УДК 004.8

В. П. Коваленко; О. О. Ковалюк, канд. техн. наук, доц.

РОЗРОБКА ГІБРИДНОЇ АРХІТЕКТУРИ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ КЕРУВАННЯ КОНТРОЛЕМ ДОСТУПУ НА БАЗІ AI-АГЕНТІВ ТА ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ

Сьогодні глобальна економіка переживає цифрову трансформацію, що призводить до стрімкого збільшення обсягів даних та ускладнення IT-інфраструктури. У статті обґрунтовано доцільність застосування агентного штучного інтелекту та великих мовних моделей у процесах управління ідентифікацією та доступом (IAM) в умовах гібридної корпоративної інфраструктури. Показано, що традиційні моделі RBAC/ABAC у великих організаціях стикаються з «вибухом ролей», зростанням витрат і ризиком формального погодження, а також не здатні якісно опрацьовувати неструктуровані текстові обґрунтування та динамічний контекст. Метою дослідження є проєктування гібридної інтелектуальної системи керування контролем доступу, яка поєднує детерміновані політики з контекстним аналізом AI-агента.

Запропоновано трирівневу архітектуру: рівень взаємодії (ChatOps), рівень оркестрації (BPMN-оркестрація) і рівень прийняття рішень, реалізований як «sandwich»-підхід. Останній включає попередню перевірку «жорстких» обмежень засобами Policy-as-Code, семантичний аналіз ризиків LLM-агентом із застосуванням Chain-of-Thought, few-shot та структурованого JSON-виводу, а також поствалідацію відповіді. Формалізовано бізнес-процес у BPMN 2.0 із маршрутизацією запитів за рівнем ризику та підпроцесом Just-in-Time з автоматичним відкликанням доступу.

Експериментальне дослідження на тестовому стенді зі сценаріями рутинних запитів, підвищення привілеїв, екстреного доступу та промт-ін'єкцій показало, що система автоматизує обробку стандартних запитів із точністю близько 95 % і виявляє спроби маніпуляції. Новизна підходу полягає у поєднанні ймовірнісного LLM-міркування з детермінованим шаром політик, який виступає запобіжником проти галюцинацій і забезпечує пояснюваність та придатність рішень до аудиту. Практичний ефект – зниження навантаження на персонал і підвищення керованості та оперативності IAM-процесів.

Ключові слова: *інтелектуальна система керування доступом, штучний інтелект, великі мовні моделі, LLM-агенти, оркестрація бізнес-процесів.*

Вступ

Сьогодні глобальна економіка переживає цифрову трансформацію, що призводить до стрімкого збільшення обсягів даних та ускладнення IT-інфраструктури. Компанії все частіше відмовляються від монолітних систем на користь мікросервісів, хмарних рішень та SaaS-платформ. У таких умовах старі методи захисту, які покладалися на ізоляцію мережі, вже не працюють ефективно, адже чіткого периметра більше не існує. Тому ключовим елементом кіберзахисту стає управління ідентифікацією та доступом (Identity and Access Management – IAM), яке фактично перетворюється на новий периметр безпеки [1 – 4].

Відзначено, що класичні моделі контролю доступу (RBAC та ABAC) стають недостатньо ефективними при динамічному масштабуванні систем. Статистика показує, що у великих компаніях кількість ролей може сягати десятків тисяч, що робить систему непрозорою та складною в управлінні. Це явище отримало назву «вибух ролей» (role explosion) [5 – 9]. Через велике навантаження адміністратори часто погоджують запити формально (т. зв. «rubber stamping»), що призводить до надання зайвих прав та підвищує ризик витоку даних [10]. Звіти аналітиків підтверджують, що саме людські помилки при налаштуванні доступу часто стають причиною інцидентів.

Традиційні алгоритми не здатні ефективно обробляти неструктурований контекст (наприклад, текстові пояснення потреби в доступі) та адаптуватися до нових загроз у реальному часі. Тому виникає потреба у використанні інтелектуальних агентів, які можуть імітувати логіку експерта з безпеки. Розвиток генеративного штучного інтелекту (GenAI) та

великих мовних моделей (LLM) дозволяє створювати адаптивні системи, що поєднують розуміння природної мови із суворим дотриманням правил безпеки [11].

Практична цінність таких досліджень полягає у можливості автоматизувати до 90 % рутинних операцій з надання доступу, що дозволить фахівцям зосередитися на стратегічних завданнях. Впровадження інтелектуальної системи керування (ІСККД) допоможе реалізувати принципи нульової довіри (Zero Trust) на новому рівні: забезпечити безперервний аналіз ризиків для кожного запиту, а не лише статичну перевірку належності користувача до групи [2 – 4].

