та предиката.

У permissioned blockchain правила участі й підтвердження транзакцій визначаються консорціумом; відповідні властивості реалізовано, зокрема, у Hyperledger Fabric [18]. Придатність такого підходу для захищеної аудиторської інфраструктури показано в [19]. Однак незмінність не усуває ризику надмірного змісту запису, тому необхідно формально розмежувати on-chain і off-chain дані.

У попередніх дослідженнях автора розглянуто застосування блокчейну для захисту персональних даних, державних реєстрів і міжорганізаційного обміну [21-25], а також інтеграцію політик, стандартів і протоколів у процес побудови систем захисту [26]. Невирішеним залишається попередній контроль відповідності запиту меті та формування з політики предиката, який не потребує передавання первинного атрибута.

Дослідницький розрив охоплює нормативний, криптографічний та аудиторський рівні. Вимога мінімізації не перетворена на правило вибору єдиного дозволеного предиката; доказ без контекстного зв'язування можна спробувати використати для іншої мети або аудиторії; незмінний запис у блокчейні може містити кореляційні поля. Отже, ці вимоги мають виконуватися в межах одного протоколу.

Результати аналізу нормативних документів і праць систематизовано в табл. 1. Для кожного механізму визначено його функцію, невирішене питання та відповідну вимогу до методу.

Таблиця 1

Зіставлення наявних механізмів із невирішеними протокольними питаннями

Джерело або механізм	Визначена функція	Що не визначено для міжреєстрової операції	Вимога до методу
Закони України, GDPR	мета, правомірність, мінімізація, підзвітність	машинно виконувана відповідність «мета – предикат»	версійний профіль дозволених запитів
ISO/IEC 27565:2026	орієнтири застосування ZKP для захисту приватності	доменна авторизація перевіряльника й параметрів предиката	оператор запиту та контекстні умови
ISO/IEC 27551:2021	атрибутна незв'язувана автентифікація	аудит правової підстави конкретної операції	прив'язка доказу до мети й правової підстави
W3C VC 2.0, OpenID4VP	модель цифрового посвідчення й презентації	обов'язкова мінімальність доменного запиту	перелік предикатів і точних параметрів
BBS / вибіркове розкриття	приховування частини тверджень, незв'язувані докази	числові предикати й контроль on-chain метаданих	змінна ZKP-схема й заборона сталих ID
Permissioned blockchain	узгоджений незмінний аудиторський журнал	захист низькоентропійних і кореляційних полів	ключове аудиторське зобов'язання, off-chain дані

Як видно з табл. 1, наявні підходи створюють необхідні, але недостатні умови. Нормативні джерела визначають властивості, стандарти презентацій – формат обміну, ZKP – приховування значення, а блокчейн – незмінність запису. Для їх узгодження потрібне правило, за яким дозволеність предиката

перевіряється до формування доказу, доказ зв'язується з одноразовим контекстом, а в блокчейн надходить лише прихований аудиторський якір.

Мета статті. Метою статті є розроблення та експериментальна перевірка методу формування залежного від політики протоколу міжреєстрової взаємодії, який перетворює вимоги щодо мети, правової підстави й мінімізації даних на виконувані умови, забезпечує перевірку предиката без розкриття вихідного атрибута та формує приватний аудиторський якір у permissioned blockchain.

Гіпотеза дослідження полягає в тому, що поєднання версійного профілю, контекстно зв'язаного ZK-доказу та ключового аудиторського зобов'язання дасть змогу не передавати первинний атрибут і водночас виявляти порушення мети, параметрів, аудиторії, актуальності або цілісності.

Основними завданнями статті виступають:

Визначити модель загроз і перетворити вимоги нормативних документів та стандартів на протокольні інваріанти.

Розробити версійний профіль мінімального розкриття та оператор формування дозволеного предикатного запиту.

Формалізувати контекстно зв'язаний ZK-доказ, умови прийняття й приватний аудиторський якір.

Розробити алгоритм взаємодії та розмежувати on-chain і off-chain дані.

Реалізувати прототип і перевірити його за показниками розкриття атрибутів, часу формування й перевірки доказу, розміру доказу та правильності обробки заданих сценаріїв.

РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ

Модель загроз і властивості безпеки. У протоколі взаємодіють реєстр-запитувач R_q, сервіс політик P_s, реєстр-джерело R_s і шлюз-перевіряльник V_g. Permissioned blockchain L_p підтримується відомими членами консорціуму. Транспортні з'єднання захищені TLS 1.3 [17], однак прикладний вузол може бути неправильно налаштованим, надмірно запитувати дані або діяти в межах скомпрометованих повноважень. Реєстр-джерело вважається довіреним щодо правильності засвідченого атрибута; достовірність первинного запису перебуває поза межами ZKP.

Модель загроз охоплює недозволену мету, розширення параметрів предиката, підміну verifier_id, повторне відтворення презентації, використання застарілої версії політики або статусу, модифікацію доказу чи аудиторського запису та кореляцію on-chain метаданих. Поза моделлю залишено компрометацію ключа реєстру-джерела, змову всіх уповноважених аудиторів і побічні канали конкретної ZKP-реалізації. Відповідність загроз, протокольних інваріантів і контрольних умов наведено в табл. 2. Кожен негативний сценарій пов'язано з конкретною умовою перевірки.

