КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 1 (33), 2026
ISSN 2663 – 4023
DOI 10.28925/2663-4023.2026.33.1231
УДК 004.056:004.738.5

Овсянко Дмитро Олексійович
аспірант кафедри безпеки інформаційних технологій
Національний університет «Львівська політехніка», Львів, Україна
ORCID: 0009-0003-7758-0613
dmytro.o.ovsianko@lpnu.ua

Нємкова Олена Анатоліївна
д.т.н., професор, професор кафедри безпеки інформаційних технологій
Національний університет «Львівська політехніка», Львів, Україна
ORCID: 0000-0003-0690-2657
olena.a.niemkova@lpnu.ua

КОНЦЕПТУАЛЬНА АРХІТЕКТУРА ТА ФОРМАЛЬНА МОДЕЛЬ САМОСУВЕРЕННИХ ЦИФРОВИХ ДВІЙНИКІВ В IoT-ЕКОСИСТЕМАХ

Анотація. У статті представлено концептуальну архітектуру самосуверенних цифрових двійників (SSDT) для IoT-екосистем, що забезпечує децентралізоване управління ідентичністю та даними пристроїв незалежно від централізованих провайдерів. Запропоноване рішення усуває основні недоліки традиційних IoT-систем, що пов'язані з централізованим зберіганням даних, ризиками компрометації провайдера та відсутністю гарантій приватності. Розроблено трирівневу архітектуру: фізичний рівень забезпечує автентичний збір даних на IoT-пристроях з криптографічним підписанням; рівень цифрового двійника на обчислювальному шлюзі реалізує управління децентралізованими ідентифікаторами (DID), зберігання облікових даних, оцінювання політик доступу та генерацію доказів з нульовим розголошенням; блокчейн-рівень гарантує незмінний аудит через приватний блокчейн з аварійностійким консенсусом, смарт-контрактами для реєстру DID, управління статусами облікових даних та журналювання операцій доступу. Формалізовано модель SSDT як кортеж, що включає децентралізований ідентифікатор, множину атрибутів, функцію стану, політики доступу, криптографічні ключі та історію операцій, з чітко визначеними інваріантами безпеки. Проаналізовано модель загроз на основі методології STRIDE, адаптованої до розподілених IoT-систем. Аналіз охоплює ключові активи (ідентичності, облікові дані, телеметрію та приватні ключі) і згруповано загрози за напрямами: цілісність обміну, загрози ідентичності та конфіденційність. Ідентифіковано атаки типу Man-in-the-Middle, replay-атаки, підміну пристроїв, підробку облікових даних, обхід політик доступу та компрометацію ключів. Запропоновано комплекс контрзаходів, що включає mutual TLS, криптографічне підписання повідомлень, часові мітки, обов'язкову реєстрацію DID у блокчейні, атестацію пристроїв, перевірку статусу відкликання, обмеження частоти запитів, ротацію ключів та апаратні модулі безпеки. Для забезпечення приватності використано механізми доказів з нульовим розголошенням. Результати дослідження підтверджують можливість створення масштабованих, приватних і самосуверенних систем управління IoT-пристроями. Архітектура забезпечує горизонтальну масштабованість, низьку латентність завдяки периферійній обробці та приватність за проєктуванням. Практична цінність полягає в можливості застосування в індустріальному IoT, персональних моніторингових системах та розумних містах. Напрямки подальших досліджень включають формальну верифікацію, оптимізацію ZKP для ресурсозалежних пристроїв та сумісність з існуючими IoT-платформами.

Ключові слова: цифрові двійники; децентралізовані ідентифікатори; Інтернет речей; блокчейн; верифіковані облікові дані; докази з нульовим розголошенням; апаратні модулі безпеки; приватність даних; розподілені системи.

ВСТУП

Стрімке зростання екосистем Інтернету речей (IoT) породжує суттєві виклики в управлінні ідентичністю та даними мільярдів підключених пристроїв. Традиційні централізовані підходи демонструють фундаментальні обмеження щодо масштабованості, забезпечення приватності та стійкості до відмов. Концепція цифрових двійників набула значного поширення в індустріальних застосуваннях, проте більшість існуючих реалізацій базується на централізованих хмарних платформах, які здійснюють повний контроль над доступом до даних і ідентифікацією пристроїв. Така централізація створює єдині точки відмови, суттєво обмежує автономію власників пристроїв і породжує серйозні проблеми приватності.

Модель самосуверенної ідентичності (Self-Sovereign Identity, SSI) пропонує принципово альтернативний підхід, за якого індивіди (або пристрої) зберігають повний контроль над своїми ідентифікаційними даними без залучення будь-яких посередників. Ключовими технологічними елементами SSI є децентралізовані ідентифікатори (DID), верифіковані облікові дані (Verification Credentials, VC, далі – облікові дані) та довірчі блокчейни. Застосування принципів SSI до середовища IoT відкриває можливість створення самосуверенних цифрових двійників (Self-Sovereign Digital Twins, SSDT), які поєднують переваги традиційних цифрових двійників із гарантіями автономії та захисту приватності.

Водночас адаптація технологій SSI до контексту IoT стикається з низкою унікальних викликів: обмеженими обчислювальними ресурсами пристроїв, масштабами екосистем, що налічують мільярди пристроїв, критичними вимогами до обробки даних у реальному часі та гетерогенністю апаратних платформ. Існуючі дослідження цифрових двійників приділяють недостатню увагу децентралізованому управлінню ідентичністю [1], роботи в галузі SSI переважно зосереджені на людській ідентичності [2], а блокчейн-рішення для IoT рідко інтегрують повний стек компонентів SSI [3].

Аналіз останніх досліджень і публікацій. Проблема децентралізованого управління ідентичністю та даними IoT-пристроїв через концепцію самосуверенних цифрових двійників перетинається з кількома активними напрямками досліджень, кожен з яких вніс значний внесок у розуміння окремих аспектів цієї проблеми.

Фундаментальні роботи Grieves та Vickers заклали концептуальну основу цифрових двійників як віртуальних представлень фізичних об’єктів з двонапрямним обміном даними [4]. Tao et al. розширили цю концепцію для розумного виробництва, демонструючи практичну цінність цифрових двійників для оптимізації виробничих процесів через моніторинг у реальному часі [1]. Проте ці дослідження фокусуються виключно на функціональних аспектах моделювання, залишаючи поза увагою критичні питання власності даних та автономії управління. Atzori et al. систематизували виклики IoT-екосистем, включаючи масштабованість та гетерогенність пристроїв [5], але не запропонували рішень для децентралізованого управління ідентичністю.

Концепція самосуверенної ідентичності, систематизована Mühle et al., визначила принципи орієнтованого на користувача управління без централізованих провайдерів [2]. Стандарти World Wide Web Consortium (W3C) для децентралізованих ідентифікаторів та облікових даних забезпечили взаємодію між реалізаціями SSI [6], [7] та створили технологічну основу децентралізованої ідентифікації. Chaum запропонував механізми некорельованих псевдонімів для автентифікації зі збереженням приватності [8], а Camenisch та Lysyanskaya розробили ефективні схеми для анонімних облікових даних із селективним розкриттям [9]. Проте зазначені роботи зосереджені переважно на сценаріях людської ідентичності. Вони не враховують специфічні обмеження IoT-пристроїв. До таких обмежень належать обмеженість обчислювальних ресурсів, вимоги обробки даних у реальному часі та масштаб екосистем з мільярдами пристроїв.

Застосування блокчейн-технологій для IoT досліджувалось Vukolić, який порівняв Proof-of-Work та візантійсько-стійкі механізми консенсусу в контексті масштабованості [10]. Androulaki et al. представили Hyperledger Fabric як модульну платформу зі змінними реалізаціями консенсусу [3], що створило архітектурну основу для приватних блокчейнів у корпоративних сценаріях. Проте жодна з цих робіт не інтегрує повний стек компонентів SSI (DID, облікові дані, селективне розкриття) в єдину архітектуру для цифрових двійників.

Механізми збереження приватності на основі ZKP, теоретична основа яких закладена Goldwasser et al. [11], отримали практичну реалізацію через короткі неінтерактивні аргументи знання (SNARKs) від Ben-Sasson et al. [12] та Bulletproofs від Bünz et al. [13]. Kosba et al. продемонстрували інтеграцію ZKP у смарт-контракти через Hawk framework [14], доводячи можливість поєднання прозорості блокчейну з приватністю транзакцій. Однак обчислювальні накладні витрати цих механізмів та їх практична інтеграція в IoT-середовища з обмеженими ресурсами та цифровими двійниками залишається

недостатньо дослідженою.

Модель периферійних обчислень, визначена Shi et al. [15], та концепція туманних обчислень Bonomi et al. [16] створили архітектурну основу для локальної обробки даних з низькою латентністю. Satyanarayanan et al. запропонували архітектуру cloudlet як проміжний рівень між пристроями та хмарою [17], а Yi et al. [18] систематизували виклики туманних обчислень, включаючи безпеку та сумісність. Ці роботи демонструють переваги підходу з пріоритетом периферії, проте не розглядають, як периферійна інфраструктура може підтримувати самосуверенні цифрові двійники з децентралізованим управлінням ідентичністю.