Таким чином, розробка гібридних архітектур на базі Agentic AI, які вирішують проблеми масштабованості RBAC та людського фактору, є вкрай актуальним завданням.

Аналіз останніх досліджень і публікацій

Аналіз останніх літературних джерел дозволяє виділити основні тенденції та невирішені проблеми у сфері управління доступом.

У роботах [12] детально розглядаються переваги та недоліки моделі Role-Based Access Control (RBAC). Автори погоджуються, що RBAC залишається стандартом де-факто для більшості підприємств завдяки своїй зрозумілості та простоті адміністрування на початкових етапах. Однак, як зазначається в [12], при зростанні організації виникає проблема "Role Explosion". Намагання врахувати всі можливі комбінації доступу призводить до створення вузькоспеціалізованих ролей, кількість яких може перевищувати кількість користувачів. Це ускладнює аудит і призводить до помилок. Залишається невирішеним питання динамічної адаптації рольової моделі без постійного ручного втручання адміністраторів.

Альтернативна модель на основі атрибутів (ABAC), описана в [13], пропонує більшу гнучкість завдяки використанню атрибутів користувача чи середовища. Проте її впровадження є складним і вимагає значних ресурсів для перевірки прав у реальному часі [12]. Крім того, дослідники вказують на проблему «пояснюваності»: у складних системах ABAC важко зрозуміти, чому доступ було надано чи заборонено, що ускладнює проходження аудитів.

Значна увага також приділена проблемі формального погодження під час сертифікації доступу. Дослідження [14] показують, що перевантажені менеджери часто затверджують запити автоматично. Це стається через «втому від прийняття рішень» та відсутність контексту. Наявні інструменти управління ідентифікацією (IGA) намагаються вирішити це простими методами, але вони не здатні аналізувати зміст бізнес-обґрунтувань.

Останні дослідження [11] фокусуються на застосуванні штучного інтелекту та великих мовних моделей (LLM) у кібербезпеці. У роботах [15] пропонується використовувати AI-агентів для автоматизації IGA. Проте більшість цих досліджень є теоретичними або розглядають AI лише як інструмент для аналізу логів. Питання використання LLM безпосередньо для прийняття рішень про доступ залишається маловивченим, особливо з огляду на проблему «галюцинацій» моделей [16].

Таким чином, аналіз джерел дозволяє виділити такі невирішені питання:

1. Відсутність гібридних моделей. Більшість рішень пропонують вибір: або жорсткі правила (RBAC/ABAC), або повний перехід на AI. Ефективні комбінації, де AI доповнює детерміновані політики, майже не розглядаються.

2. Проблема довіри та галюцинацій. Досі не розроблені надійні архітектурні шаблони, які дозволили б безпечно використовувати ймовірнісні моделі (LLM) у критичних процесах надання доступу та гарантували б відсутність помилкових дозволів.

3. Ігнорування неструктурованого контексту. Традиційні системи не вміють обробляти текстові пояснення, хоча саме вони часто є ключовим фактором для прийняття рішень людиною.

Ці питання залишаються відкритими через об'єктивну складність поєднання чіткої (детермінованої) логіки безпеки з ймовірнісною природою нейромереж, а також через

відносну новизну технологій Agentic AI.

Узагальнюючи, можна сформулювати головну проблему: наразі відсутня науково обґрунтована архітектура та методологія для створення інтелектуальних систем керування доступом. Така система мала б поєднувати масштабованість і контекстну обізнаність AI-агентів із гарантіями безпеки та прозорістю традиційних політик, усуваючи при цьому людський фактор у рутинних завданнях.

Мета та задачі дослідження

Враховуючи проаналізовану проблематику, метою дослідження є проєктування та реалізація гібридної архітектури інтелектуальної системи керування контролем доступу, яка інтегрує детерміновані механізми Policy-as-Code та LLM-агентів для контекстного аналізу запитів у корпоративних інформаційних системах.

Для досягнення поставленої мети необхідно вирішити такі задачі:

1. Спроєктувати гібридну архітектуру системи, яка поєднує чіткі механізми перевірки політик за допомогою підходу “Політика як код” (Policy-as-Code) із семантичним аналізом контексту на базі LLM. При цьому необхідно передбачити захист від «галюцинацій» та маніпуляцій даними.

2. Розробити алгоритми та методику інженерії промтів (prompt engineering) для LLM-агентів, використовуючи підходи Chain-of-Thought (CoT) та ReAct [17]. Це забезпечить структурованість, пояснюваність та можливість верифікації рішень про надання доступу.