Таблиця 2

Загрози, інваріанти та умови їх виконання

Загроза	Протокольний інваріант	Перевірочна умова	Наслідок порушення
недозволена мета або підстава	доказ формується лише для дозволеної пари purpose–legal_basis	g_1, g_2	відмова до звернення по атрибут
надмірний предикат або параметри	запит точно відповідає переліку дозволених предикатів	g_3	відмова оператора формування запиту
підміна аудиторії	verifier_id входить до контексту підпису й доказу	g_4	недійсний підпис або виклик
повторне відтворення	nonce одноразовий, строк дії обмежений	g_5, g_6	повторна презентація відхиляється
застаріла політика або статус	version і status_epoch є актуальними	g_7, g_8	відмова до перевірки ZKP
підміна зобов'язання або доказу	підпис реєстру та ZKP перевіряються над одним ctx	g_9, g_{10}	криптографічна відмова
кореляція через ledger	стабільні ID, C і Π не записуються on-chain	g_{11}	транзакція з надмірними полями не приймається
зміна off-chain доказових даних	ключове зобов'язання повторно обчислюється аудитором	g_{12}	невідповідність аудиторському якорю

За табл. 2 умови g_1-g_8 перевіряються до криптографічних операцій, g_9-g_{10} підтверджують цілісність дозволеного твердження, g_{11} контролює схему транзакції, а g_{12} використовується під час аудиту. Позитивний результат ZKP не скасовує відмову за доменною умовою.

Нормативно-протокольний профіль мінімального розкриття

Сирий запит реєстру подано кортежем

$$q = \langle v, p, \ell, \varphi, \theta, n, t_0, t_e, s_e \rangle, \tag{1}$$

де v – ідентифікатор перевіряльника; p – код мети; ℓ – правова підстава; φ – предикат; θ – його публічні параметри; n – одноразове випадкове значення; t_0, t_e – час формування та завершення строку дії; s_e – епоха статусу.

Профіль мінімального розкриття π_j є версійним нормативно-технічним об'єктом:

$$\pi_j = \langle \mathrm{id}_j, \nu_j, V_j, P_j, L_j, \Phi_j, \Theta_j, A_j, I_j, \tau_j, S_j \rangle, \tag{2}$$

де id_j – ідентифікатор профілю; ν_j – його версія; V_j, P_j і L_j – відповідно дозволені множини перевіряльників, цілей і правових підстав; Φ_j – множина дозволених предикатів; Θ_j – множина допустимих параметрів предикатів; $A_j \subseteq V_j \times P_j \times L_j \times \Phi_j \times \Theta_j$ – множина дозволених комбінацій перевіряльника, мети, правової підстави, предиката та його публічних параметрів; I_j – множина довірених джерел; τ_j – максимально допустимий строк життя запиту; S_j – правило перевірки актуальності статусу. Профіль підписується уповноваженим органом і зберігається разом з історією версій, що забезпечує можливість відтворення правил, чинних на момент виконання конкретної операції.

Оператор $\mathcal{K}_{\pi_j}$ на основі профілю π_j формує канонічний дозволений запит q^* або повертає відмову $\perp (rc)$ із кодом першої порушеної умови:

$$\mathcal{K}_{\pi_j}(q) = \begin{cases} q^* = \mathrm{Canon}(q, \nu_j), & G_{\pi_j}(q) = 1, \\ \perp (rc), & G_{\pi_j}(q) = 0. \end{cases} \tag{3}$$

де функція $G_{\pi_j}(q)$ визначає виконання сукупності обов'язкових умов профілю:

$$G_{\pi_j} G_{\pi_j}(q) = \left[(v, p, \ell, \varphi, \theta) \in A_j\right] \wedge \left[0 \le t_e - t_0 \le \tau_j\right] \wedge \left[t_{\mathrm{now}} \le t_e\right] \wedge \left[\mathrm{Fresh}(s_e, S_j, t_0) = 1\right] \tag{4}$$

Квадратні дужки у формулі (4) позначають індикатор істинності відповідної умови. Належність кортежу $(v, p, \ell, \varphi, \theta)$ до множини A_j забезпечує перевірку не лише окремої допустимості його складових, а й дозволеності їх конкретної комбінації. Таким чином, визначена мета та правова підстава однозначно пов'язуються з конкретним предикатом і його публічними параметрами. Це унеможливлює використання дозволеної мети для перевірки іншої умови або заміну, наприклад, предиката $x \ge 18$ серією довільно сформованих діапазонних запитів, що могли б поступово звужувати приховане значення атрибута.

Умова $0 \le t_e - t_0 \le \tau_j$ обмежує максимально допустимий строк життя запиту, тоді як перевірка $t_{\mathrm{now}} \le t_e$ виключає обробку запиту після завершення строку його дії. Функція $\mathrm{Fresh}(s_e, S_j, t_0)$ визначає відповідність епохи статусу правилам актуальності, встановленим профілем для моменту формування запиту.

Склад профілю та безпекове призначення його полів наведено в табл. 3. Семантичні, криптографічні та часові параметри розмежовано таким чином, щоб зміна використовуваної ZKP-схеми не призводила до зміни правової семантики відповідної послуги.