Постановка проблеми. Аналіз існуючих досліджень виявив, що жодна з робіт не пропонує цілісної архітектури, яка б інтегрувала: (1) функціональність цифрових двійників для IoT з (2) децентралізованим управлінням ідентичністю через технології SSI, (3) гарантіями приватності через докази з нульовим розголошенням, (4) незмінним аудитом через блокчейн, та (5) низькою латентністю через периферійні обчислення. Зокрема, залишаються невирішеними питання адаптації технологій SSI під обмеження ресурсів IoT-пристроїв, інтеграції механізмів селективного розкриття в архітектуру цифрових двійників, балансу між зберіганням у блокчейні та поза блокчейном для масштабованості, та забезпечення самосуверенності при збереженні реального часу для критичних операцій.

У межах цього дослідження запропоновано концептуальну трирівневу архітектуру самосуверенних цифрових двійників, спрямовану на усунення зазначених прогалин. Розроблена архітектура забезпечує верифікований збір даних на фізичному рівні, впровадження децентралізованих механізмів управління ідентичністю та контроль доступу з дотриманням принципів приватності на рівні обчислювального шлюзу. Достовірність та незмінність аудиту гарантується використанням приватного блокчейн.

Метою статті є розроблення концептуальної архітектури самосуверенних цифрових двійників (SSDT) для екосистем Інтернету речей (IoT), яка забезпечує децентралізоване управління ідентичністю, захист приватності даних, високу масштабованість і збереження низької латентності критичних операцій, зазвичай до 10-50 мс.

Для досягнення зазначеної мети вирішуються такі задачі:

- формалізація моделі SSDT з чітким визначенням її компонентів та інваріантів безпеки;
- розроблення трирівневої архітектури системи;
- аналіз моделі загроз та формування комплексу контрзаходів;
- проектування механізмів забезпечення приватності на основі доказів з нульовим розголошенням (ZKP);
- розроблення гібридної архітектури зберігання даних та створення прототипної реалізації (PoC) для експериментальної верифікації запропонованих рішень.

Наукова новизна роботи полягає в розробленні концептуальної архітектури, яка вперше інтегрує повний стек технологій самосуверенної ідентичності з цифровими двійниками IoT-пристроїв. Запропонована архітектура забезпечує децентралізоване управління ідентичністю за допомогою периферійних обчислень, гарантії приватності завдяки механізмам доказів з нульовим розголошенням, а також незмінність і верифікованість аудиту через використання дозвільного блокчейну. Практична цінність отриманих результатів полягає в можливості створення масштабованих IoT-систем із гарантіями автономії власників пристроїв. Такі системи є особливо важливими для індустріального Інтернету речей, інфраструктури розумних міст та персональних систем моніторингу стану здоров’я.

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Концепція системи SSDT. Традиційні цифрові двійники зазвичай керуються централізованими сервісами, що створює проблеми власності даних, єдиних точок відмови та обмеженого контролю для фактичного власника пристрою або даних. Концепція самосуверенного цифрового двійника пропонує новий підхід до управління цифровими двійниками, надаючи повний контроль користувачам або пристроям над їхніми цифровими представленнями та даними. SSDT визначається як цілісне цифрове представлення сутності (людини, машини, IoT-пристрою), яке управляється безпосередньо власником, а не централізованими службами. Ця концепція наділяє пристрої та сутності власною ідентичністю, контролем над даними та можливістю селективного розкриття інформації через криптографічно захищені механізми. На відміну від традиційних підходів, де цифровий двійник є пасивним представленням, SSDT функціонує як автономний агент, здатний самостійно приймати рішення щодо доступу до власних даних та взаємодії з зовнішніми сервісами.

Основні принципи SSDT базуються на трьох ключових засадах. По-перше, принцип власності даних визначає, що користувачі зберігають повний контроль над доступом до свого SSDT, включаючи

право визначати, які дані розкриваються, кому і за яких умов. По-друге, децентралізація через використання DIDs та блокчейн-реєстрів усуває необхідність у центральних органах управління, пом'якшуючи ризики єдиних точок атак та відмови. По-третє, принципи криптографічної безпеки забезпечується вбудованими ключами та протоколами, які гарантують надійну автентифікацію та авторизацію без залежності від третіх сторін [19]. Також, підвищення приватності досягається через механізми селективного розкриття та докази з нульовим розголошенням, які дозволяють підтверджувати коректність атрибутів без розкриття повного набору даних.

Архітектура SSDT організована як трирівнева система з чітким розділенням відповідальності, де кожен рівень відповідає за специфічну функціональність та взаємодіє з іншими рівнями через стандартизовані протоколи. Фізичний рівень представлений IoT-пристроями, що виконують збір первинних даних, локальне підписання телеметрії через ключ, згенерований в апаратному модулі, та передачу підписаних даних через захищені протоколи до вищого рівня. При цьому, кожне повідомлення включає часову мітку з точністю до мілісекунд, та номер послідовності для виявлення втрат пакетів [20]. Рівень цифрового двійника є центральним компонентом архітектури, на якому розміщено самосуверенний цифровий двійник з повним набором функціональності: управління DID за допомогою локального криптографічного гаманця з ключами, захищеними модулем довіреної платформи (Trusted Platform Module, TPM), зберігання та управління обліковими даними, підтримка поточного стану з агрегацією телеметрії від множинних IoT-пристроїв та збереженням історії станів, реалізація поведінкових моделей, виконання політик доступу, генерація доказів з нульовим розголошенням, та взаємодія з блокчейном. Блокчейн-рівень містить ключові смарт-контракти, що реалізують критичну бізнес-логіку: реєстрації та розв'язання децентралізованих ідентифікаторів, управління життєвим циклом облікових даних з підтримкою масового відкликання при компрометації емітента, та верифікації запитів доступу через перевірку та логування всіх операцій доступу.

Формальна модель самосуверенного цифрового двійника визначається як кортеж

$$SSDT=(DID,A,S,P,K,H),$$

де кожен компонент представляє ключовий аспект цифрового двійника [21] (Рис. 1).

Рис. 1. Формальна модель самосуверенного цифрового двійника

Компонент DID є децентралізованим ідентифікатором, що унікально ідентифікує цифровий двійник у глобальному просторі імен. Компонент A представляє множину атрибутів цифрового двійника, які можуть включати статичні характеристики та динамічні властивості, причому кожен атрибут може бути асоційований з обліковими даними для підтвердження його автентичності. Компонент S описує поточний стан цифрового двійника, який є функцією часу S(t) та відображає актуальні значення всіх динамічних атрибутів у момент часу t. Компонент P представляє множину політик доступу, які визначають умови, за яких зовнішні сутності можуть отримати доступ до атрибутів або стану цифрового двійника [22]. Компонент K включає криптографічні ключі для підписання даних, автентифікації та шифрування. Компонент H представляє історію станів цифрового двійника з часовою міткою та криптографічними підписами для забезпечення цілісності [23].

Розглянута архітектура SSDT містить в собі п'ять ключових ролей, кожна з яких виконує конкретні функції в системі (Рис. 2). Роль IoT-пристрою на фізичному рівні полягає в зборі даних із навколишнього середовища за допомогою вбудованих сенсорів, забезпеченні автентичності зібраних даних через криптографічні підписи (які генеруються на рівні самого пристрою), виконанні команд, що надходять з вищого рівня, за допомогою вбудованих актуаторів, а також у безпечній передачі даних на вищий рівень архітектури через захищені канали зв'язку.