3. Розробити формалізовану модель процесу надання та перегляду прав доступу. Вона має враховувати взаємодію між користувачами, власниками ресурсів та інтелектуальними агентами, а також визначати точки, де інтеграція ШІ буде найбільш ефективною.

4. Провести експериментальне дослідження системи на тестових сценаріях (рутинні запити, спроби підвищення привілеїв, екстрений доступ). **Мета** – оцінити рівень автоматизації та точність виявлення аномалій.

Комплексне виконання зазначених кроків забезпечить розробку не лише архітектури, але й верифікованої методології інтеграції LLM-агентів у критичні процеси управління доступом.

Об’єкт дослідження

Об’єктом дослідження є процес керування доступом до корпоративних інформаційних ресурсів у гібридній IT-інфраструктурі підприємства.

Зазначений об’єкт дослідження характеризується такими ознаками:

- запити на доступ містять текстове обґрунтування потреби;
- інфраструктура має централізований каталог ідентичностей та ресурсів;
- затримка звернення до хмарної LLM є допустимою для асинхронних процедур погодження доступу.

Методи дослідження

Для вирішення поставлених задач використовувався комплекс теоретичних та емпіричних методів:

1. Архітектурне проєктування. Для побудови гібридної архітектури застосовано принципи мікросервісної архітектури та підхід "Policy-as-Code". Як інструментарій реалізації обрано:
   - Camunda Platform 8 (на базі рушія Zeebe) – для оркестрації процесів та управління станом заявок. Вибір обґрунтований підтримкою BPMN та можливостями інтеграції через конектори.
   - LangChain та LangGraph – фреймворки для розробки агентної логіки. LangGraph дає можливість реалізувати циклічні графи виконання (cyclic graphs) для

підтримки патернів самокорекції (self-correction) агента.

- Oso Cloud (мова Polar) або Open Policy Agent (OPA) – для реалізації детермінованого шару політик. Це забезпечує виконання "жорстких" правил, які не залежать від ймовірності [18,19].

2. Інженерія промтів (Prompt Engineering). Для реалізації інтелектуального аналізу використовувалися сучасні методи взаємодії з LLM:
   - Chain-of-Thought (CoT) – спонукання моделі до покрокового міркування перед формуванням відповіді, що знижує ймовірність логічних помилок [20].
   - Few-Shot Prompting – надання моделі прикладів коректних та некоректних запитів у контексті промту для покращення точності класифікації ("in-context learning") [21].
   - Structured Output (схема JSON) – примусове формування відповіді у форматі JSON для забезпечення програмної обробки результатів [22].
3. Експериментальні методи. Для перевірки ефективності розробленої системи було розгорнуто тестовий стенд, що імітує інфраструктуру IT-компанії. У дослідженні використовувалися набори синтетичних даних, які відображають реальні профілі загроз та типові шаблони доступу. Оцінка роботи системи проводилася за ключовими метриками Accuracy, Precision та Recall, які використовувалися для оцінювання загальної правильності класифікації, точності виявлення ризикових запитів та повноти їх виявлення відповідно. Експерименти проводилися з використанням моделей сімейства GPT-4o (через API). Середовище виконання – Python 3.10, Docker-контейнери а також Camunda 8.

Проєктування гібридної архітектури ІСККД

Враховуючи обрані підходи, в роботі спроєктовано гібридну архітектуру ІСККД, наведену на рис. 1.

