КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 4 (32), 2026
ISSN 2663 – 4023
DOI 10.28925/2663-4023.2026.32.1185
УДК 004.056.5

Абібулаєв Азіз Русланович
аспірант кафедри захисту інформації
Національний Університет «Львівська Політехніка», Львів, Україна
ORCID: 0009-0004-2875-5154
aziz.r.abibulaiev@lpnu.ua

Піскозуб Андріян Збігнєвич
к.т.н., доцент кафедри захисту інформації
Національний Університет «Львівська Політехніка», Львів, Україна
ORCID: 0000-0002-3582-2835
andriian.z.piskozub@lpnu.ua

Атамуратов Едем Айдерович
аспірант кафедри прикладної математики
Національний Університет «Львівська Політехніка», Львів, Україна
ORCID: 0009-0006-2917-9448
edem.atamuratov.asp.2025@lpnu.ua

КРИТЕРІЇ РИЗИКУ ТА АЛГОРИТМИ ML ДЛЯ РОЗПІЗНАВАННЯ НЕБЕЗПЕКИ У ХМАРНОМУ СЕРЕДОВИЩІ

Анотація. У статті запропоновано та експериментально перевірено ML-орієнтований конвеєр виявлення небезпечних подій в API/HTTP-трафіку хмарних сервісів. Підхід поєднує два числові канали: (i) керований канал на структурованих ознаках події для стабільного розділення на нормальні або атаки, та (ii) додатковий канал нетипової поведінки, який підсилює реакцію на рідкісні або нові сценарії, слабко представлені у розмічених даних. Ключовою методологічною ідеєю є уніфікація різнорідних виходів моделей у спільну ймовірнісну шкалу ризику за допомогою калібрування, температурного масштабування та корекції на апріорну частку атак, що забезпечує порівнюваність оцінок між моделями різної природи. Для керованого контролю хибно-позитивних спрацювань застосовано вибір порогу за FPR-бюджетом, а для стабілізації групових оцінок використано ансамблювання, зокрема усереднення в адитивній шкалі відношення шансів. Після об'єднання сигналів каналів фінальне рішення стабілізується політиками керованості (зокрема гістерезисом), щоб уникати частих перемикань стану спрацювання в потоковому режимі. Окремо, якість моделей верифіковано як табличними метриками, так і візуально через аналіз розподілів оцінок ризику для нормальних подій та атак, що дозволяє інтерпретувати зону перекриття та вплив порогу на хибні сповіщення про небезпеку. Експерименти на тестовій вибірці показали високу якість керованого каналу: для основного ML-ансамблю отримано ROC-AUC=0.9843, PR-AUC=0.9511 та F1=0.8400 при FPR близько 0.051, тоді як базова лінійна модель має суттєво нижчі значення F1. Додатковий канал нетипової поведінки формує практично корисний сигнал, який доповнює керований канал при контрольованому рівні хибних сповіщень про небезпеку. Запропонована постановка придатна до масштабування на інші типи API та профілі навантаження, оскільки розділяє політики (пороги, ваги, параметри калібрування) від основного потоку обробки подій. Отримані результати підтверджують придатність підходу для інтеграції в інфраструктуру моніторингу та реагування у хмарному середовищі з керованими політиками порогів і оновленням моделей.

Ключові слова: машинне навчання; оцінювання ризику; захист API/HTTP; визначення аномалій; OWASP top-10; ризик-орієнтовані сповіщення; хмарні мікросервіси.

ВСТУП

Сучасні хмарні середовища, на нашу думку, найточніше описуються трьома рисами: високою динамічністю, розподіленістю сервісів і домінуванням API-взаємодій (Application Programming Interface). Мікросервісні архітектури, автоматизоване масштабування, безперервні процеси інтеграції та доставки коду CI/CD (Continuous Integration, Continuous Delivery), а також наявність публічних інтерфейсів доступу формують ситуацію, за якої безпека дедалі менше залежить від статичних периметрів і дедалі більше – від здатності системи оперативно інтерпретувати телеметрію та трафік у контексті ризику. У таких умовах традиційні сигнатурні підходи (правила, шаблони, статичні індикатори компрометації) природно схильні до деградації: вони чутливі до змін у поведінці сервісів, недостатньо стійкі до обфускації та, що не менш важливо, слабко відображають потенційні наслідки події для бізнесу.