ISSN 2663 – 4023

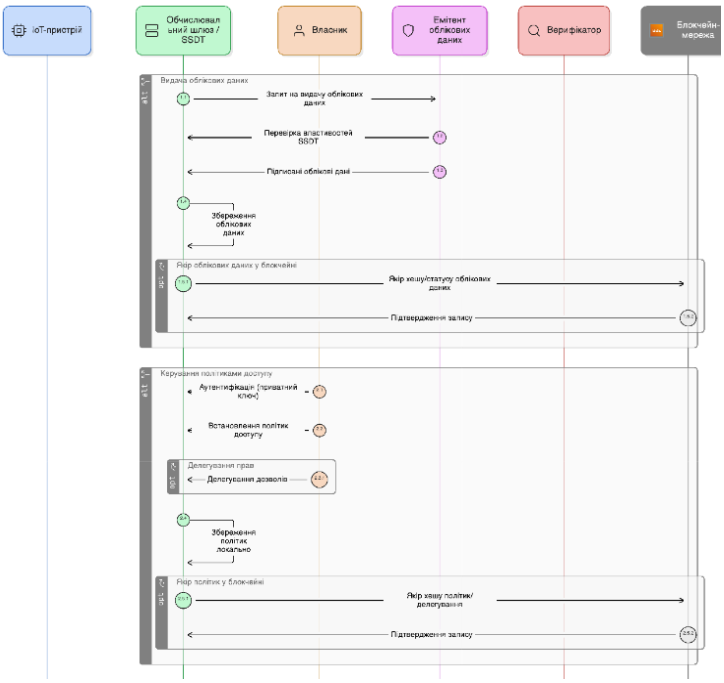

Рис.2. Діаграма активностей в моделі SSDT

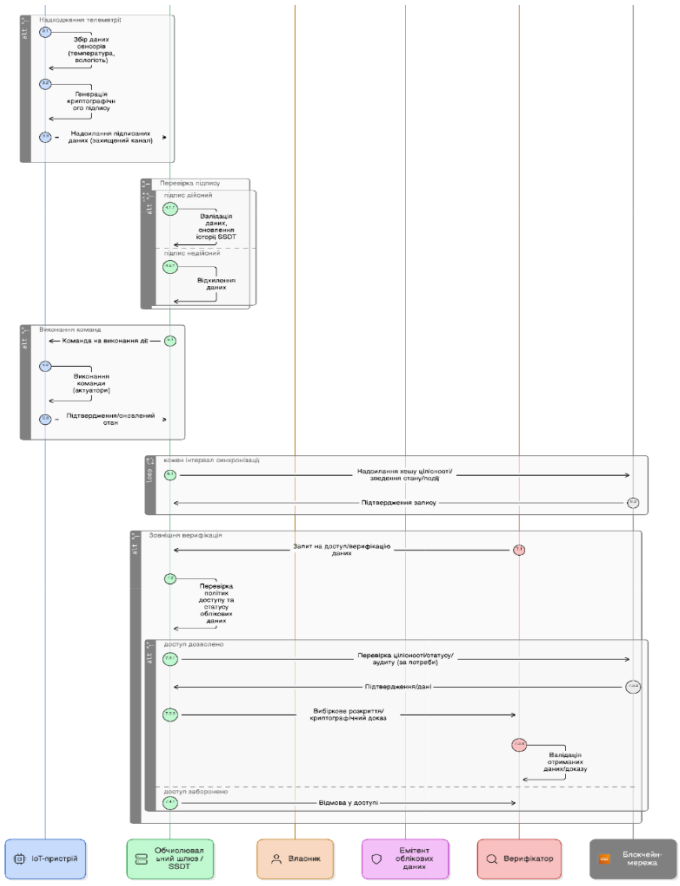

Рис.2 (*Продовження*). *Діаграма активностей в моделі SSDT*

Роль периферійного обчислювального шлюзу виконують обчислювальні вузли рівня цифрового двійника. На цих вузлах розміщуються екземпляри SSDT. Вони відповідають за керування DID-

ідентичністю, зберігання облікових даних, ведення стану та історії, виконання політик доступу і взаємодію з блокчейн-мережею. Роль власника представлена сутністю, яка має повний контроль над двійником. Контроль здійюснється за допомогою приватного ключа. Власник приймає рішення щодо політик доступу і делегування повноважень. Роль верифікатора виконується зовнішніми сервісами, які формують запити доступу до даних SSDT, верифікують надані криптографічні докази та використовують дані відповідно до узгоджених умов. Роль емітента облікових даних виконується довіреними організаціями, які верифікують властивості SSDT та видають підписані облікові дані з можливістю їх подальшого відкликання.

Функціональні вимоги до системи визначають ключові операції, що мають бути реалізовані. Система повинна забезпечувати:

- створення децентралізованих ідентифікаторів відповідно до W3C DID Core та їх реєстрацію в блокчейні;
- агрегацію і верифікацію даних від множинних IoT-пристроїв з автентифікацією джерела;
- делегування доступу через видачу облікових даних із обмеженнями та вбудованими ключами шифрування;
- багатоетапну перевірку прав доступу, що охоплює валідацію структури облікових даних, перевірку цифрових підписів емітентів, терміну дії та статусу відкликання;
- негайне відкликання раніше виданих повноважень шляхом оновлення статусу в блокчейні.

Нефункціональні вимоги визначають якісні характеристики системи. Вони охоплюють приватність, безпеку, продуктивність, масштабованість та енергоефективність. Приватність забезпечується вибірковим розкриттям атрибутів із використанням доказів з нульовим розголошенням, мінімізацією обсягу даних, що публікуються у блокчейні, шляхом розміщення криптографічних зобов'язань замість повних наборів даних, а також псевдонімізацією ідентифікаторів для запобігання кореляції активності. Безпека передбачає автентифікацію на рівні застосунку за допомогою криптографічних підписів, забезпечення цілісності даних через криптографічне хешування, конфіденційність через симетричне шифрування з унікальними ключами, а також захист приватних ключів із використанням апаратних модулів безпеки (Hardware Security Modules, HSM) із унеможливлення екстракції ключа. Продуктивність вимагає, щоб затримка критичних операцій не перевищувала кількох секунд, зокрема для розв'язання ідентифікаторів, генерації криптографічних доказів і наскрізної (end-to-end) обробки запитів доступу. Масштабованість досягається підтримкою розподіленої архітектури, у якій кожен SSDT функціонує автономно, горизонтальною масштабованістю блокчейн-мережі та використанням ефективних структур даних із логарифмічною складністю верифікації. Енергоефективність є критичною для IoT-пристроїв із батарейним живленням і потребує застосування режимів низького енергоспоживання між циклами збору даних, а також апаратних криптографічних прискорювачів для зменшення обчислювальних витрат.

Модель загроз та припущення безпеки. Модель загроз для описаної системи сформовано на основі методології STRIDE (Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privilege), адаптованої до розподілених IoT-систем [22]. Аналіз фокусується на ключових активах системи – ідентичностях, облікових даних, телеметрії від IoT-пристроїв, а також приватних ключах, що забезпечують довіру між компонентами. Відповідно, загрози згруповано за трьома напрямами: (i) цілісність і автентичність обміну, (ii) загрози ідентичності, (iii) конфіденційність і контроль доступу (табл. 1).

Таблиця 1

Загрози за методологією STRIDE та відповідні запобіжні заходи

№	Напрям	Категорії загроз	Контрзаходи
i	Загрози цілісності та автентичності обміну	атаки типу *Man-in-the-Middle*	взаємна автентифікація на рівні mutual TLS для всіх мережевих з'єднань; криптографічне підписання повідомлень на рівні застосунку
		replay-атаки	включення до кожного повідомлення часової мітки та номера послідовності з подальшою перевіркою на боці отримувача
ii	Загрози ідентичності	spoofing (підміна ідентичності IoT-пристрою або SSDT)	обов'язкова реєстрація DID у блокчейн-реєстрі з перевіркою унікальності та автентичності; атестація пристрою; використання апаратних модулів безпеки; механізми списку дозволених; допуск лише пристроїв із чинними сертифікатами

Продовження таблиці 1

		підробка облікових даних	перевірка цифрових підписів емітентів; валідація ланцюжка довіри до центру сертифікації; перевірка статусу відкликання через блокчейн-реєстр
iii	Загрози конфіденційності та контролю доступу	обхід політик доступу; експлуатація помилок у механізмах авторизації	комплексна верифікація облікових даних: перевірка структури, підписів, строків дії, статусу відкликання та доказу володіння ключем
		масовані / автоматизовані спроби доступу	обмеження частоти запитів (*rate limiting*)
		компрометація приватних ключів	зберігання ключів лише в апаратних модулях безпеки з властивістю невитягуваності; регулярна ротація ключів; багатофакторна автентифікація для критичних операцій

Загрози цілісності та автентичності обміну. У каналах взаємодії між компонентами релевантними є атаки типу Man-in-the-Middle, за яких зловмисник може перехоплювати та модифікувати дані під час передавання [24]. Для зменшення цієї загрози застосовується взаємна автентифікація на рівні mutual TLS для всіх мережевих з'єднань, а також криптографічне підписання повідомлень на рівні застосунку, що забезпечує наскрізну автентичність незалежно від транспортного рівня [25]. Окремий клас становлять replay-атаки, коли перехоплене повідомлення повторно надсилається з метою введення системи в оману. Контрзаходи реалізуються шляхом включення до кожного повідомлення часової мітки та номера послідовності з подальшою перевіркою на боці отримувача [26].

Загрози ідентичності. У межах моделі загроз STRIDE для категорії Spoofing (підміна ідентичності) критичною загрозою є імітація легітимних суб'єктів – як фізичних IoT-пристроїв, так і екземплярів SSDT. Це може призвести до несанкціонованого доступу до системи або неправомірного підвищення привілеїв. Зменшення ризику забезпечується обов'язковою реєстрацією DID у блокчейн-реєстрі з перевіркою унікальності та автентичності, а також атестацією пристрою (device attestation) з використанням апаратних модулів безпеки для криптографічного підтвердження виконання SSDT на довіреному обладнанні з валідною прошивкою. Додатково застосовуються механізми "списку дозволених" на рівні протоколів взаємодії, які обмежують підключення лише авторизованими пристроями з чинними сертифікатами [27]. Ще одна суттєва загроза – підробка облікових даних з метою отримання доступу поза політиками системи. Вона нейтралізується перевіркою цифрових підписів емітентів, валідацією ланцюжка довіри до центру сертифікації (CA), а також перевіркою статусу відкликання через запити до блокчейн-реєстру.