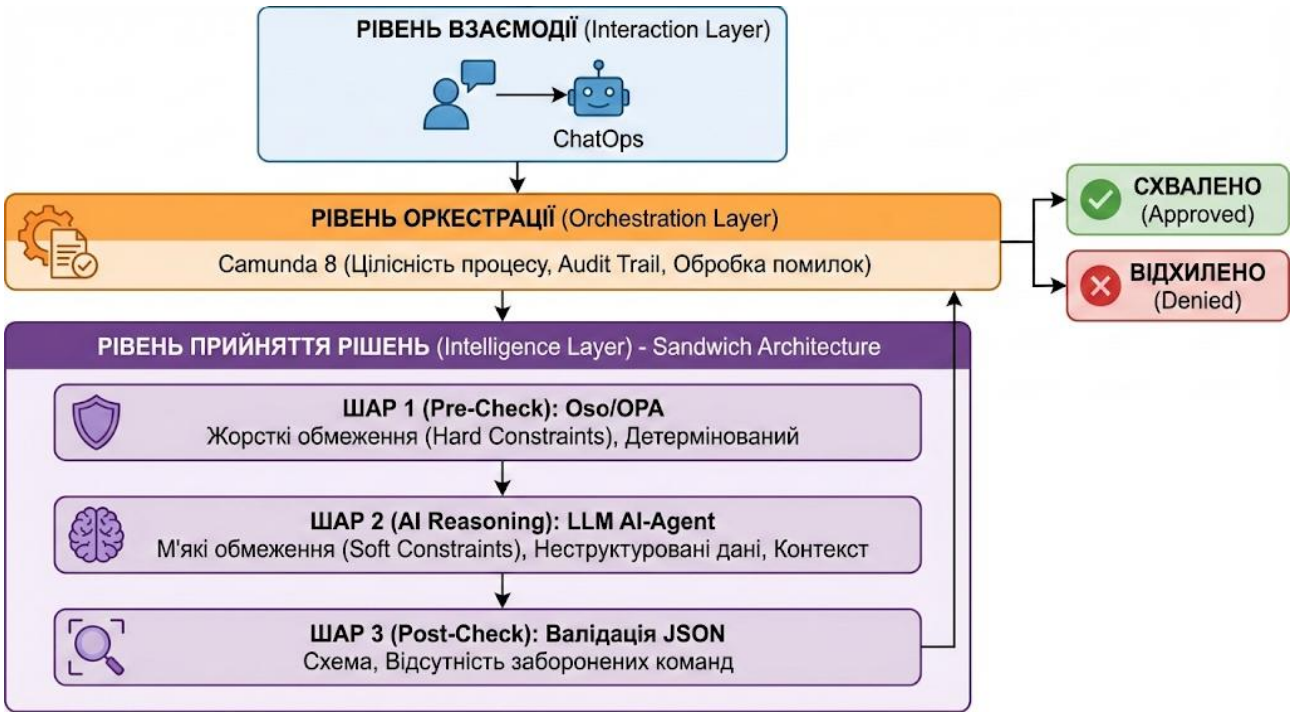

Рис. 1. Концептуальна схема гібридної архітектури ІСККД

Запропонована архітектура складається з трьох рівнів:

1. Рівень взаємодії (Interaction Layer): Забезпечує інтерфейс для користувачів. Реалізовано через чат-ботів (ChatOps), що дозволяє подавати запити природною

мовою.

2. Рівень оркестрації (Orchestration Layer): Базується на Camunda 8. Відповідає за цілісність процесу, логування всіх кроків (Audit Trail) та обробку помилок.
3. Рівень прийняття рішень (Intelligence Layer): Реалізує гібридний багатошаровий підхід (Sandwich Architecture):
   - Шар 1 (Pre-Check). Детермінований рушій політик (Oso/OPA). Перевіряє "жорсткі" обмеження (Hard Constraints), наприклад: "Контрактори ніколи не мають доступу до фінансових звітів". Це фільтрує очевидні порушення з нульовою ймовірністю помилки.
   - Шар 2 (AI Reasoning).AI-агент на базі LLM. Аналізує "м'які" обмеження та неструктуровані дані. Наприклад: "Чи відповідає текстове пояснення задачі ролі користувача?", "Чи є запит аномальним для цього часу доби?".
   - Шар 3 (Post-Check). Валідація вихідного JSON від LLM на відповідність схемі та відсутність заборонених команд.

Така архітектура забезпечує глибокий захист: навіть якщо LLM згенерує помилковий дозвіл (галюцинацію), детермінований шар політик заблокує його, якщо це суперечить фундаментальним правилам.

Розробка алгоритмів промт-інженерії

Для взаємодії з великою мовною моделлю розроблено алгоритм роботи агента, що базується на структурованому промті.

Системний промт передбачає такі елементи:

- **Role Definition:** "Ти – Sentinel, експерт з безпеки доступу (IAM Security Officer). Твоя мета – захист за принципом найменших привілеїв".
- **Input Context:** Визначення структури вхідних даних (JSON з атрибутами user, resource, context).
  - **Reasoning Steps (CoT):** Інструкція виконати аналіз крок за кроком:
    - Крок 1: Перевірка відповідності ролі та ресурсу.
    - Крок 2: Семантичний аналіз обґрунтування (чи є посилання на тікет? чи змістовне обґрунтування?).
    - Крок 3: Розрахунок рівня ризику (0-100) за формулою:

$$Risk = W_1 \cdot Sensitivity + W_2 \cdot Anomaly + W_3 \cdot (1 - JustificationQuality)$$

де

Sensitivity - оцінка чутливості ресурсу, до якого запитується доступ;

Anomaly- оцінка аномальності запиту порівняно з типовою поведінкою користувача або типовими сценаріями доступу;

JustificationQuality- оцінка якості та обґрунтованості текстового пояснення потреби в доступі;

W_1, W_2, W_3 - вагові коефіцієнти відповідних складових ризику, що визначають їх відносний внесок у підсумковий показник.