Водночас саме API-трафік є ключовим каналом реалізації значної частини атак на хмарні системи, зокрема ін’єкцій, несанкціонованого доступу до ресурсів, зловживання бізнес-логікою, BruteForce та Credential Stuffing, а також ексфільтрації даних через легітимні механізми. Для практичної експлуатації системи безпеки, як показує досвід експлуатаційних команд, недостатньо лише фіксувати факт аномалії або підозрілої поведінки. Критичною стає задача розпізнавання небезпеки: визначення того, наскільки подія є ризиковою саме в поточному середовищі, який її потенційний вплив, і які керовані дії є доцільними з позиції операційного центру безпеки SOC (Security Operations Center) та практик DevSecOps.

Це, у свою чергу, зумовлює потребу в поєднанні підходів машинного навчання, які доповнюють один одного за типом сигналу. З одного боку, потрібні керовані моделі на структурованих ознаках, здатні стабільно розділяти нормальні події від атак на змішаному трафіку. З іншого боку, необхідні моделі нетипової поведінки, що навчаються на нормальних подіях і формують сигнал відхилення від норми, який є корисним для нових або рідкісних сценаріїв, слабко представлених у розмічених даних. Водночас ефективність ML-підходів (Machine Learning) у хмарі визначається не лише точністю окремих моделей, а й наявністю чітких критеріїв ризику, формалізованої математичної постановки конвеєра та керованих механізмів перетворення різнорідних оцінок у узгоджене рішення.

У цій статті ми зосереджуємося на побудові ML-конвеєра оцінювання ризику подій API/HTTP (Hypertext Transfer Protocol), який об’єднує керований числовий канал та канал нетипової поведінки, приводить їхні виходи до єдиної ймовірнісної шкали ризику $p(x)\in[0,1]$ та забезпечує керований контроль хибно-позитивних спрацювань через політику вибору робочого порогу.

Постановка проблеми. На думку авторів, центральна складність розпізнавання небезпеки у хмарному середовищі полягає в тому, що одна й та сама спостережувана подія може мати принципово різний рівень ризику залежно від контексту виконання та характеристик активу. Показовим є приклад із параметром запиту. Підозрілий рядок у тестовому ізольованому середовищі нерідко є результатом легітимного тестування або перевірки функціоналу. Водночас аналогічний рядок у промисловому сервісі, який обробляє персональні дані та доступний з публічної мережі, може бути індикатором потенційного інциденту з високим очікуваним впливом. Отже, небезпека не зводиться до бінарного рішення “атака/не атака”. Вона формується як результат поєднання (i) сигналів аномальності та нетиповості, (ii) оцінок керованих моделей на структурованих

ознаках події та (iii) контекстних факторів ризику, які визначають критичність активу і умови його експлуатації.

У прикладному вимірі задачу додатково ускладнюють особливості хмарного API-трафіку, які безпосередньо впливають на якість та операційну придатність виявлення атак:

- поведінкові профілі сервісів змінюються через випуски нових версій ПЗ (програмного забезпечення), еластичне масштабування, сезонність навантаження та еволюцію клієнтських шаблонів використання, через що є нестабільними;
- реальні атаки становлять малу частку загального потоку подій, через що зростає кількість хибно-позитивних спрацювань, перевантажуються процеси обробки сповіщень і підвищуються витрати команд безпеки на первинний аналіз;
- ознаки подій є неоднорідними, тому для однієї узгодженої оцінки ризику потрібно поєднувати структурні та числові характеристики запиту, параметри доступу й часові шаблони, а також показники складності та варіативності вхідних даних (ентропійні та частотні характеристики);
- рішення має ухвалюватися в реальному часі та в межах жорстких вимог до затримки, водночас система повинна масштабуватися під пікові навантаження і зберігати стабільність політики реагування;
- для практичного використання потрібна пояснюваність, оскільки необхідно надавати причини спрацювання (ключові ознаки та фактори ризику) та рекомендаційні дії, які можна застосувати в роботі операційного центру безпеки.