Таблиця 3

Структура версійного профілю мінімального розкриття

Поле	Зміст	Обов'язкова перевірка	Захищувана властивість
profile_id, version	ідентифікатор і незмінна версія	версія чинна на t_0	відтворюваність рішення
verifier_set	дозволені організації/сервіси	$v \in V_j$	обмеження аудиторії

Поле	Зміст	Обов'язкова перевірка	Захищувана властивість
purpose_set	коди функцій або послуг	$p \in P_j$	обмеження мети
legal_basis_set	нормативні підстави	$\ell \in L_j$	правомірність звернення
predicate_catalog	дозволені φ і точні θ	точний збіг	семантична мінімальність
issuer_trust	ключі/сертифікати джерел	ключ чинний і не відкликаний	автентичність атрибута
status_policy	джерело та епоха статусу	$s_e = S_j$	актуальність
freshness	TTL і правила nonce	строк і одноразовість	захист від replay
proof_suite	дозволена ZKP-схема і параметри	suite входить до allowlist	криптографічна гнучкість
audit_schema	поля on-chain/off-chain, retention	схема смартконтракту	приватність і аудит

Як видно з табл. 3, профіль є вхідними даними алгоритму. Додавання нового сервісу потребує нової версії, а ретроспективний аудит використовує правила, чинні на момент операції. Після компіляції формується контекст операції. Щоб виключити різні представлення однакових полів, об'єкт серіалізується за детермінованим правилом JCS [14]:

$$ctx = H\left(\mathrm{JCS}\left(id_j \parallel v_j \parallel v \parallel p \parallel \ell \parallel \varphi \parallel \theta \parallel n \parallel t_0 \parallel t_e \parallel s_e\right)\right). \quad (5)$$

Метод виконує п'ять послідовних дій: нормалізує запит, зіставляє його з чинним профілем, обчислює ctx, формує одноразове криптографічне зобов'язання та перевіряє доменні й криптографічні умови. Результат для прикладного сервісу й аудиторський якір утворюються окремо. Відмова на будь-якій умові завершує обробку без повернення прихованого атрибута. Таким чином, профіль перетворює вимогу мінімізації на виконувану конфігурацію з наперед визначеною відповідністю між метою, підставою, перевіряльником і параметрами предиката.

Формальна модель протоколу та контекстне зв'язування доказу. Нехай реєстр-джерело зберігає атрибут x, а G і H_G – незалежні генератори групи простого порядку q_g. Для кожного дозволеного запиту джерело створює нове випадкове $p \in \mathbb{Z}_{q_g}$ та одноразове зобов'язання Педерсена [11]:

$$C = xG + pH_G. \quad (6)$$

Нове значення p для кожного ctx унеможливлює пряме зіставлення двох зобов'язань, сформованих для одного й того самого атрибута. Реєстр-джерело засвідчує зобов'язання та контекст операції цифровим підписом:

$$\sigma_I = \mathrm{Sign}_{sk_I}(ctx \parallel C \parallel schema\_id \parallel s_e) \quad (7)$$

Підпис (7) не розкриває x, але підтверджує походження зобов'язання, одноразовий запит та епоху статусу. У дослідженій моделі реєстр-джерело бере участь у кожній операції. Для роботи через цифровий гаманець цей етап може бути замінений сумісною зі ZKP схемою посвідчень без зміни профілю та контрольних умов.

Для діапазонного предиката $\varphi(x)\colon L \le x \le U$ формується неінтерактивний доказ

$$\Pi_\varphi \leftarrow \mathrm{NIZK}\{(x, \mathrm{p})\colon C = xG + rH_G \wedge L \le x \le U\}(ctx). \quad (8)$$

Включення ctx до виклику Fiat-Shamir [13] криптографічно зв'язує доказ із конкретною операцією та унеможливлює його коректне повторне використання для іншої мети, аудиторії, версії політики, часового контексту або nonce. Публічними залишаються L, U, зобов'язання C та булевий результат; значення x та маскувальний параметр p перевіряльнику не передаються.

Остаточне рішення формується як кон'юнкція доменних і криптографічних умов, а не лише як результат перевірки ZK-доказу:

$$d = \left(\bigwedge_{k=1}^{8} g_k\right) \wedge \mathrm{VerifySig}_{pk_I}(\sigma_I, ctx, C, s_e) \wedge \mathrm{VerifyZK}(\Pi_\varphi, ctx, C, \theta). \quad (9)$$

Якщо $d = 0$, повертається узагальнений reason_code, який без окремого дозволу політики не розкриває, чи пов'язана відмова зі значенням атрибута. Детальна причина зберігається лише в захищених off-chain доказових даних для уповноваженого аудитора.

Алгоритм міжреєстрової взаємодії та приватний аудит

Для аудиту формується ключове зобов'язання:

$$a_i = \mathrm{HMAC}_{K_A}\left(\mathrm{JCS}\left(ctx_i \parallel H(C_i) \parallel H(\Pi_i) \parallel d_i \parallel r_i \parallel \nu_j \parallel b_i\right)\right), \quad (10)$$

де r_i – код результату, b_i – часовий інтервал, а K_A – ключ аудиту, доступний визначеному політикою кворуму. On-chain запис містить a_i, profile_id, часовий інтервал і технічний ідентифікатор транзакції, але не містить сталого ідентифікатора особи, C_i, Π_i або повного контексту. Канонічні доказові дані зберігаються шифровано off-chain відповідно до політики зберігання. HMAC [16] не дає змоги перевіряти припущення про низькоентропійні поля без володіння K_A.

Сервіс політик спочатку підтверджує семантичну допустимість R^*, після чого шлюз перевіряє підпис профілю, підпис реєстру-джерела та ZKP. Permissioned blockchain залучається після прийняття рішення й отримує лише підсумковий аудиторський якір. Такий порядок не допускає підміни доменної авторизації позитивним результатом криптографічної перевірки.