Загрози конфіденційності та контролю доступу. Спроби несанкціонованого доступу можуть реалізовуватися як через обхід політик доступу, так і через експлуатацію помилок у механізмах авторизації. Тому система застосовує комплексну верифікацію облікових даних: багаторівневу перевірку запитів: валідацію структури облікових даних, перевірку підписів, строків дії, статусу відкликання та доказу володіння ключем (proof of possession). Для зменшення ризику підбору/масованих спроб доступу додатково використовується обмеження частоти запитів (rate limiting). Найбільш критичною залишається компрометація приватних ключів, оскільки вона дає змогу повністю імітувати легітимного власника SSDT [28]. Контрзаходи базуються на стратегії управління життєвим циклом криптографічних ключів, що включає: зберігання ключів виключно в апаратних модулях безпеки з властивістю невитягуваності (non-extractable), регулярну ротацію ключів для зменшення "вікна вразливості" у разі компрометації, а також багатофакторну автентифікацію для критичних операцій.

Припущення безпеки. Модель базується на таких припущеннях: (i) апаратні модулі безпеки є довіреними та коректно реалізують властивість невитягуваності ключового матеріалу; (ii) блокчейн-мережа забезпечує Byzantine Fault Tolerance за умови не більше однієї третини зловмисних вузлів; (iii) центри сертифікації коректно верифікують ідентичність перед видачею сертифікатів; (iv) використані криптографічні примітиви (Ed25519, AES-256, SHA-256) залишаються стійкими до відомих атак у межах життєвого циклу системи [3, 16, 19].

Загальна архітектура та принципи проектування. Архітектуру цифрового двійника для IoT-пристроїв побудовано як трирівневу децентралізовану систему (Рис. 3), що розділяє відповідальність між (i) збором даних, (ii) керуванням ідентичністю та політиками доступу і (iii) формуванням спільного рівня довіри [29]. Фізичний рівень забезпечує збір телеметрії безпосередньо на IoT-пристроях і криптографічне підтвердження походження даних. Рівень цифрового двійника реалізує повну функціональність

самосуверенного цифрового двійника на обчислювальному шлюзі поблизу пристрою, включаючи локальне керування ідентичністю та автономне ухвалення рішень щодо доступу до даних. Блокчейн-рівень надає блокчейн-реєстр довіри для перевірки ідентичностей і незмінного аудиту критичних операцій. На відміну від централізованих підходів, запропонована архітектура переносить контроль над цифровим двійником від хмарного провайдера до власника пристрою. Це забезпечує автономність роботи за тимчасової недоступності мережі та підвищує приватність завдяки локальному виконанню чутливих операцій без передавання сирих даних до зовнішніх сервісів.

Принцип розділення відповідальності реалізується через чітко визначені інтерфейси взаємодії між рівнями архітектури. Фізичний рівень передає дані на рівень цифрового двійника за протоколом «публікація–підписка», використовуючи транспортне шифрування та цифровий підпис кожного повідомлення для забезпечення наскрізної автентичності джерела телеметрії. Застосування асиметричної криптографії дає змогу перевіряти підписи без доступу до приватних ключів пристроїв [30]. Рівень цифрового двійника взаємодіє з рівнем блокчейну через два типи інтерфейсів: транзакційний – для операцій зміни стану реєстру, та інтерфейс запитів – для читання поточного стану. Кожна транзакція потребує підтвердження з боку кількох незалежних вузлів, що забезпечує досягнення консенсусу та унеможливлює односторонні зміни. Верифіковані дані та криптографічні докази передаються між рівнями у структурованих форматах із вбудованими підписами, що дозволяє будь-якій стороні незалежно перевіряти автентичність і цілісність інформації без довіри до посередників.

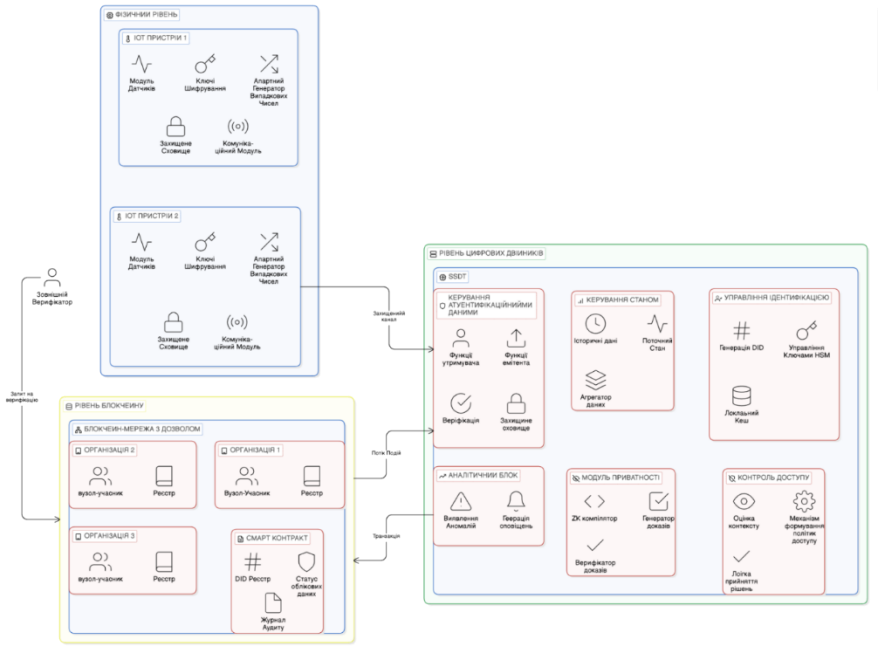

Рис.3. Концептуальна схема архітектури системи

Принцип децентралізації ґрунтується на відмові від централізованих провайдерів ідентичності та єдиних точок контролю. Кожен SSDT функціонує як автономний агент із власною децентралізованою ідентичністю, яка не залежить від централізованих реєстрів чи центрів сертифікації та може бути перевірена будь-якою стороною на основі криптографічних доказів. Приватні ключі зберігаються виключно на обчислювальному шлюзі в апаратних модулях безпеки з властивістю невитягуваності, що унеможливлює їх вилучення навіть за фізичного доступу до обладнання та гарантує, що операції від імені SSDT може виконувати лише легітимний власник. Розподілений реєстр довіри підтримується консорціумом організацій без можливості одностороннього переписування історії; узгодження стану забезпечується візантійсько-стійкими алгоритмами, які зберігають коректність навіть за наявності зловмисних учасників.

Принцип приватності за проєктуванням забезпечується вбудованими механізмами селективного розкриття атрибутів на основі доказів з нульовим розголошенням. Замість передавання повних наборів даних зовнішнім верифікаторам SSDT формує криптографічні докази, які підтверджують коректність певних тверджень про дані без розкриття самих даних. Наприклад, можна довести, що температура перебуває в заданому діапазоні, не розкриваючи точного значення, або що пристрій виготовлено авторизованим виробником, не розкриваючи серійного номера. Такі докази характеризуються стислою

перевіркою (перевірка значно швидша за повторне виконання обчислень над сирими даними) та властивістю нульового розголошення (верифікатор не отримує додаткової інформації, окрім факту істинності твердження). Додатково система підтримує псевдонімізацію шляхом генерації множинних контекстних (попарних) ідентифікаторів для різних сценаріїв взаємодії, що запобігає кореляції активності SSDT між різними сервісами.

Принцип пріоритету периферійних обчислень зменшує залежність від постійного з'єднання з хмарними сервісами та знижує затримку критичних операцій. Керування ідентичністю, зберігання повноважень доступу, застосування політик доступу та генерація криптографічних доказів виконуються локально на обчислювальному шлюзі без звернень до віддалених серверів, що зменшує затримку порівняно з хмарною обробкою. Шлюз підтримує локальне зашифроване сховище операційних даних і кеш для часто використовуваних ідентифікаторів та повноважень, завдяки чому система зберігає працездатність навіть за тимчасової недоступності блокчейну або інших зовнішніх сервісів. Синхронізація з реєстром виконується періодично шляхом публікації криптографічних зобов'язань замість реєстрації кожної окремої операції, що зменшує навантаження на реєстр і підвищує масштабованість.

Принцип аудитованості забезпечується незмінною реєстрацією критичних подій життєвого циклу SSDT у блокчейні. Операції створення або зміни ідентичності, видачі чи відкликання повноважень доступу, а також надання або відмови в доступі до даних формують події, які записуються до реєстру з часовою міткою та криптографічним зв'язком із попередніми записами (хеш-зв'язування). У результаті формується журнал аудиту, в якому будь-яка спроба підміни або видалення історичних записів буде виявлена через порушення хеш-ланцюга, а розподілена природа реєстру забезпечує збереження цих даних на множинних незалежних вузлах. Авторизовані аудитори можуть отримувати відфільтровані подання журналу аудиту із збереженням приватності завдяки механізмам селективного розкриття.

Принцип масштабованості реалізується архітектурними рішеннями, що забезпечують ефективну роботу системи зі зростанням кількості учасників. Кожен SSDT функціонує незалежно на власному обчислювальному шлюзі без спільного стану з іншими SSDT, тому масштабування досягається шляхом додавання нових екземплярів без взаємної координації. Блокчейн-рівень може масштабуватися через сегментацію (поділ на незалежні частини) або використання кількох ізольованих каналів, що дозволяє паралельну обробку незалежних транзакцій різними підмножинами вузлів. Застосування криптографічних зобов'язань замість зберігання повних наборів даних у реєстрі зменшує накладні витрати на зберігання та вимоги до пропускної здатності, оскільки розмір зобов'язання є сталим і не залежить від обсягу первинних даних. Ефективні структури даних, зокрема дерева Меркла, забезпечують логарифмічну складність перевірки: час верифікації зростає логарифмічно, а не лінійно зі збільшенням обсягу даних.