Таким чином, у формальному вигляді проблема зводиться до побудови системи, яка для кожної події API-трафіку формує ризик-орієнтоване рішення на основі узгоджених оцінок ML-моделей, коригує підсумковий рівень ризику з урахуванням контексту активу й середовища, забезпечує стабільність рішень у потоці (зокрема за рахунок політик стабілізації та гістерезису) та надає пояснення, достатні для операційних процесів. Водночас наявні підходи зазвичай розв’язують лише частину задачі, наприклад виявлення аномалій або окрему класифікацію подій, але не забезпечують уніфікованої шкали ризику та керованої політики вибору робочої точки з контролем хибних сповіщень про небезпеку у реальному потоці.

Огляд літературних джерел. У наукових працях, присвячених безпеці хмарних середовищ, на нашу думку, досить чітко простежується спільна позиція. Зростання динамічності сервісів, обсягів телеметрії та варіативності атак поступово знижує ефективність суто сигнатурних механізмів. Саме тому підходи на основі машинного навчання дедалі частіше розглядаються як інструмент підвищення адаптивності виявлення і підтримки проактивного захисту. Ми вважаємо переконливим аргумент, який повторюється в багатьох роботах, що ML-підходи здатні фіксувати нетипові шаблони поведінки та зменшувати затримку між появою загрози й її ідентифікацією. Водночас ми оцінюємо як критично важливі зауваження авторів щодо обмежень, які визначають практичну придатність. Це якість даних, дрейф розподілів, ризик хибно-позитивних спрацювань і потреба в інтерпретованості для використання в операційних процесах SOC та DevSecOps [1], [2], [3].

Систематизація критеріїв і методів для хмарної безпеки значною мірою спирається на оглядові та таксономічні дослідження. На нашу думку, їхня цінність полягає в тому, що вони задають рамку для коректного порівняння алгоритмів і для постановки експлуатаційних вимог. У комплексному огляді з таксономією загроз, класів захисту та технік ML і DL (Deep Learning) виокремлено виклики, які ми також вважаємо визначальними для практичних систем розпізнавання небезпеки. Це

незбалансованість класів, підвищена чутливість до хибно-позитивних спрацювань, вимоги до масштабованості, а також відкрита проблема пояснюваності як умови довіри до автоматизованих рішень [4]. Аналіз тенденцій DL і ML у безпеці хмарних обчислень, на нашу думку, правильно підсвічує зсув фокуса в бік виявлення аномалій, поведінкової аналітики, безперервного моніторингу та інтеграції з процесами реагування.

Водночас ми погоджуємося з тезою, що прозорість рішень і керованість порогів дедалі частіше стають бар'єром впровадження в реальних середовищах [5]. Огляд класів алгоритмів машинного навчання в контексті хмарної безпеки додатково підтримує висновок, який ми також поділяємо. Вибір моделі має визначатися не лише точністю, а й профілем помилок, обчислювальною складністю та стійкістю до змін середовища, оскільки саме ці властивості прямо впливають на операційну придатність виявлення атак [6].

Окремий напрям у літературі, який ми вважаємо доречним для нашої постановки задачі, стосується застосування глибинного навчання безпосередньо до HTTP-запитів як до послідовностей символів або токенів. У таких підходах модель навчається розрізняти нормальні події та атаки на основі представлення запиту, а потреба в ручному проєктуванні частини ознак зменшується. На нашу думку, сильна сторона цього класу рішень полягає у здатності підхоплювати варіативні шаблони вхідних даних. Водночас ми оцінюємо як практично важливий момент питання керованості хибно-позитивних спрацювань, оскільки висока чутливість без контрольованої робочої точки може перевантажувати первинний аналіз. Прикладом такого підходу є архітектура DL-WAD (Deep Learning solution for Web Attack Detection), у якій передбачено нормалізацію запиту, кодування вхідних даних та класифікацію за допомогою глибинної моделі [7].