Алгоритм виконання методу має такий вигляд:

1. прийняти q, перевірити синтаксичну схему, типи й допустимий розмір полів;
2. отримати підписану чинну версію π_j та виконати $\mathcal{K}(q, \pi_j)$; за результату $\perp$ (r) завершити операцію;
3. канонізувати R^*, обчислити ctx, зарезервувати nonce в атомарному сховищі одноразовості;
4. передати R^* реєстру-джерелу; сформувати одноразове C, підпис σ_I та Π_φ;
5. перевірити актуальність ключа джерела, підпис профілю, g_1-g_8, σ_I і Π_φ; сформувати d за (9);
6. сформувати off-chain доказові дані, обчислити a_i за (10) та подати мінімальну транзакцію смартконтракту;
7. перевірити квитанцію блокчейну, позначити nonce використаним і повернути d, reason_code та tx_id;
8. під час аудиту відновити дозволений набір даних із захищеного сховища, повторно обчислити a_i й порівняти його з on-chain значенням.

Атомарність кроків 3 і 7 потребує окремої обробки. Якщо транзакція не потрапила до блокчейну, але nonce вже використано, система створює технічну подію відновлення, не дозволяючи повторну презентацію доказу. Режим відмови або контрольованої черги за тимчасової недоступності блокчейну визначається профілем послуги. Формули (5)-(10) пов'язують політику, одноразовий запит, джерельний атрибут, результат і аудит без зберігання атрибута чи повного доказу в розподіленому реєстрі.

Прототип, сценарії експерименту та виявлення маніпуляцій. Експеримент перевіряв три положення: коректний предикат обчислюється без передавання x; зміна критичного поля призводить до відмови; витрати криптографічної частини можна виміряти окремо від часу блокчейн-мережі.

Прототип реалізовано мовою Python 3.12.13 у середовищі Linux x86_64. Операції secp256k1 виконувалися через OpenSSL 3.5.7, засвідчення зобов'язання – Ed25519 [15], канонізація – відповідно до JCS [14], аудит – HMAC-SHA-256 [16]. Для повторюваності тестів використано детермінований генератор скалярів із фіксованим початковим значенням; у виробничому середовищі такий режим не застосовується.

Предикат мав вигляд $18 \le x \le 120$. Для прототипу введено $y = x - 18$, $z = 120 - x$ і $k = \lceil \log_2(120 - 18 + 1) \rceil = 7$. Кожне значення розкладалося на біти, для яких формувалися зобов'язання $B_{y,i}$, $B_{z,i}$ та OR-доказ належності біта множині $\{0,1\}$ [12]:

$$\begin{aligned} B_{y,i} &= b_{y,i}G + \rho_{y,i}H_G, \quad C - 18G = \sum_{i=0}^{k-1} 2^i B_{y,i}, \\ B_{z,i} &= b_{z,i}G + \rho_{z,i}H_G, \quad 120G - C = \sum_{i=0}^{k-1} 2^i B_{z,i}. \end{aligned} \quad (11)$$

Спільне виконання рівностей (11) доводить нижню й верхню межі без розкриття x. У промисловій системі поле proof_suite може визначати компактніший доказ діапазону або сумісну схему цифрових посвідчень.

Після 15 прогрівальних запусків виконано 150 вимірювань часу формування та перевірки доказу. Мережевий обмін, звернення до реального державного реєстру, консенсус і підтвердження транзакції не враховувалися. Розмір визначався для канонічного JSON-представлення без транспортних заголовків. Функціональна перевірка охоплювала десять класів по 200 запусків: валідний запит, недозволену мету, надмірні параметри, іншу аудиторію, прострочення, застарілі політику й статус, повтор nonce, підміну доказу та підміну off-chain даних. Результат кожного запуску був булевим; останній клас перевіряв виявлення невідповідності під час аудиту.

Для порівняння семантичного розкриття введено показник

$$MDR = 1 - \frac{\sum_{a \in A} w_a \, \delta_a^{\text{prop}}}{\sum_{a \in A} w_a \, \delta_a^{\text{base}}}, \tag{12}$$

де A – множина первинних атрибутів базової відповіді; w_a – вага чутливості; $\delta_a = 1$, якщо атрибут розкрито перевіряльнику. У контрольному прикладі ваги однакові, тому для повного запису із семи атрибутів знаменник дорівнює 7. Публічні параметри предиката, булевий результат і технічні метадані аналізуються окремо, оскільки зменшення семантичного розкриття може супроводжуватися збільшенням обсягу переданих даних.

Оцінювання ефективності та аналіз результатів. Числові результати вимірювань наведено в табл. 4. Основними показниками обрано медіану та 95-й перцентиль; середнє й стандартне відхилення подано як додаткові характеристики.

Таблиця 4

Час виконання та розмір доказу в референсному прототипі

Операція/показник	Медіана	95-й перцентиль	Середнє	Стандартне відхилення
підпис зобов'язання Ed25519, мс	0,064	0,091	0,072	0,032
формування $П_\varphi$, мс	30,587	32,939	30,751	1,127
перевірка підпису Ed25519, мс	0,137	0,184	0,143	0,023
перевірка $П_\varphi$, мс	34,663	37,003	34,943	1,136
розмір серіалізованого $П_\varphi$, байт	7185	7185	7185	0

За табл. 4, основний час припадає на формування та перевірку доказу діапазону; підпис і його перевірка додають менше 0,2 мс за 95-м перцентилем. Сумарна медіана локального криптографічного шляху становить близько 65,45 мс без урахування мережі й блокчейну. Розмір 7185 байтів зумовлений семибітовою декомпозицією обох меж і JSON-кодуванням.

Розподіл часу формування та перевірки доказу за всіма 150 спостереженнями подано на рис. 1.