Фізичний рівень збору даних. Фізичний рівень архітектури SSDT забезпечує автентичний збір телеметричних даних безпосередньо на IoT-пристроях, криптографічне підтвердження походження даних і первинну обробку для підвищення якості даних. Ключовою особливістю цього рівня є вбудовані криптографічні можливості, які дозволяють пристроям автономно генерувати пари ключів, підписувати дані та проходити автентифікацію без залежності від зовнішніх сервісів. Розміщення криптографічних операцій на пристрої (замість централізованої автентифікації) забезпечує можливість незалежної перевірки автентичності джерела даних кінцевим отримувачем навіть у разі компрометації мережевої інфраструктури або проміжних вузлів – шляхом перевірки цифрового підпису.

Життєвий цикл IoT-пристрою в системі SSDT починається з ініціалізації під час першого запуску. На цьому етапі генерується криптографічна пара ключів із використанням апаратного генератора випадкових чисел (True Random Number Generator, TRNG), що забезпечує достатню ентропію. Використання TRNG є критичним, оскільки передбачуваність ключів у разі застосування слабких програмних генераторів може призвести до повної компрометації системи. Приватний ключ зберігається в захищеній області пам'яті із шифруванням та обмеженням доступу, що зменшує ризик його вилучення через програмні вразливості або атаки побічними каналами. Публічний ключ використовується для формування ідентифікатора пристрою, який однозначно визначає пристрій у системі та застосовується для перевірки підписів.

Збір телеметрії виконується періодично: пристрій зчитує значення з підключених датчиків і формує структуровані повідомлення. Типове повідомлення містить:

- часову мітку для фіксації часової послідовності подій;
- номер послідовності для виявлення втрат або дублювання пакетів;
- показники датчиків із метаданими (тип вимірювання, одиниці);
- ідентифікатор пристрою як ознаку джерела даних.

Номер послідовності дає змогу виявляти атаки повторного відтворення повідомлень, коли зловмисник надсилає перехоплені дані повторно: отримувач може помітити дублікати або розриви в послідовності. Часова мітка синхронізується через мережевий протокол часу, щоб підтримувати узгодженість часових відміток між розподіленими пристроями.

Підписання повідомлень виконується за схемою «хешування → підпис»: спочатку обчислюється хеш від канонічного подання структури повідомлення, після чого хеш підписується приватним ключем пристрою. Такий підхід робить вартість підписання незалежною від розміру повідомлення та спрощує перевірку. Канонічне подання перед хешуванням є необхідним, оскільки різні варіанти серіалізації одного й того самого логічного повідомлення можуть давати різні хеш-значення і, відповідно, некоректні підписи. Обчислений підпис додається до повідомлення окремим полем, що дозволяє отримувачу незалежно перевірити автентичність за допомогою публічного ключа пристрою.

Взаємодія з рівнем цифрового двійника реалізується через протокол «публікація–підписка», який усуває жорстку прив'язку між джерелами та споживачами даних. IoT-пристрої публікують підписані повідомлення у тематичні канали, не знаючи про конкретних отримувачів, що підтримує гнучку маршрутизацію та паралельне використання тих самих даних кількома споживачами. Транспортний захист каналу зв'язку на основі mTLS забезпечує конфіденційність і захист від підміни на рівні з'єднання; водночас підписи на рівні застосунку додають наскрізну безпеку навіть у разі компрометації транспортного рівня. Механізми якості обслуговування гарантують доставку критичних повідомлень за нестабільного з'єднання завдяки підтвердженням отримання та повторним передаванням.

Енергоефективність є критичною вимогою для IoT-пристроїв із батарейним живленням і безпосередньо впливає на рішення фізичного рівня. Використання режимів низького енергоспоживання між циклами збору даних зменшує середнє енергоспоживання; пробудження виконується апаратними таймерами без потреби постійної активності центрального процесора. Апаратні криптографічні прискорювачі виконують криптографічні операції енергоефективніше, ніж програмні реалізації, що безпосередньо подовжує час автономної роботи. Оптимізація протоколів передавання даних (зменшення розміру корисного навантаження та використання компактних форматів серіалізації) знижує витрати енергії на радіопередавання, яке часто є домінуючою складовою енергоспоживання бездротових IoT-пристроїв.

Рівень цифрового двійника. Рівень цифрового двійника (Рис. 4) є центральним компонентом архітектури SSDT і реалізує повну функціональність самосуверенного цифрового двійника: автономне керування ідентичністю, локальне ухвалення рішень щодо доступу та формування криптографічних доказів. Розміщення цього функціоналу на периферійному обчислювальному шлюзі, а не в хмарі, знижує затримку критичних операцій, підвищує приватність завдяки локальній обробці чутливих даних і забезпечує працездатність навіть за тимчасової недоступності мережі. Обчислювальний шлюз у такій архітектурі виступає довіреною обчислювальною базою SSDT: він виконує критичні для безпеки операції в захищеному середовищі та використовує апаратні механізми захисту приватних ключів.

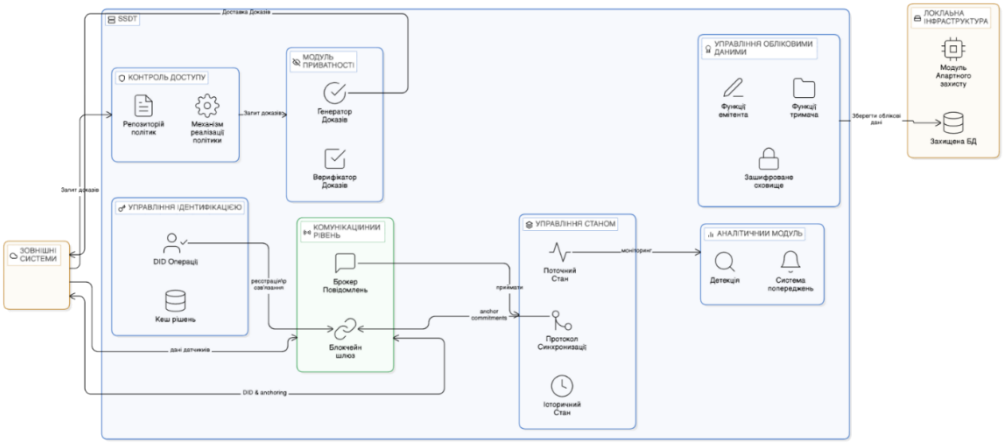

Рис.4. Схема рівня цифрового двійника

Керування децентралізованою ідентичністю охоплює операції життєвого циклу DID: створення, реєстрацію, розв'язання (отримання актуального опису) та деактивацію. Створення DID реалізується шляхом генерації криптографічної пари ключів в апаратному модулі безпеки та формування ідентифікатора через криптографічне виведення з публічного ключа. Це забезпечує перевірний зв'язок

між DID і ключем без потреби у централізованій реєстрації. Документ DID формується відповідно до специфікації W3C і містить публічний ключ, методи автентифікації, кінцеві точки сервісів та інформацію про контролер; для захисту від підміни документ підписується. Реєстрація DID у блокчейні виконується через транзакцію, яка фіксує документ DID разом із часовою міткою та криптографічним доказом існування, що дозволяє будь-якій стороні перевірити факт реєстрації в конкретний момент часу та унеможливлює «оформлення заднім числом».

Розв'язання DID здійснюється запитом до блокчейну з отриманням документа DID; при цьому локальний кеш із періодом актуальності (TTL) зменшує затримку для часто використовуваних DID. Актуалізація кешу підтримується сповіщеннями про події з реєстру у разі оновлення або деактивації DID, що забезпечує узгодженість кешованого та еталонного стану з допустимою затримкою. Деактивація DID реалізується транзакцією деактивації, яка позначає DID як неактивний, зберігаючи історичний запис у реєстрі для цілей аудиту.

Керування обліковими даними охоплює дві ролі: емітента (видача облікових даних іншим сторонам) та утримувача (зберігання облікових даних, виданих зовнішніми емітентами). Видача облікових даних передбачає формування структурованого документа згідно з W3C VC Data Model: атрибути суб'єкта, метадані емітента, строк чинності та криптографічний доказ у вигляді цифрового підпису. Використання децентралізованих ідентифікаторів для емітента та суб'єкта забезпечує контрольований зв'язок без залежності від централізованих реєстрів. За потреби делегування доступу до зашифрованих облікових даних можуть бути включені ключові матеріали: ключ шифрування даних підписується публічним ключем отримувача, що гарантує можливість розшифрування лише призначеним адресатом.