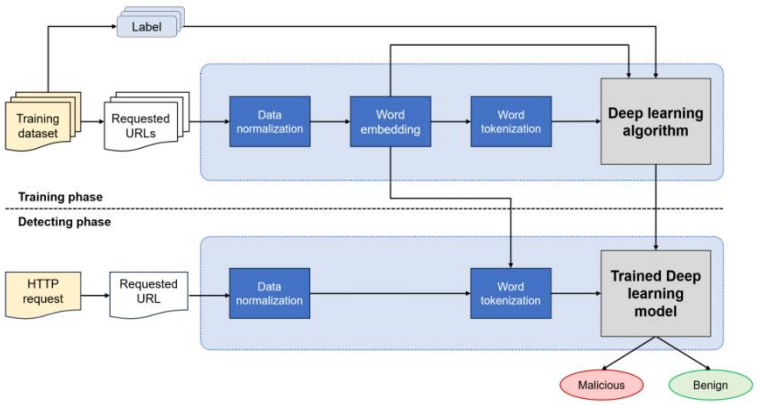

Рис. 1. Загальна архітектура DL-WAD для виявлення веб-атак за HTTP-запитами. Нормалізація, кодування вхідних даних, класифікація глибинною моделлю [7]

На рівні архітектурних підходів у літературі послідовно обґрунтовується доцільність комбінування різних парадигм навчання. Йдеться про навчання з учителем, без учителя та напівкероване навчання. Також наголошується на потребі інтегрувати їхні результати в єдину логіку прийняття рішень для проактивного виявлення і пом'якшення загроз. Ми вважаємо цей напрям обґрунтованим, оскільки різні парадигми природно компенсують слабкі місця одна одної. Зокрема, у роботах, присвячених побудові рамок хмарної безпеки з використанням методів ШІ та

машинного навчання, підкреслюється важливість гібридних і ансамблевих підходів. Вони можуть зменшувати хибно-позитивні спрацювання та підвищувати здатність до виявлення нетипових атак, що узгоджується з ризик-орієнтованим трактуванням сигналів виявлення [8]. Праці з фокусом на поведінковій аналітиці у хмарі демонструють, що сигнали UBA (User Behavior Analytics) та UEBA (User and Entity Behavior Analytics) підвищують чутливість до компрометації та зловживань доступом. На нашу думку, практичний висновок тут полягає в тому, що контекст користувача і шаблони взаємодії доцільно розглядати як фактори ризикової оцінки, а не як другорядну телеметрію [9].

Окремий пласт джерел формує критерії, пов'язані з практичною експлуатацією систем виявлення. Ми вважаємо переконливим акцент авторів на тому, що недостатньо лише виявити підозрілу подію. Важливо скоротити повний цикл інциденту та забезпечити керовану реакцію. Це вимагає стабільних рішень і відтворюваних політик їх застосування. У дослідженнях щодо автоматизації реагування на інциденти на основі методів ШІ (штучний інтелект) підкреслюється роль інтелектуальної підтримки первинного аналізу та кореляції подій, а також автоматизованого запуску дій реагування. На нашу думку, саме ці процеси роблять вимоги до стабільності оцінок і якості порогової політики не менш важливими, ніж значення інтегральних метрик [10]. У роботі, присвяченій забезпеченню відповідності вимогам у хмарі з використанням машинного навчання, систематизовано набір метрик якості виявлення, серед них accuracy (точність), precision (прогноз), recall (влучність), F1-оцінка, ROC/AUC (Receiver Operating Characteristic, Area Under Curve). Також наведено операційні показники ефективності, зокрема час реакції та скорочення середнього часу до виявлення і середнього часу до реагування, MTTD (Mean Time To Detect) і MTTR (Mean Time To Repair) [11]. Ми оцінюємо це як важливе підґрунтя для того, щоб оцінювання системи відображало реальні витрати на обробку хибних сповіщень про небезпеку.

Оскільки машинне навчання стає складовою хмарної безпеки, у літературі окремо підкреслюється потреба враховувати безпеку самої ML-компоненти як частину критеріїв надійності рішення. Систематичний огляд загроз для хмарно розміщених моделей, включно з poisoning, evasion, model stealing та суміжними класами атак, пропонує підхід, за якого оцінювання системи має враховувати не лише здатність до виявлення, а й стійкість моделі та ризики компрометації конвеєра обробки [12]. На нашу думку, цей аспект часто недооцінюють у прикладних роботах, хоча в реальній експлуатації він безпосередньо впливає на довіру до автоматизованого рішення.