- **Guardrails:** Список заборонених дій (наприклад, ігнорування інструкцій користувачем).
- **Output Format:** Вимога повернути лише правильний JSON.

Вагові коефіцієнти W_1, W_2, W_3 нормуються в інтервалі [0;1], де 0 відповідає мінімальному ризику, а 1 – максимальному. Лінійна модель обрана як компроміс між інтерпретованістю, простотою аудиту та можливістю експертного налаштування вагових коефіцієнтів.

У межах дослідження вагові коефіцієнти обрано експертним шляхом з урахуванням пріоритетності факторів ризику в задачах контролю доступу: W_1=0,45, W_2=0,35, W_3=0,20, причому $W_1+W_2+W_3$=1. Найбільшу вагу надано показнику чутливості ресурсу як найбільш критичному фактору прийняття рішення, дещо меншу – показнику аномальності запиту, тоді

як якість обґрунтування розглядається як допоміжний контекстний критерій.

Алгоритм обробки запиту:

1. Агент отримує запит.
2. Використовує надані йому інструменти для отримання додаткових даних (наприклад звертається до агента політик, щоб зрозуміти жорсткі обмеження політики безпеки).
3. Генерує "думку" (reasoning) та фінальне рішення.
4. Якщо впевненість моделі низька (аналізуються логарифмічні ймовірності, logprobs) або структура JSON порушена, запускається цикл самокорекції, де агент отримує повідомлення про помилку і намагається згенерувати відповідь знову.

Таким чином, розроблена методика промт-інженерії, що використовує Chain-of-Thought та структурований вихід (JSON), забезпечує високу передбачуваність роботи LLM-агента та його інтеграцію в детермінований процес оркестрації.

Експериментальне дослідження

Для перевірки запропонованих рішень було проведено експериментальне дослідження. Традиційний процес, що включає ручне створення тікету, перевірку менеджером та виконання адміністратором, було трансформовано у напівавтономний потік.

AI-агент, як актор у процесі, не просто виконує команди, а виступає його активним учасником. Ключові варіанти використання для AI-агента включають аналіз контексту та ризиків, а також генерацію рекомендації. Ці дії розширюють базовий сценарій обробки запиту.

Для деталізації логіки виконання розроблено модель у нотації BPMN 2.0.

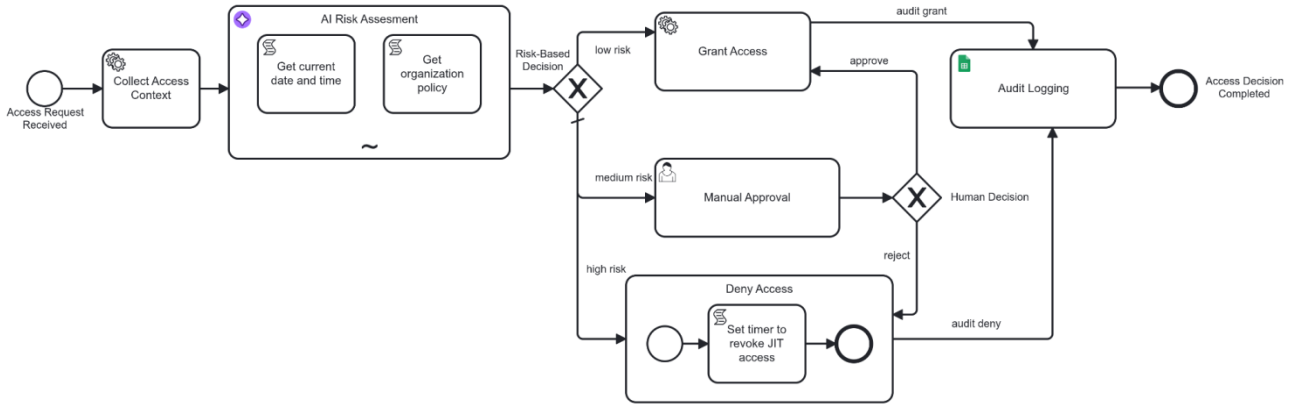

Рис. 2. BPMN-діаграма процесу обробки запиту

Запропоновано такі основні елементи процесу:

1. Start Event. Отримання запиту через API (наприклад, з Slack-бота або порталу Jira).
2. Service Task (Enrichment). Автоматичний збір контексту. Система звертається до Identity Provider (IdP) для отримання атрибутів користувача (департамент, дата найму, історія порушень) та до CMDB для отримання метаданих ресурсу (рівень критичності, власник).
3. Service Task (AI Analysis). Це ключова точка інтеграції. Збагачений контекст передається AI-агенту.
4. Exclusive Gateway (Decision Routing). На основі вердикту агента потік розгалужується на три шляхи:
   - Path A (Low Risk) -автоматичне затвердження та надання доступу.
   - Path B (Medium Risk/Uncertainty) – створення User Task для ручного погодження власником ресурсу. При цьому користувач отримує згенерований ШІ звіт з поясненням ризиків.
   - Path C (High Risk/Policy Violation) – автоматична відмова з повідомленням причини.