Зберігання облікових даних реалізується в локальній зашифрованій базі даних з індексами для швидкого пошуку за емітентом, суб'єктом або датою завершення строку дії [31]. Перевірка облікових даних під час їх пред'явлення виконується як багатокрокова процедура: перевірка структури на відповідність моделі, перевірка підпису емітента, перевірка строку дії, перевірка факту відкликання та доказ володіння ключем суб'єкта (підтвердження контролю приватного ключа). Перевірка відкликання здійснюється запитом до реєстру відкликань у блокчейні; для масштабування може застосовуватися пакетне відкликання на основі стиснених бітових векторів.

Керування станом підтримує поточне представлення SSDT, агрегуючи телеметрію з множинних IoT-пристроїв, а також історію станів для часових запитів. Оновлення стану запускається після надходження телеметричних повідомлень і включає перевірку підписів (автентичність джерела), перевірку номерів послідовності (виявлення пропусків/дублікатів), валідацію показників (виявлення аномалій) та атомарне оновлення стану з журналюванням для формування сліду аудиту. Історичні дані зберігаються у базі часових рядів із політиками зберігання, що балансують витрати на пам'ять і можливості аналітичних запитів. Для ефективного аналізу використовуються функції агрегації, які обчислюють статистики у часових «вікнах» без потреби переглядати всі сирі показники.

Політики доступу реалізуються механізмом оцінювання запитів відповідно до правил на основі атрибутів. Політики задаються як структуровані документи з умовами щодо атрибутів запитувача, ресурсу та контексту середовища. Оцінювання запиту включає зіставлення з релевантними політиками, комбінування кількох рішень за правилом пріоритетів (наприклад, «дозвіл має пріоритет» або «заборона має пріоритет») та формування рішення з поясненням для потреб аудиту. Використання декларативної мови опису політик замість імперативної реалізації спрощує аудит і створює передумови для формальної перевірки властивостей.

Генерація доказів з нульовим розголошенням дозволяє SSDT підтверджувати твердження про свої дані без розкриття первинних значень. Обчислення, яке підлягає доведенню, описується у вигляді арифметичної схеми (системи обмежень), а її підготовка виконується попередньо. Побудова доказу відбувається під час роботи системи: обчислюються значення «свідка» з приватних входів, завантажуються параметри доказування та виконується алгоритм побудови доказу; тривалість залежить від складності схеми. Згенеровані докази мають сталий розмір і можуть перевірятися за мілісекунди. Параметри доказування формуються в межах багатосторонньої процедури налаштування, безпека якої зберігається за умови чесності принаймні одного учасника.

Блокчейн-рівень. Блокчейн-рівень формує інфраструктуру довіри для реєстрації децентралізованих ідентифікаторів, керування статусами облікових даних та аудиту операцій доступу. Використання приватного блокчейну замість публічного забезпечує контроль членства з відомими ідентичностями учасників, вищу пропускну здатність завдяки оптимізованим алгоритмам узгодження стану та відповідність регуляторним вимогам через механізми контролю доступу. Модульна побудова платформи дозволяє замінювати реалізації механізму консенсусу, сервісів членства та середовища виконання смарт-контрактів, адаптуючи систему до різних сценаріїв розгортання. Мережева топологія

включає кілька організацій, які представляють зацікавлені сторони екосистеми. Кожна організація підтримує вузли-учасники, що зберігають копії реєстру та виконують смарт-контракти. Сервіс упорядкування транзакцій забезпечує узгодження їх порядку з використанням візантійсько-стійких або аварійностійких алгоритмів залежно від прийнятих припущень довіри. Керування членством і автентифікація учасників здійснюються через центри сертифікації, які видають сертифікати X.509. Механізм каналів дозволяє створювати ізольовані журнали для різних груп організацій із спільною видимістю лише в межах відповідного каналу.

Механізм консенсусу забезпечує узгодженість між розподіленими вузлами щодо порядку та коректності транзакцій. Типовий порядок обробки транзакції складається з кількох етапів: клієнт надсилає пропозицію транзакції вузлам, визначеним політикою підтвердження; ці вузли виконують смарт-контракт і формують набір читання/запису; клієнт збирає підтверджені відповіді та передає транзакцію сервісу упорядкування; сервіс формує блок та розповсюджує вузлам; після цього вузли перевіряють транзакції та фіксують їх у реєстрі. Політики підтвердження задають, які комбінації підписів вузлів потрібні для визнання транзакції чинною, що дозволяє налаштовувати моделі довіри – від підтвердження однією організацією до багатостороннього підтвердження. На етапі валідації перевіряється виконання політики підтвердження, відсутність конфліктів у наборах читання/запису та додаткові правила, визначені смарт-контрактом, перед остаточною фіксацією.

Смарт-контракти реалізують прикладну логіку для реєстру DID, керування статусами повноважень та контролю доступу. Смарт-контракт реєстру DID підтримує операції реєстрації, виконання, оновлення та деактивації DID із перевіркою унікальності та повноважень ініціатора. Смарт-контракт статусів повноважень забезпечує операції реєстрації, перевірки статусу та відкликання; для підвищення ефективності може застосовуватися пакетне відкликання на основі стиснених структур даних, що зменшує накладні витрати на зберігання. Смарт-контракт контролю доступу фіксує запити доступу та результати їх оцінювання, забезпечуючи виконання політик на рівні реєстру і генеруючи події для сповіщення зовнішніх систем.

Структура реєстру складається з двох взаємопов’язаних частин: журналу блоків (незмінної послідовності блоків) і поточного стану (ключ–значення), який відображає актуальну версію даних. Журнал блоків забезпечує виявлення підміни історії завдяки криптографічному хеш-зв’язуванню: кожен блок містить хеш попереднього, тому зміна будь-якого історичного блоку порушує цілісність усіх наступних. Поточний стан, своєю чергою, дозволяє виконувати запити до актуальних значень без необхідності відтворювати всю історію журналу. Оновлення стану відбувається на основі наборів читання/запису, сформованих під час виконання транзакцій, із автоматичним виявленням і розв’язанням конфліктів.

Механізм подій дозволяє смарт-контрактам генерувати події під час критичних операцій і сповіщати зовнішні застосунки. Клієнтські застосунки можуть підписуватися на визначені типи подій та отримувати сповіщення в режимі реального часу через сталі з’єднання. Для надійної доставки подій підтримується контрольна фіксація прогресу обробки (збереження номера останнього опрацьованого блоку), що дозволяє коректно відновлювати роботу після перезапуску клієнта. Додатково застосовується фільтрація подій, щоб клієнти отримували лише релевантні повідомлення замість повного потоку подій.

Вимоги до прототипу SSDT. Головною метою розробки прототипу SSDT є експериментальна верифікація ключових архітектурних рішень та протоколів, запропонованих у попередніх розділах. Прототип покликаний продемонструвати технічну можливість реалізації самосуверенної моделі управління цифровими двійниками в децентралізованому середовищі з використанням технологій блокчейну. Цей процес верифікації виходить за межі теоретичного аналізу і дозволяє оцінити практичну застосовність розроблених рішень у контрольованих умовах. Першочерговим завданням прототипу є демонстрація життєздатності архітектури SSDT. Прототип має підтвердити, що запропонована трирівнева архітектура, яка включає рівень ідентичності, рівень даних та рівень взаємодії, може функціонувати як цілісна система з дотриманням принципів самосуверенності. Особливу увагу приділено перевірці можливості збереження повного контролю власника над своїми даними та ідентифікаторами протягом усього життєвого циклу цифрового двійника, що становить фундаментальну основу концепції SSDT.

З функціональної точки зору прототип має реалізувати лише базові операції життєвого циклу SSDT. До них належать створення нового цифрового двійника з генерацією DID та криптографічних ключів, реєстрація документа DID у блокчейні, делегування прав доступу до даних двійника через видачу облікових даних, базова синхронізація стану між локальним сховищем та блокчейн-рівнем, а також відкликання раніше виданих повноважень. Водночас, прототип не включає складні сценарії міжмашинної взаємодії з множинними двійниками, повний набір операцій управління даними (версіонування, аудит, резервне копіювання), інтеграцію з зовнішніми IoT-пристроями та системами, а

також реалізацію всіх можливих типів облікових даних. Такий обмежений набір функціональності є достатнім для демонстрації ключових принципів архітектури, але не претендує на повноту промислового рішення. Модель даних, використана в прототипі, також є спрощеною порівняно з повноцінною системою. Для демонстраційних цілей прототип оперує базовими атрибутами ідентичності, такими як DID, публічні ключі та сервісні кінцеві точки, мінімальним набором метаданих, включаючи часові мітки, версії та статус, а також демонстраційними даними стану для ілюстрації механізмів доступу. Повна онтологія даних IoT-пристроїв та складні семантичні моделі, які можуть бути необхідними в реальних застосуваннях, залишаються поза межами прототипу. Це рішення дозволяє зосередитися на верифікації архітектурних принципів, не перевантажуючи прототип складністю предметної області.

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

У роботі розроблено концептуальну архітектуру самосуверенних цифрових двійників для IoT-екосистем, що забезпечує децентралізоване управління ідентичністю та даними пристроїв з гарантіями приватності та незмінності аудиту. Модель SSDT формалізовано як кортеж, що включає децентралізований ідентифікатор, множину атрибутів, функцію стану, політики доступу, криптографічні ключі та історію операцій, з визначенням інваріантів безпеки для гарантування коректності функціонування системи.