Вимоги до масштабованості та сервісної реалізації моделей у хмарі підтримуються роботами, що описують розподілені та сервісні парадигми, а також хмарні способи розгортання аналітики. Огляд розподілених рішень і підходів "програмне забезпечення як послуга" (SaaS) у площині хмарних обчислень підкреслює, що практична цінність ML-компонент залежить від еластичності, продуктивності та здатності інтегруватися в хмарні сервіси з прогнозованими затримками [13]. Ми вважаємо, що для задач API/HTTP моніторингу це означає потребу в потоковій обробці, відтворюваному застосуванні порогових політик і можливості оновлювати моделі без порушення основного потоку обробки подій.

У контексті критеріїв ризику важливе місце займають аспекти управління і відповідності вимогам. Небезпека у хмарі часто визначається не лише технічним фактом аномалії, а й контекстом доступу, політиками та наслідками витоку даних. Роботи з фокусом на врядуванні та правових аспектах міграції даних у хмару

підкреслюють, що оцінка ризику має враховувати умови обробки даних у міжнародному середовищі [14]. Підходи до інтегрованої аналітики відповідності демонструють вимоги придатності до аудиту та відстежуваності, а також необхідність узгоджувати технічне виявлення з регуляторними вимогами [15]. Окремо виділяється клас ризиків, пов'язаних із неконтрольованим зберіганням секретів у вихідному коді. Компрометація ключів і токенів може напряму призводити до несанкціонованого доступу до хмарних ресурсів і суттєво змінювати оцінку небезпеки події [16]. Практичні рекомендації з Zero Trust у хмарних середовищах задають рамку контекстної довіри, яка включає безперервну валідацію, мінімізацію привілеїв і прив'язку рішень доступу та реагування до контексту [17]. На нашу думку, саме ця рамка підсилює вимогу до ризик-орієнтованого рішення, а не лише до детектора аномалій.

Найбільш безпосередньо до тематики цієї роботи, на нашу думку, наближаються підходи, що розглядають оцінювання ризику як результат узгодження сигналів виявлення з контекстом і керованими політиками прийняття рішень. Ми оцінюємо як ключові дві практичні вимоги. Перша вимога це наявність єдиної інтерпретованої шкали оцінки ризику, яка робить порівнюваними виходи моделей різної природи. Друга вимога це кероване обмеження хибно-позитивних спрацювань у потоці, оскільки саме вони визначають навантаження на первинний аналіз та операційну придатність системи. У межах цих вимог актуальним стає конвеєрний підхід, де результати керованих моделей та моделей нетипової поведінки узгоджуються в одну системну оцінку, а робоча точка визначається політикою, пов'язаною з допустимим рівнем хибних сповіщень про небезпеку [18], [19].

Як наслідок, огляд літератури формує підґрунтя для ML-орієнтованої постановки задачі цієї роботи. На нашу думку, потрібна система, яка в потоковому режимі формує узгоджену оцінку ризику для нормальних подій та атак, забезпечує керованість порогів і стабільність рішень, а також надає достатню інтерпретацію для використання в процесах SOC та DevSecOps.

Мета статті. Метою дослідження є розроблення та експериментальна перевірка ризик-орієнтованого підходу до розпізнавання небезпеки в API та HTTP-трафіку хмарного середовища на основі числових моделей машинного навчання. Підхід поєднує основний керований канал машинного навчання та додатковий канал виявлення нетипової поведінки, який підсилює чутливість до нових або рідкісних сценаріїв атак. Методика передбачає узгодження виходів моделей у спільній ймовірнісній шкалі ризику $p(x)\in[0,1]$ із застосуванням калібрування та керованих корекцій, а також контроль хибно-позитивних спрацювань через вибір порога за FPR-бюджетом (False Positive Rate).