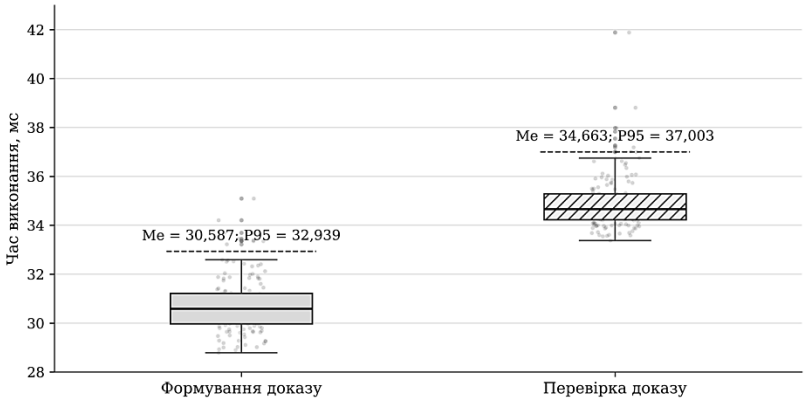

Рис.1. Розподіл часу формування та перевірки ZK-доказу

Як видно з рис. 1, міжквартильні діапазони обох операцій є вузькими; поодинокі верхні викиди пов'язані з виконанням у загальному операційному середовищі. Перевірка доказу стабільно потребує більше часу, ніж його формування.

Результати функціональних перевірок наведено в табл. 5. Окремо подано прийняті, відхилені та помилкові рішення.

Таблиця 5

Результати функціональних перевірок, 200 запусків на клас

Сценарій	Прийнято	Відхилено/виявлено	Помилкове прийняття	Помилкове відхилення
валідний запит	200	0	0	0
недозволена мета	0	200	0	–
надмірні параметри	0	200	0	–
інша аудиторія	0	200	0	–
прострочений запит	0	200	0	–
застаріла політика	0	200	0	–
застарілий статус	0	200	0	–
повтор nonce	0	200	0	–
підміна доказу	0	200	0	–
підміна off-chain даних	0	200	0	–

За табл. 5 усі 200 валідних запитів прийнято, а 1800 змінених випадків відхилено або виявлено під час аудиту. Мета, параметри, версія політики та статус перевіряються до ZKP; підміна off-chain даних встановлюється повторним обчисленням a_i. Отриманий результат стосується визначених тестових класів і не характеризує всі можливі атаки на виробничу систему.

Порівняння трьох способів перевірки вікового предиката наведено в табл. 6. Воно відокремлює семантичну мінімізацію від транспортних витрат.

Таблиця 6

Порівняння розкриття та протокольних властивостей для перевірки вікового предиката

Варіант відповіді	Розкриті первинні атрибути	Значення віку відоме перевіряльнику	Контекстна прив'язка	Відтворюваний приватний аудит	Додатковий обсяг доказу
повний реєстровий запис	7	так	залежить від API	журнал містить деталі	0 байт
вибіркове розкриття віку	1	так	можлива	журнал містить атрибут/ID	0 байт
запропонований метод	0	ні	обов'язкова	ключове зобов'язання	7185 байтів

За табл. 6 для дослідженого предиката $MDR = 1$: порівняно з повним семіатрибутним записом розкриття первинних атрибутів зменшено на 100%, а порівняно з вибірковим передаванням віку не розкривається саме значення. Додаткові витрати становлять 7185 байтів і близько 65 мс локальних ZKP-операцій. Прийнятність такого співвідношення залежить від чутливості атрибута, частоти звернень і доступності компактнішої криптографічної схеми.

ВИСНОВКИ

Запропонований у роботі протокол змінює порядок міжреєстрової перевірки. Реєстр-запитувач не одержує дату народження, статус, ідентифікатор чи інший вихідний атрибут, а отримує підтвердження того, що визначена умовою вимога виконується. До формування ZK-доказу оператор KK зіставляє запит із чинною версією профілю: перевіряє дозволеного одержувача, мету звернення, правову підставу, вид

предиката, його параметри, строк дії та актуальність статусу. Якщо хоча б одна із цих умов не виконується, обробка припиняється до звернення по первинний атрибут. Такий порядок не дозволяє використати правильний криптографічний доказ для іншої мети, іншого перевіряльника або застарілої політики. Для кожної дозволеної операції реєстр-джерело створює нове зобов'язання атрибута та засвідчує його разом із контекстом запиту. Одноразове значення і часові межі унеможливлюють повторне подання того самого доказу. У permissioned blockchain записується не персональна інформація і не повний ZK-доказ, а ключоване аудиторське зобов'язання, версія профілю, часовий інтервал і технічний результат операції. Дані, потрібні уповноваженому аудитору для повторної перевірки, зберігаються зашифрованими поза блокчейном. Завдяки цьому незмінність аудиторського запису не супроводжується створенням нового реєстру персональних даних.

Перевірка прототипу охопила 150 вимірювань часу та 2000 запитів десяти класів. Медіана формування доказу становила 30,587 мс, його перевірки – 34,663 мс; розмір JSON-представлення дорівнював 7185 байтам. Усі 200 коректних запитів було прийнято. У 1800 змінених випадках система відхилила недозволену мету, надмірні параметри, іншу аудиторію, прострочений запит, застарілу політику або статус, повторне використання nonce та підміну доказових даних. Первинне значення атрибута в жодному тесті перевіряльнику не передавалося.