5. Sub-process (JIT Access). У разі надання доступу запускається таймер, після закінчення якого доступ автоматично відкликається.

Така модель дозволяє реалізувати принцип надання доступу “точно в час” (Just-in-Time) та мінімізувати час очікування для безпечних запитів.

Експериментальне дослідження було проведено з використанням Camunda Platform 8 в якості Рівня оркестрації. Сервіс Camunda використовувався для реалізації та управління формалізованою BPMN-моделлю процесу доступу. Сюди входить управління послідовними етапами роботи: збором контексту, аналіз даних AI-агентом, прийняття рішення та автоматичне надання доступу.

Перевірка проводилася на тестовому стенді, що моделював типову корпоративну IT-інфраструктуру з централізованим керуванням ідентичностями, каталогом ресурсів і механізмом погодження доступу. Для оцінювання роботи системи було сформовано 200 синтетичних запитів, розподілених на чотири групи:

1. 80 запитів на стандартний доступ;
2. 50 запитів на підвищення привілеїв;
3. 30 запитів на екстрений доступ;
4. 40 атакувальних або маніпулятивних сценаріїв, зокрема промт-ін’єкцій.

Для кожного сценарію попередньо задавався еталонний клас рішення: «дозволити», «передати на ручне погодження» або «відхилити». Оцінювання проводилося шляхом порівняння рішень системи з передньо визначеним еталонним рішенням.

Як основні метрики використано:

- точність класифікації рішень (Accuracy);
- точність виявлення ризикових запитів (Precision), яка характеризує частку запитів, віднесених системою до ризикових, що справді є ризиковими;
- повноту виявлення ризикових запитів (Recall), яка характеризує частку фактично ризикових і маніпулятивних сценаріїв, коректно виявлених системою;
- частку автоматично оброблених стандартних запитів;
- частку коректно виявлених промт-ін’єкцій.

Проведене експериментальне дослідження забезпечує необхідні дані для їх подальшого аналізу та інтерпретації.

Аналіз результатів

Отримані результати підтверджують ефективність запропонованого гібридного підходу.

За результатами експерименту встановлено, що:

- точність класифікації стандартних запитів (Accuracy) становила близько 95 %;
- частка автоматично оброблених низькоризикових запитів склала 82 %;
- повнота виявлення ризикових і маніпулятивних сценаріїв (Recall) становила 91 %;
- точність виявлення запитів категорії високого ризику (Precision) становила 93 %;
- усі тестові сценарії промт-ін’єкцій були або заблоковані детермінованим шаром політик, або передані на ручний розгляд.

Запропонована гібридна архітектура дозволяє автоматично обробляти стандартні запитів на доступ. При цьому точність обробки залишається високою, складає 95 %. Система також успішно виявила спроби промт-ін’єкції. Це підтвердило надійність двошарового підходу, який поєднує детерміновані політики та аналіз AI-агентом. Головний практичний ефект досягається завдяки автоматизації рутинних завдань, що значно знижує навантаження на персонал, а також усуває проблему формального погодження запитів.

Висока точність системи зумовлена тим, що LLM здатна враховувати нюанси контексту, які недоступні для звичайних статичних правил. Наприклад, у сценарії підвищення привілеїв агент зміг відрізнити легітимний запит (виправлення критичної помилки в продукті, підтверджене номером тікета через пошук за допомогою наданих агенту інструментів) від спроби несанкціонованого розширення прав. Це стало можливим саме завдяки семантичному

аналізу тексту запиту.

На відміну від класичних IGA-систем, які базуються на періодичних перевірках та статичних ролях, запропонована система працює в реальному часі. Порівняно з рішеннями на базі машинного навчання [23], агент не просто сигналізує про аномалію, а приймає рішення і, що важливо, пояснює його. Наявність такого пояснення робить систему прозорою для аудиту та вирішує проблему «чорної скриньки» [24].