Актуальність дослідження зумовлена практичною потребою підвищення ефективності захисту хмарних сервісів у середовищах, де API є основним каналом взаємодії і водночас основною поверхнею атаки. Для експлуатації, на нашу думку, критично важливо відокремлювати події з підвищеним ризиком від шуму телеметрії, зменшувати хибно-позитивні спрацювання без суттєвої втрати чутливості та отримувати інтерпретовані результати, придатні для автоматизованого або напівавтоматизованого реагування в процесах SOC та DevSecOps. Очікуваний науково-прикладний ефект полягає у підвищенні практичної придатності виявлення атак завдяки відтворюваності конвеєра оцінювання, масштабованості інфраструктурної реалізації та керованій політиці робочої точки.

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Опис експериментальної постановки та даних. У межах цього дослідження ми розглядаємо задачу автоматизованого розпізнавання небезпечних подій у хмарному середовищі на основі числових моделей машинного навчання. На практиці підхід поєднує два взаємодоповнювальні ML-канали. Перший канал є керованим і навчається на розмічених даних. Другий канал виявляє нетипову поведінку та працює як додатковий сигнал для нових або рідкісних сценаріїв. Об'єктом аналізу є подія x, яка відповідає окремому мережевому або API-запиту чи запису телеметрії та описується вектором структурованих числових ознак.

Для узгодженої математичної постановки подію x подаємо як вектор числових характеристик:

$$x = (x_{num}) \quad (1),$$

де x є подією, тобто одним запитом або одним записом телеметрії, що підлягає оцінюванню; x_{num} вектором числових ознак події, до яких належать статистики, лічильники, агрегати, технічні індикатори та похідні характеристики запиту.

Надалі під оцінкою розуміється скалярна величина, що відображає рівень ризику події. Для порівняння між моделями, оцінки приводяться до шкали $p(x)\in[0,1]$ після калібрування. Для оцінювання якості моделей вводиться бінарна мітка класу:

$$y \in \{0, 1\} \quad (2),$$

де y є міткою класу; y = 0 відповідає нормальній події (benign), а y = 1 відповідає атаці (attack).

Набір даних розбивається на підвибірки з різним призначенням, а саме навчальну, валідаційну, калібрувальну та тестову. На нашу думку, таке розбиття є принципово важливим для коректності експерименту, оскільки система включає моделі різних типів і окремі процедури калібрування оцінок та вибору порога спрацювання. Моделі каналу нетипової поведінки навчаються на даних нормальної поведінки, формують статистичне представлення норми та реагують на відхилення від неї. Керовані моделі навчаються на змішаних даних, тобто на нормальних подіях та атаках, і безпосередньо оптимізують розділення класів $y=0$ та $y=1$. Калібрування оцінок і вибір порога виконуються на окремій калібрувальній підвибірці, відокремленій від тестової, що зменшує ризик зміщення оцінювання та штучного завищення показників. Тестова підвибірка використовується лише для фінальної перевірки узагальненості.

Критерії оцінювання, які використовуються в розділі результатів, охоплюють кілька взаємопов'язаних аспектів. Якість розділення нормальних подій та атак аналізується за допомогою ROC-AUC (Receiver Operating Characteristic – Area Under the Curve), PR-AUC (Precision-Recall Area Under the Curve), F1, точності та повноти. Керований контроль хибно-позитивних спрацювань забезпечується через обмеження FPR і вибір порога за принципом бюджету хибних сповіщень про небезпеку. Узгодження оцінок у єдиній шкалі [0,1] підтримується за рахунок калібрування. Стійкість у потоковому застосуванні підсилюється за рахунок керованих політик стабілізації, зокрема за рахунок механізмів, які зменшують коливання рішень при малих змінах оцінки.