Наведені часові показники стосуються локального прототипу без мережевої затримки та блокчейн-консенсусу. Наступним етапом має бути випробування протоколу в мережі Hyperledger Fabric із кількома організаціями, вимірювання повного часу операції, заміна експериментального доказу діапазону компактнішою схемою та перевірка інтеграції з цифровими посвідченнями.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Verkhovna Rada of Ukraine. (2010). *Pro zakhyst personalnykh danykh* [On personal data protection] (Law of Ukraine No. 2297-VI, June 1, 2010). https://zakon.rada.gov.ua/laws/show/2297-17#Text
2. Verkhovna Rada of Ukraine. (2021). *Pro publichni elektronni reiestry* [On public electronic registers] (Law of Ukraine No. 1907-IX, November 18, 2021). https://zakon.rada.gov.ua/laws/show/1907-20#Text
3. European Parliament & Council of the European Union. (2016). Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 (General Data Protection Regulation). *Official Journal of the European Union, L 119*, 1–88. https://eur-lex.europa.eu/eli/reg/2016/679/oj
4. International Organization for Standardization & International Electrotechnical Commission. (2026). *Information security, cybersecurity and privacy protection – Guidelines on privacy preservation based on zero-knowledge proofs* (ISO/IEC 27565:2026). ISO. https://www.iso.org/standard/80398.html
5. International Organization for Standardization & International Electrotechnical Commission. (2021). *Information security, cybersecurity and privacy protection – Requirements for attribute-based unlinkable entity authentication* (ISO/IEC 27551:2021). ISO. https://www.iso.org/standard/72018.html
6. World Wide Web Consortium. (2025). *Verifiable credentials data model v2.0* (W3C Recommendation, May 15, 2025). https://www.w3.org/TR/vc-data-model-2.0/
7. OpenID Foundation. (2025). *OpenID for verifiable presentations 1.0* (OpenID Final Specification, July 9, 2025). https://openid.net/specs/openid-4-verifiable-presentations-1_0.html
8. Bernstein, G., & Sporny, M. (2026). *Data integrity BBS cryptosuites v1.0* (W3C Candidate Recommendation Draft, April 7, 2026). World Wide Web Consortium. https://www.w3.org/TR/vc-di-bbs/
9. Flamini, A., Sciarretta, G., Scuro, M., Sharif, A., Tomasi, A., & Ranise, S. (2024). On cryptographic mechanisms for the selective disclosure of verifiable credentials. *Journal of Information Security and Applications, 83*, Article 103789. https://doi.org/10.1016/j.jisa.2024.103789
10. Podda, E., Hölzmer, P., Amard, A., Sedlmeir, J., & Fridgen, G. (2025). The impact of zero-knowledge proofs on data minimisation compliance of digital identity wallets. *Internet Policy Review, 14*(3), 1–29. https://doi.org/10.14763/2025.3.2019
11. Pedersen, T. P. (1992). Non-interactive and information-theoretic secure verifiable secret sharing. In *Advances in Cryptology – CRYPTO '91* (Lecture Notes in Computer Science, Vol. 576, pp. 129–140). Springer. https://doi.org/10.1007/3-540-46766-1_9
12. Cramer, R., Damgård, I., & Schoenmakers, B. (1994). Proofs of partial knowledge and simplified design of witness hiding protocols. In *Advances in Cryptology – CRYPTO '94* (Lecture Notes in Computer Science, Vol. 839, pp. 174–187). Springer. https://doi.org/10.1007/3-540-48658-5_19
13. Fiat, A., & Shamir, A. (1987). How to prove yourself: Practical solutions to identification and signature problems. In *Advances in Cryptology – CRYPTO '86* (Lecture Notes in Computer Science, Vol. 263, pp. 186–194). Springer. https://doi.org/10.1007/3-540-47721-7_12

14. Rundgren, A., Jordan, B., & Erdtman, S. (2020). *JSON Canonicalization Scheme (JCS)* (RFC 8785). Internet Engineering Task Force. https://www.rfc-editor.org/rfc/rfc8785
15. Josefsson, S., & Liusvaara, I. (2017). *Edwards-Curve Digital Signature Algorithm (EdDSA)* (RFC 8032). Internet Engineering Task Force. https://www.rfc-editor.org/rfc/rfc8032
16. Krawczyk, H., Bellare, M., & Canetti, R. (1997). *HMAC: Keyed-hashing for message authentication* (RFC 2104). Internet Engineering Task Force. https://www.rfc-editor.org/rfc/rfc2104
17. Rescorla, E. (2026). *The Transport Layer Security (TLS) Protocol Version 1.3* (RFC 9846). Internet Engineering Task Force. https://doi.org/10.17487/RFC9846
18. Androulaki, E., Barger, A., Bortnikov, V., et al. (2018). Hyperledger Fabric: A distributed operating system for permissioned blockchains. In *Proceedings of the Thirteenth EuroSys Conference* (Article 30, pp. 1–15). ACM. https://doi.org/10.1145/3190508.3190538
19. Putz, B., Menges, F., & Pernul, G. (2019). A secure and auditable logging infrastructure based on a permissioned blockchain. *Computers & Security, 87*, Article 101602. https://doi.org/10.1016/j.cose.2019.101602
20. National Institute of Standards and Technology. (2020). *NIST Privacy Framework: A tool for improving privacy through enterprise risk management* (Version 1.0). https://www.nist.gov/privacy-framework/privacy-framework
21. Balatska, V. S., & Opirskyy, I. R. (2023). Ensuring personal data confidentiality and cybersecurity support using blockchain. *Cybersecurity: Education, Science, Technique, 4*(20), 6–19. https://doi.org/10.28925/2663-4023.2023.20.619
22. Balatska, V., & Poberezhnyk, V. (2024). Concept of using blockchain technologies to enhance personal data security of the Diia platform: Compliance with GDPR and Ukrainian legislation. *Cybersecurity: Education, Science, Technique, 2*(26), 268–290. https://doi.org/10.28925/2663-4023.2024.26.681
23. Balatska, V., Slobodian, N., & Opirskyy, I. (2024). Blockchain for enhancing transparency and trust in government registries. *CEUR Workshop Proceedings, 3826*, 50–59. https://ceur-ws.org/Vol-3826/
24. Balatska, V., & Dmytriv, N. (2025). Interorganizational exchange of confidential personal data based on permissioned blockchain. *Cybersecurity: Education, Science, Technique, 1*(29), 178–193. https://doi.org/10.28925/2663-4023.2025.29.875
25. Balatska, V. (2026). Blockchain-oriented approach to ensuring traceability and verifiability of information protection system policy implementation. *Cybersecurity: Education, Science, Technique, 4*(32), 674–685. https://doi.org/10.28925/2663-4023.2026.32.1136
26. Balatska, V. S., Ivanusa, A. I., & Panovyk, U. M. (2025). Method for integrating information security policies, standards, and protocols into the process of building a comprehensive information protection system in an organization. *Cybersecurity: Education, Science, Technique, 3*(31), 283–297. https://doi.org/10.28925/2663-4023.2025.31.1021

КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 2 (34), 2026
ISSN 2663 – 4023

Valeriia Balatska
Doctor of Philosophy, Associate Professor,
Senior Lecturer of the Department of Information Security Management,
Lviv State University of Life Safety, Lviv, Ukraine
ORCID: 0000-0002-6262-6792
v.balatska@ldubgd.edu.ua

METHOD FOR FORMING AN INTER-REGISTRY INTERACTION PROTOCOL WITH MINIMAL PERSONAL DATA DISCLOSURE

Abstract. The article addresses the scientific and practical problem of excessive personal data disclosure during interactions between government electronic registries. In many cases, an administrative or legal decision requires only confirmation that a specified condition has been satisfied; nevertheless, the requesting system receives the value of the original attribute or a fragment of the registry record. This approach increases the amount of data available to the verifier, complicates control over its subsequent use, and creates additional risks of correlating information across different information systems. The aim of the study is to develop a method for constructing a policy-dependent inter-registry interaction protocol that verifies whether a request is permitted, confirms a specified predicate without disclosing the original attribute, and creates a verifiable audit record without storing personal data or persistent correlation data in a permissioned blockchain. The method combines a versioned minimum-disclosure profile; validation of the purpose, legal basis, audience, validity period, and predicate parameters; binding of the ZK proof to the context of a specific operation; and the generation of a keyed audit commitment. Only the minimum information required for subsequent integrity verification is recorded in the blockchain, whereas the complete evidentiary data are retained in a protected off-chain repository. The formal model is presented for a range predicate and supplemented with decision-acceptance conditions, replay protection, and policy freshness validation. Domain-specific conditions are checked before cryptographic operations are performed, preventing a cryptographically valid proof from being accepted for an unauthorised purpose. The study also defines the composition of on-chain and off-chain data, the procedure for recomputing the audit commitment, and scenarios for detecting proof modification, nonce reuse, outdated policy versions, and stale registry assertions. The reference prototype was implemented on the secp256k1 curve using Pedersen commitments, bit decomposition, non-interactive OR proofs, Ed25519, and HMAC-SHA-256. After 15 warm-up runs, 150 timing measurements and 2,000 functional tests across ten classes were conducted. The median proof generation time was 30.587 ms, while the median verification time was 34.663 ms; the corresponding 95th percentiles were 32.939 ms and 37.003 ms. The serialised proof size was 7,185 bytes. All valid requests were accepted, while violations involving the purpose, parameters, audience, validity period, policy version, status freshness, nonce uniqueness, and data integrity were detected. For the evaluated predicate, the original attribute was not transmitted to the verifier. The principal scientific contribution is the formalisation of an end-to-end relationship between an authorised purpose, a minimal predicate, a context-bound proof, and a private audit anchor. The proposed method can be used as a control layer between a policy service, a source registry, and a permissioned blockchain.

Keywords: government electronic registries; inter-registry interaction; personal data; data minimisation; zero-knowledge proof; permissioned blockchain; protocol; access policy; audit commitment.

REFERENCES (TRANSLATED AND TRANSLITERATED)

1. Verkhovna Rada of Ukraine. (2010). *Pro zakhyst personalnykh danykh* [On personal data protection] (Law of Ukraine No. 2297-VI, June 1, 2010). https://zakon.rada.gov.ua/laws/show/2297-17#Text
2. Verkhovna Rada of Ukraine. (2021). *Pro publichni elektronni reiestry* [On public electronic registers] (Law of Ukraine No. 1907-IX, November 18, 2021). https://zakon.rada.gov.ua/laws/show/1907-20#Text
3. European Parliament & Council of the European Union. (2016). Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 (General Data Protection Regulation). *Official Journal of the European Union, L 119*, 1–88. https://eur-lex.europa.eu/eli/reg/2016/679/oj