Розроблено трирівневу архітектуру, що складається з фізичного рівня, рівня цифрового двійника та блокчейн-рівня. Фізичний рівень реалізує автентичний збір даних на IoT-пристроях з криптографічним підписанням. Рівень цифрового двійника на обчислювальному шлюзі забезпечує управління децентралізованими ідентифікаторами через апаратні модулі безпеки, зберігання облікових даних, оцінювання політик доступу та генерацію доказів з нульовим розголошенням. Блокчейн-рівень гарантує незмінний аудит через приватний блокчейн з аварійностійким консенсусом та смарт-контрактами для реєстру DID, управління статусами облікових даних та журналювання операцій доступу.

Проаналізовано модель загроз SSDT-системи на основі методології STRIDE, адаптованої до розподілених IoT-систем. Аналіз зосереджено на ключових активах системи, зокрема ідентичностях, облікових даних, телеметрії від IoT-пристроїв і приватних ключах, а загрози згруповано за трьома основними напрямами: цілісність і автентичність обміну, загрози ідентичності, а також конфіденційність і контроль доступу. У межах цих напрямів ідентифіковано атаки типу Man-in-the-Middle, replay-атаки, підміну легітимних IoT-пристроїв або екземплярів SSDT, підробку облікових даних, обхід політик доступу, масовані спроби доступу та компрометацію приватних ключів. Для зменшення відповідних ризиків запропоновано комплекс контрзаходів, що включає взаємну автентифікацію на рівні mutual TLS, криптографічне підписання повідомлень на рівні застосунку, використання часових міток і номерів послідовності, обов’язкову реєстрацію DID у блокчейн-реєстрі, атестацію пристроїв, перевірку цифрових підписів емітентів і статусу відкликання облікових даних, багаторівневу верифікацію запитів, обмеження частоти запитів, а також захищене управління життєвим циклом криптографічних ключів із використанням апаратних модулів безпеки, ротації ключів і багатофакторної автентифікації для критичних операцій.

Результати дослідження демонструють можливість створення масштабованих та приватних систем управління IoT-пристроями з гарантіями самосуверенності. Запропонована архітектура забезпечує горизонтальну масштабованість через незалежне функціонування кожного SSDT, низьку латентність через локальну обробку на периферії та приватність за проєктуванням через вбудовані механізми селективного розкриття. Напрямки подальших досліджень включають формальну верифікацію властивостей безпеки через перевірку моделей, оптимізацію генерації доказів з нульовим розголошенням для пристроїв з обмеженими ресурсами, дослідження сумісності з існуючими IoT-платформами, аналіз регуляторних вимог та розробку механізмів управління для децентралізованого приватного блокчейну. Практична цінність результатів полягає в можливості застосування архітектури для IoT-систем у сферах з критичними вимогами приватності: індустріальний IoT, персональні моніторингові системи, розумні міста та інші. Архітектура SSDT забезпечує технологічну основу для реалізації принципів суверенності даних в IoT-екосистемах.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Tao, F., Zhang, M., & Nee, A. Y. C. (2019). *Digital twin driven smart manufacturing*. Academic Press. https://doi.org/10.1016/C2018-0-02206-9

2. Mühle, A., Grüner, A., Gayvoronskaya, T., & Meinel, C. (2018). A survey on essential components of a self-sovereign identity. *Computer Science Review*, *30*, 80-86. https://doi.org/10.48550/arXiv.1807.06346
3. 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* (pp. 1-15). https://doi.org/10.48550/arXiv.1801.10228
4. Grieves, M., & Vickers, J. (2017). Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In *Transdisciplinary perspectives on complex systems* (pp. 85-113). Springer. https://doi.org/10.1007/978-3-319-38756-7_4
5. Atzori, L., Iera, A., & Morabito, G. (2010). The internet of things: A survey. *Computer Networks*, *54*(15), 2787-2805. https://doi.org/10.1016/j.comnet.2010.05.010
6. Sporny, M., Longley, D., & Chadwick, D. (2022). *Decentralized identifiers (DIDs) v1.0*. W3C Recommendation. Retrieved February 17, 2026, from https://www.w3.org/TR/did-core/
7. Sporny, M., Longley, D., & Chadwick, D. (2022). *Verifiable credentials data model v1.1*. W3C Recommendation. Retrieved February 17, 2026, from https://www.w3.org/TR/vc-data-model/
8. Chaum, D. (1985). Security without identification: Transaction systems to make big brother obsolete. *Communications of the ACM*, *28*(10), 1030-1044.
9. Camenisch, J., & Lysyanskaya, A. (2001). An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In *International Conference on the Theory and Applications of Cryptographic Techniques*(pp. 93-118).Springer.https://doi.org/10.1007/3-540-44987-6_7
10. Vukolić, M. (2015). The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication. In *International Workshop on Open Problems in Network Security* (pp. 112-125). Springer. https://doi.org/10.1007/978-3-319-39028-4_9
11. Goldwasser, S., Micali, S., & Rackoff, C. (1989). The knowledge complexity of interactive proof systems. *SIAM Journal on Computing*, *18*(1), 186-208. https://doi.org/10.1145/22145.22178
12. Ben-Sasson, E., Chiesa, A., Tromer, E., et al. (2014). Succinct non-interactive zero knowledge for a von Neumann architecture. In *23rd USENIX Security Symposium* (pp. 781–796). https://dl.acm.org/doi/10.5555/2671225.2671275
13. Bünz, B., Bootle, J., Boneh, D., et al. (2018). Bulletproofs: Short proofs for confidential transactions and more. In *2018 IEEE Symposium on Security and Privacy* (pp. 315-334). IEEE. https://doi.org/10.1109/SP.2018.00020
14. Kosba, A., Miller, A., Shi, E., et al. (2016). Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. In *2016 IEEE Symposium on Security and Privacy* (pp. 839-858). IEEE. https://doi.org/10.1109/SP.2016.55
15. Shi, W., Cao, J., Zhang, Q., et al. (2016). Edge computing: Vision and challenges. *IEEE Internet of Things Journal*, *3*(5), 637-646. https://doi.org/10.1109/JIOT.2016.2579198
16. Bonomi, F., Milito, R., Zhu, J., & Addepalli, S. (2012). Fog computing and its role in the internet of things. In *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing* (pp. 13-16). https://doi.org/10.1145/2342509.2342513
17. Satyanarayanan, M., Bahl, P., Caceres, R., & Davies, N. (2009). The case for VM-based cloudlets in mobile computing. *IEEE Pervasive Computing*, *8*(4), 14-23. https://doi.org/10.1109/MPRV.2009.82
18. Yi, S., Li, C., & Li, Q. (2015). A survey of fog computing: Concepts, applications and issues. In *Proceedings of the 2015 Workshop on Mobile Big Data* (pp. 37-42). https://doi.org/10.1145/2757384.2757397
19. Bernstein, D. J., Duif, N., Lange, T., et al. (2012). High-speed high-security signatures. *Journal of Cryptographic Engineering*, *2*(2), 77-89. https://doi.org/10.1007/s13389-012-0027-1
20. Espressif Systems.(2026). *ESP32 technical reference manual*(Version 5.7). Retrieved February 17, 2026, https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf
21. Barricelli, B. R., Casiraghi, E., & Fogli, D. (2019). A survey on digital twin: Definitions, characteristics, applications, and design implications. *IEEE Access*, *7*, 167653-167671. https://doi.org/10.1109/ACCESS.2019.2953499
22. Hu, V. C., Ferraiolo, D., Kuhn, R., et al. (2013). *Guide to attribute based access control (ABAC) definition and considerations* (NIST Special Publication 800-162). National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-162
23. Merkle, R. C. (1987). A digital signature based on a conventional encryption function. In *Conference on the Theory and Application of Cryptographic Techniques* (pp. 369-378). Springer. https://doi.org/10.1007/3-540-48184-2_32
24. Shostack, A. (2014). *Threat modeling: Designing for security*. John Wiley & Sons.