Разом з тим, запропонований підхід має ряд обмежень, які повинні враховуватися. Одним з таких обмежень є залежність від якості вхідних даних (описів ролей, запитів). Також слід враховувати вартість та затримку при використанні комерційних LLM через API, що може бути критичним для високошвидкісних систем. Також потрібно мати на увазі ймовірнісну природу великих мовних моделей. Існує ризик недетермінованості – коли модель на ті самі вхідні дані може дати різну відповідь. Щоб мінімізувати це явище, в дослідженні використано нульову температуру генерації (temperature=0) та додатковий шар жорстких правил (Policy-as-Code).

Висновки

У роботі спроєктовано гібридну архітектуру інтелектуальної системи керування контролем доступу, яка поєднує детермінований шар Policy-as-Code та LLM-агента для аналізу неструктурованого контексту запиту. Така комбінація дозволила вирішити проблему "подвійної сліпоти": система бачить і жорсткі правила, і м'який контекст.

Формалізовано BPMN-модель процесу надання доступу з маршрутизацією заявок за інтегральним показником ризику та механізмом Just-in-Time для автоматичного відкликання тимчасових прав. Це дозволило структурно відокремити низькоризикові запити, що можуть оброблятися автоматично, від запитів, які потребують експертного погодження.

Експериментальне дослідження показало, що запропонована система забезпечує точність обробки стандартних запитів на рівні близько 95 %, автоматизує обробку більшості низькоризикових заявок та виявляє маніпулятивні сценарії, включаючи промт-ін’єкції. Практичний ефект полягає у зменшенні навантаження на адміністраторів, скороченні частки формального погодження запитів та підвищенні керованості процесів IAM.

Отримані результати підтверджують доцільність використання гібридного підходу для систем контролю доступу в корпоративних інформаційних середовищах. Перспективою подальших досліджень є розширення набору тестових сценаріїв та оцінювання системи на реальних корпоративних даних.

СПИСОК ЛІТЕРАТУРИ

1. Identity and Access Management Architecture in the SILVANUS Project / P. Rajba et al. ARES *2024: The 19th International Conference on Availability, Reliability and Security (Vienna, Austria)*. New York, NY, USA, 2024. P. 1–9. URL:https://doi.org/10.1145/3664476.3670935 (дата звернення: 15.02.2026).

2. Bertino E. Zero Trust Architecture: Does It Help? *IEEE Security & Privacy*. 2021. Т. 19, № 5. P. 95–96. URL: https://doi.org/10.1109/msec.2021.3091195 (дата звернення: 15.02.2026).

3. A Survey on Zero Trust Architecture: Challenges and Future Trends / Y. He et al. *Wireless Communications and Mobile Computing*. 2022. Т. 2022. P. 1–13. URL: https://doi.org/10.1155/2022/6476274 (дата звернення: 15.02.2026).

4. Katsis C., Bertino E. The Zero-trust Paradigm: Concepts, Architectures and Applications. *Foundations and Trends® in Privacy and Security*. 2025. Т. 8, № 2. P. 122–253. URL: https://doi.org/10.1561/3300000046 (дата звернення: 15.02.2026).

5. Penelova M. Access Control Models. *Cybernetics and Information Technologies*. 2021. Т. 21, № 4. P. 77–104. URL: https://doi.org/10.2478/cait-2021-0044 (дата звернення: 15.02.2026).

6. A systematic literature review for authorization and access control: definitions, strategies and models / A. K. Y. S. Mohamed et al. *International Journal of Web Information Systems*. 2022. Т. 18, № 2-3. P. 156–180. URL: https://doi.org/10.1108/IJWIS-04-2022-0077 (дата звернення: 15.02.2026).

7. Jia J., Guan J., Wang L. Role Mining: Survey and Suggestion on Role Mining in Access Control. *Communications in Computer and Information Science. Singapore*, 2020. P. 34–50. URL: https://doi.org/10.1007/978-981-15-9609-4_4 (дата звернення: 15.02.2026).

8. Guclu M., Bakir C., Hakkoymaz V. A New Scalable and Expandable Access Control Model for Distributed

Database Systems in Data Security. *Scientific Programming*. 2020. Т. 2020. P. 1–10. URL: https://doi.org/10.1155/2020/8875069 (дата звернення: 15.02.2026).

9. An Efficient Attribute-Based Access Control (ABAC) Policy Retrieval Method Based on Attribute and Value Levels in Multimedia Networks / M. Liu et al. *Sensors*. 2020. Т. 20, №6. P. 1741. URL: https://doi.org/10.3390/s20061741 (дата звернення: 15.02.2026).