4. International Organization for Standardization & International Electrotechnical Commission. (2026). *Information security, cybersecurity and privacy protection – Guidelines on privacy preservation based on zero-knowledge proofs* (ISO/IEC 27565:2026). ISO. https://www.iso.org/standard/80398.html
5. International Organization for Standardization & International Electrotechnical Commission. (2021). *Information security, cybersecurity and privacy protection – Requirements for attribute-based unlinkable entity authentication* (ISO/IEC 27551:2021). ISO. https://www.iso.org/standard/72018.html
6. World Wide Web Consortium. (2025). *Verifiable credentials data model v2.0* (W3C Recommendation, May 15, 2025). https://www.w3.org/TR/vc-data-model-2.0/
7. OpenID Foundation. (2025). *OpenID for verifiable presentations 1.0* (OpenID Final Specification, July 9, 2025). https://openid.net/specs/openid-4-verifiable-presentations-1_0.html
8. Bernstein, G., & Sporny, M. (2026). *Data integrity BBS cryptosuites v1.0* (W3C Candidate Recommendation Draft, April 7, 2026). World Wide Web Consortium. https://www.w3.org/TR/vc-di-bbs/
9. Flamini, A., Sciarretta, G., Scuro, M., Sharif, A., Tomasi, A., & Ranise, S. (2024). On cryptographic mechanisms for the selective disclosure of verifiable credentials. *Journal of Information Security and Applications, 83*, Article 103789. https://doi.org/10.1016/j.jisa.2024.103789
10. Podda, E., Hölzmer, P., Amard, A., Sedlmeir, J., & Fridgen, G. (2025). The impact of zero-knowledge proofs on data minimisation compliance of digital identity wallets. *Internet Policy Review, 14*(3), 1–29. https://doi.org/10.14763/2025.3.2019
11. Pedersen, T. P. (1992). Non-interactive and information-theoretic secure verifiable secret sharing. In *Advances in Cryptology – CRYPTO '91* (Lecture Notes in Computer Science, Vol. 576, pp. 129–140). Springer. https://doi.org/10.1007/3-540-46766-1_9
12. Cramer, R., Damgård, I., & Schoenmakers, B. (1994). Proofs of partial knowledge and simplified design of witness hiding protocols. In *Advances in Cryptology – CRYPTO '94* (Lecture Notes in Computer Science, Vol. 839, pp. 174–187). Springer. https://doi.org/10.1007/3-540-48658-5_19
13. Fiat, A., & Shamir, A. (1987). How to prove yourself: Practical solutions to identification and signature problems. In *Advances in Cryptology – CRYPTO '86* (Lecture Notes in Computer Science, Vol. 263, pp. 186–194). Springer. https://doi.org/10.1007/3-540-47721-7_12
14. Rundgren, A., Jordan, B., & Erdtman, S. (2020). *JSON Canonicalization Scheme (JCS)* (RFC 8785). Internet Engineering Task Force. https://www.rfc-editor.org/rfc/rfc8785
15. Josefsson, S., & Liusvaara, I. (2017). *Edwards-Curve Digital Signature Algorithm (EdDSA)* (RFC 8032). Internet Engineering Task Force. https://www.rfc-editor.org/rfc/rfc8032
16. Krawczyk, H., Bellare, M., & Canetti, R. (1997). *HMAC: Keyed-hashing for message authentication* (RFC 2104). Internet Engineering Task Force. https://www.rfc-editor.org/rfc/rfc2104
17. Rescorla, E. (2026). *The Transport Layer Security (TLS) Protocol Version 1.3* (RFC 9846). Internet Engineering Task Force. https://doi.org/10.17487/RFC9846
18. Androulaki, E., Barger, A., Bortnikov, V., et al. (2018). Hyperledger Fabric: A distributed operating system for permissioned blockchains. In *Proceedings of the Thirteenth EuroSys Conference* (Article 30, pp. 1–15). ACM. https://doi.org/10.1145/3190508.3190538
19. Putz, B., Menges, F., & Pernul, G. (2019). A secure and auditable logging infrastructure based on a permissioned blockchain. *Computers & Security, 87*, Article 101602. https://doi.org/10.1016/j.cose.2019.101602
20. National Institute of Standards and Technology. (2020). *NIST Privacy Framework: A tool for improving privacy through enterprise risk management* (Version 1.0). https://www.nist.gov/privacy-framework/privacy-framework
21. Balatska, V. S., & Opirskyy, I. R. (2023). Ensuring personal data confidentiality and cybersecurity support using blockchain. *Cybersecurity: Education, Science, Technique, 4*(20), 6–19. https://doi.org/10.28925/2663-4023.2023.20.619
22. Balatska, V., & Poberezhnyk, V. (2024). Concept of using blockchain technologies to enhance personal data security of the Diia platform: Compliance with GDPR and Ukrainian legislation. *Cybersecurity: Education, Science, Technique, 2*(26), 268–290. https://doi.org/10.28925/2663-4023.2024.26.681
23. Balatska, V., Slobodian, N., & Opirskyy, I. (2024). Blockchain for enhancing transparency and trust in government registries. *CEUR Workshop Proceedings, 3826*, 50–59. https://ceur-ws.org/Vol-3826/
24. Balatska, V., & Dmytriv, N. (2025). Interorganizational exchange of confidential personal data based on permissioned blockchain. *Cybersecurity: Education, Science, Technique, 1*(29), 178–193. https://doi.org/10.28925/2663-4023.2025.29.875

25. Balatska, V. (2026). Blockchain-oriented approach to ensuring traceability and verifiability of information protection system policy implementation. *Cybersecurity: Education, Science, Technique, 4*(32), 674–685. https://doi.org/10.28925/2663-4023.2026.32.1136
26. Balatska, V. S., Ivanusa, A. I., & Panovyk, U. M. (2025). Method for integrating information security policies, standards, and protocols into the process of building a comprehensive information protection system in an organization. *Cybersecurity: Education, Science, Technique, 3*(31), 283–297. https://doi.org/10.28925/2663-4023.2025.31.1021

Отримано редакцією журналу / Received: 04.03.26
Прорецензовано / Revised: .25.03.26
Схвалено до друку / Accepted: 24.09.26