25. Callegati, F., Cerroni, W., & Ramilli, M. (2009). Man-in-the-middle attack to the HTTPS protocol. *IEEE Security & Privacy*, *7*(1), 78-81. https://doi.org/10.1109/MSP.2009.12
26. Rescorla, E. (2018). *The transport layer security (TLS) protocol version 1.3* (RFC 8446). https://doi.org/10.17487/RFC8446
27. Needham, R. M., & Schroeder, M. D. (1978). Using encryption for authentication in large networks of computers. *Communications of the ACM, 21*(12), 993-999. https://doi.org/10.1145/359657.359659
28. Trusted Computing Group. (2019). *TPM 2.0 library specification*. Retrieved February 17, 2026, from https://trustedcomputinggroup.org/resource/tpm-library-specification/
29. Anderson, R. (2020). *Security engineering: A guide to building dependable distributed systems* (3rd ed.). John Wiley & Sons.
30. Bass, L., Clements, P., & Kazman, R. (2021). *Software architecture in practice* (4th ed.). Addison-Wesley Professional.
31. Banks, A., & Gupta, R. (2014). *MQTT version 3.1.1*. OASIS Standard. Retrieved February 17, 2026, from http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html

КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 1 (33), 2026
ISSN 2663 – 4023

Dmytro Ovsianko
Postgraduate Student of Information Technology Security Department
Lviv Polytechnic National University, Lviv, Ukraine
ORCID: 0009-0003-7758-0613
dmytro.o.ovsianko@lpnu.ua

Elena Nyemkova
Doctor of Sciences, Professor
Professor of the Information Technology Security Department
Lviv Polytechnic National University, Lviv, Ukraine
ORCID: 0000-0003-0690-2657
olena.a.niemkova@lpnu

CONCEPTUAL ARCHITECTURE AND FORMAL MODEL OF SELF-SOVEREIGN DIGITAL TWINS IN IoT ECOSYSTEMS

Abstract. The paper presents a conceptual architecture of self-sovereign digital twins (SSDT) for IoT ecosystems, which provides decentralized management of device identity and data without dependence on centralized providers. The proposed solution eliminates the main shortcomings of traditional IoT systems associated with centralized data storage, provider compromise risks, and lack of privacy guarantees. A three-tier architecture is developed: the physical layer ensures authentic data collection on IoT devices with cryptographic signing; the digital twin layer on the computing gateway implements decentralized identifier (DID) management, credential storage, access policy evaluation, and zero-disclosure evidence generation; the blockchain layer guarantees immutable audit through a private blockchain with fail-safe consensus, smart contracts for the DID registry, credential status management, and access operation logging. The SSDT model is formalized as a tuple that includes a decentralized identifier, a set of attributes, a state function, access policies, cryptographic keys, and transaction history, with clearly defined security invariants. A threat model based on the STRIDE methodology adapted to distributed IoT systems is analyzed. The analysis covers key assets (identities, credentials, telemetry, and private keys) and groups threats into areas: exchange integrity, identity threats, and confidentiality. Man-in-the-Middle attacks, replay attacks, device spoofing, credential forgery, access policy bypass, and key compromise are identified. A set of countermeasures is proposed that includes mutual TLS, cryptographic message signing, timestamps, mandatory DID registration in the blockchain, device attestation, revocation status checking, request frequency limitation, key rotation, and hardware security modules. Zero-disclosure proof mechanisms are used to ensure privacy. The results of the study confirm the possibility of creating scalable, private and self-sovereign IoT device management systems. The architecture provides horizontal scalability, low latency due to edge processing, and privacy by design. The practical value lies in the possibility of application in industrial IoT, personal monitoring systems, and smart cities. Further research directions include formal verification, optimization of ZKP for resource-dependent devices, and compatibility with existing IoT platforms.

Keywords: Digital Twins; Decentralized Identifiers; Internet of Things; Blockchain; Verified Credentials; Zero-Knowledge Proofs; Hardware Security Modules; Data Privacy; Distributed Systems

REFERENCES (TRANSLATED AND TRANSLITERATED)

1. Tao, F., Zhang, M., & Nee, A. Y. C. (2019). *Digital twin driven smart manufacturing*. Academic Press. https://doi.org/10.1016/C2018-0-02206-9
2. Mühle, A., Grüner, A., Gayvoronskaya, T., & Meinel, C. (2018). A survey on essential components of a self-sovereign identity. *Computer Science Review*, *30*, 80-86. https://doi.org/10.48550/arXiv.1807.06346
3. 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* (pp. 1-15). https://doi.org/10.48550/arXiv.1801.10228

4. Grieves, M., & Vickers, J. (2017). Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In *Transdisciplinary perspectives on complex systems* (pp. 85-113). Springer. https://doi.org/10.1007/978-3-319-38756-7_4
5. Atzori, L., Iera, A., & Morabito, G. (2010). The internet of things: A survey. *Computer Networks*, *54*(15), 2787-2805. https://doi.org/10.1016/j.comnet.2010.05.010
6. Sporny, M., Longley, D., & Chadwick, D. (2022). *Decentralized identifiers (DIDs) v1.0*. W3C Recommendation. Retrieved February 17, 2026, from https://www.w3.org/TR/did-core/
7. Sporny, M., Longley, D., & Chadwick, D. (2022). *Verifiable credentials data model v1.1*. W3C Recommendation. Retrieved February 17, 2026, from https://www.w3.org/TR/vc-data-model/
8. Chaum, D. (1985). Security without identification: Transaction systems to make big brother obsolete. *Communications of the ACM*, *28*(10), 1030-1044.
9. Camenisch, J., & Lysyanskaya, A. (2001). An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In *International Conference on the Theory and Applications of Cryptographic Techniques*(pp. 93-118).Springer.https://doi.org/10.1007/3-540-44987-6_7
10. Vukolić, M. (2015). The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication. In *International Workshop on Open Problems in Network Security* (pp. 112-125). Springer. https://doi.org/10.1007/978-3-319-39028-4_9
11. Goldwasser, S., Micali, S., & Rackoff, C. (1989). The knowledge complexity of interactive proof systems. *SIAM Journal on Computing*, *18*(1), 186-208. https://doi.org/10.1145/22145.22178
12. Ben-Sasson, E., Chiesa, A., Tromer, E., et al. (2014). Succinct non-interactive zero knowledge for a von Neumann architecture. In *23rd USENIX Security Symposium* (pp. 781–796). https://dl.acm.org/doi/10.5555/2671225.2671275
13. Bünz, B., Bootle, J., Boneh, D., et al. (2018). Bulletproofs: Short proofs for confidential transactions and more. In *2018 IEEE Symposium on Security and Privacy* (pp. 315-334). IEEE. https://doi.org/10.1109/SP.2018.00020
14. Kosba, A., Miller, A., Shi, E., et al. (2016). Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. In *2016 IEEE Symposium on Security and Privacy* (pp. 839-858). IEEE. https://doi.org/10.1109/SP.2016.55
15. Shi, W., Cao, J., Zhang, Q., et al. (2016). Edge computing: Vision and challenges. *IEEE Internet of Things Journal*, *3*(5), 637-646. https://doi.org/10.1109/JIOT.2016.2579198
16. Bonomi, F., Milito, R., Zhu, J., & Addepalli, S. (2012). Fog computing and its role in the internet of things. In *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing* (pp. 13-16). https://doi.org/10.1145/2342509.2342513
17. Satyanarayanan, M., Bahl, P., Caceres, R., & Davies, N. (2009). The case for VM-based cloudlets in mobile computing. *IEEE Pervasive Computing*, *8*(4), 14-23. https://doi.org/10.1109/MPRV.2009.82
18. Yi, S., Li, C., & Li, Q. (2015). A survey of fog computing: Concepts, applications and issues. In *Proceedings of the 2015 Workshop on Mobile Big Data* (pp. 37-42). https://doi.org/10.1145/2757384.2757397
19. Bernstein, D. J., Duif, N., Lange, T., et al. (2012). High-speed high-security signatures. *Journal of Cryptographic Engineering*, *2*(2), 77-89. https://doi.org/10.1007/s13389-012-0027-1
20. Espressif Systems.(2026). *ESP32 technical reference manual*(Version 5.7). Retrieved February 17, 2026, https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf
21. Barricelli, B. R., Casiraghi, E., & Fogli, D. (2019). A survey on digital twin: Definitions, characteristics, applications, and design implications. *IEEE Access*, *7*, 167653-167671. https://doi.org/10.1109/ACCESS.2019.2953499
22. Hu, V. C., Ferraiolo, D., Kuhn, R., et al. (2013). *Guide to attribute based access control (ABAC) definition and considerations* (NIST Special Publication 800-162). National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-162
23. Merkle, R. C. (1987). A digital signature based on a conventional encryption function. In *Conference on the Theory and Application of Cryptographic Techniques* (pp. 369-378). Springer. https://doi.org/10.1007/3-540-48184-2_32
24. Shostack, A. (2014). *Threat modeling: Designing for security*. John Wiley & Sons.
25. Callegati, F., Cerroni, W., & Ramilli, M. (2009). Man-in-the-middle attack to the HTTPS protocol. *IEEE Security & Privacy*, *7*(1), 78-81. https://doi.org/10.1109/MSP.2009.12
26. Rescorla, E. (2018). *The transport layer security (TLS) protocol version 1.3* (RFC 8446). https://doi.org/10.17487/RFC8446
27. Needham, R. M., & Schroeder, M. D. (1978). Using encryption for authentication in large networks of computers. *Communications of the ACM*, *21*(12), 993-999. https://doi.org/10.1145/359657.359659

28. Trusted Computing Group. (2019). *TPM 2.0 library specification*. Retrieved February 17, 2026, from https://trustedcomputinggroup.org/resource/tpm-library-specification/
29. Anderson, R. (2020). *Security engineering: A guide to building dependable distributed systems* (3rd ed.). John Wiley & Sons.
30. Bass, L., Clements, P., & Kazman, R. (2021). *Software architecture in practice* (4th ed.). Addison-Wesley Professional.
31. Banks, A., & Gupta, R. (2014). *MQTT version 3.1.1*. OASIS Standard. Retrieved February 17, 2026, from http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html

Отримано редакцією журналу / Received: 23.02.26
Прорецензовано / Revised: 02.03.26
Схвалено до друку / Accepted: 25.06.26