10. Wagner B. Liable, but Not in Control? Ensuring Meaningful Human Agency in Automated Decision-Making Systems. *Policy & Internet*. 2019. Т. 11, №1. P. 104–122. URL: https://doi.org/10.1002/poi3.198 (дата звернення: 15.02.2026).

11. Generative AI and LLMs for Critical Infrastructure Protection: Evaluation Benchmarks, Agentic AI, Challenges, and Opportunities / Y. Yigit et al. *Sensors*. 2025. Т. 25, №6. P. 1666. URL: https://doi.org/10.3390/s25061666 (дата звернення: 15.02.2026).

12. Ahsan M. S., Pathan A.-S. K. A Comprehensive Survey on the Requirements, Applications, and Future Challenges for Access Control Models in IoT: The State of the Art. IoT. *MDPI*. 2025. Т. 6, №1. P. 9. URL: https://doi.org/10.3390/iot6010009 (дата звернення: 15.02.2026).

13. ABAC Lab: An Interactive Platform for Attribute-based Access Control Policy Analysis, Tools, and Datasets / T. Bui et al. *SACMAT '25*. New York, NY, USA, 2025. P. 111–116. URL: https://doi.org/10.1145/3734436.3734441 (дата звернення: 15.02.2026).

14. Gupta V. Optimizing Access Recertifications. *IDPro Body of Knowledge*. 2025. Т. 1, № 16. URL: https://doi.org/10.55621/idpro.119 (дата звернення: 15.02.2026).

15. Bono J., Cheng B., Lozano J. Randomized Controlled Trials for Conditional Access Optimization Agent. *ArXiv.2511.138652025*. URL: https://doi.org/10.48550/arXiv.2511.13865 (дата звернення: 15.02.2026).

16. Hiriyanna S., Zhao W. Multi-Layered Framework for LLM Hallucination Mitigation in High-Stakes Applications: A Tutorial. Computers. *MDPI*. 2025. Т. 14, № 8. URL: https://doi.org/10.3390/computers14080332 (дата звернення: 15.02.2026).

17. ReAct: Synergizing Reasoning and Acting in Language Models / S. Yao et al. 2023. *ArXiv.2210.03629*. URL: https://doi.org/10.48550/arXiv.2210.03629 (дата звернення: 15.02.2026).

18. Yang B. Enforcement of Separation of Duty Constraints in Attribute-Based Access Control. *Computers & Security*. 2023. Ст. 103294. URL: https://doi.org/10.1016/j.cose.2023.103294 (дата звернення: 15.02.2026).

19. Cedar: A New Language for Expressive, Fast, Safe, and Analyzable Authorization / J. W. Cutler et al. *Proceedings of the ACM on Programming Languages*. 2024. Т. 8, OOPSLA1. P. 670–697. URL: https://doi.org/10.1145/3649835 (дата звернення: 15.02.2026).

20. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models / J. Wei et al. *ArXiv.2201.11903*. 2022. URL:https://doi.org/10.48550/arXiv.2201.11903 (дата звернення: 15.02.2026).

21. Language Models are Few-Shot Learners / T. B. Brown et al. *ArXiv.2005.14165*. 2020. URL:https://doi.org/10.48550/arXiv.2005.14165 (дата звернення: 15.02.2026).

22. Are LLMs good at structured outputs? A benchmark for evaluating structured output capabilities in LLMs / Y. Liu et al. *Information Processing & Management*. 2024. Т. 61, №5. P. 103809. URL: https://doi.org/10.1016/j.ipm.2024.103809 (дата звернення: 15.02.2026).

23. Machine Learning-Enhanced Attribute-Based Authentication for Secure IoT Access Control / J. Saleem et al. *Sensors*. 2025. Т. 25, № 9. P. 2779. URL: https://doi.org/10.3390/s25092779 (дата звернення: 15.02.2026).

24. Pakina A. K., Pujari T., Goel A. Explainable AI and Governance: Enhancing Transparency and Policy Frameworks through Retrieval-Augmented Generation (RAG). *IOSR Journal of Computer Engineering*. 2023. Т. 25. P. 65–79. URL: https://doi.org/10.9790/0661-2506016579 (дата звернення: 15.02.2026).

Стаття надійшла до редакції 04.03.2026.
Стаття пройшла рецензування 16.03.2026.
Стаття опублікована 31.03.2026.

Коваленко Володимир Петрович – аспірант кафедри комп'ютерних систем управління, ORCID: 0009-0000-2576-7337, e-mail: digit.vova@gmail.com.

Ковалюк Олег Олександрович – канд. техн. наук, доцент кафедри комп'ютерних систем управління, ORCID: 0000-0002-0718-010X.

Вінницький національний технічний університет.