Перелік фактично задіяних моделей і каналів. У реалізованій системі оцінювання ризику ми використовуємо три функціональні компоненти, два з яких є ML-каналами, а третій є модулем об'єднання. Кожен канал формує власні оцінки, після чого модуль об'єднання приводить їх до узгодженого системного рішення, яке далі може бути використане в практичних процесах моніторингу та реагування:

1. Канал виявлення нетипової поведінки. Цей канал, на нашу думку, є важливим для фіксації подій, що відрізняються від звичайних режимів роботи сервісу за числовими ознаками. Його роль полягає в тому, щоб формувати сигнал відхилення від норми навіть у ситуаціях, коли конкретний сценарій атаки не був представлений у навчальних даних. У межах каналу застосовано такі моделі, як Isolation Forest, One-Class SVM, Dense Autoencoder та LSTM-AE.
2. Керований канал машинного навчання. Цей канал є основним числовим каналом оцінювання ризику, оскільки моделі навчаються на розмічених даних і безпосередньо оптимізують розділення нормальних подій та атак. У реалізації використано Logistic Regression, HistGradientBoosting, RandomForest і ExtraTrees. Додатково використовується rules_clf, який задає набір явних правил для числових ознак і, на нашу думку, підсилює інтерпретованість і стабільність щодо частини типових сценаріїв.
3. Модуль об'єднання рішень (fusion). Модуль об'єднання приймає оцінки з керованого числового каналу та каналу нетипової поведінки, після чого формує підсумкову оцінку ризику. На цьому етапі застосовується політика прийняття рішення та механізми стабілізації, що дозволяє узгоджувати сигнали різної природи в межах ML-підходів і підвищувати стійкість рішень у потоці подій.

У підсумку керований числовий ML-канал дає основну оцінку ризику, канал нетипової поведінки додає сигнал відхилення від звичайних режимів, а модуль об'єднання рішень (fusion) зводить ці оцінки до одного рішення та стабілізує його в потоці подій.

Загальна формалізація конвеєра оцінювання ризику. Загальна формалізація конвеєра оцінювання ризику. У цьому підрозділі ми формалізуємо повний ML-конвеєр оцінювання ризику, тобто шлях від сирих виходів окремих моделей до фінального рішення системи. Оскільки різні моделі природно повертають оцінки в різних шкалах і з різною інтерпретацією, у системі застосовується калібрування, яке приводить ці оцінки до спільної ймовірнісної шкали ризику ["0,1"]. На нашу думку, саме цей крок є критичним. Він робить порівнюваними виходи моделей різних типів, дає підставу для узгодженого вибору порогів і дозволяє коректно об'єднувати сигнали в одному правилі прийняття рішення.

Крок 1. Сира оцінка кожної моделі.

Для події x кожна модель з індексом k повертає сиру оцінку:

$$r_{k(x)} \quad (3),$$

де k є індексом моделі; $r_{k(x)}$ є сирою оцінкою аномальності або сирим сигналом моделі k для події x; x є подією у вигляді (1).

Крок 2. Калібрування у ймовірнісну оцінку ризику.

Сиру оцінку перетворюємо на ймовірнісну оцінку ризику:

$$P_{k(x)} \in [0, 1] \quad (4),$$

де $p_{k(x)}$ є ймовірнісною оцінкою ризику від моделі k після калібрування; [0, 1] є уніфікованою шкалою, яка дозволяє порівнювати та агрегувати оцінки різних моделей.

Крок 3. Основна ML-оцінка (керований числовий канал).

Оцінки керованих моделей агрегуються в єдину основну оцінку:

$$P_{ML,primary}(x)= Agg_{primary}\left(\{p_k(x)\}_{k \in primary}\right) \quad (5),$$

де $P_{ML,primary}(x)$ є основною оцінкою ризику за керованим числовим ML-каналом; $Agg_{primary}(\cdot)$ є оператором агрегації оцінок для основного каналу; primary є множиною індексів моделей, що входять до керованого каналу; $\{p_k(x)\}_{k \in primary}$ є набором каліброваних оцінок моделей з множини primary.

Крок 4. Додаткова ML-оцінка (канал виявлення нових або невідомих аномалій)

Оцінки моделей "відхилення від норми" агрегуються у додаткову оцінку:

$$P_{ML,aux}(x)= Agg_{aux}\left(\{p_k(x)\}_{k \in aux}\right) \quad (6),$$

де $P_{ML,aux}(x)$ є додатковою оцінкою ризику, що характеризує нетиповість або відхилення від норми; $Agg_{aux}(\cdot)$ є оператором агрегації оцінок для додаткового каналу; aux є множиною індексів моделей, що входять до каналу виявлення нових або невідомих аномалій; $\{p_k(x)\}_{k \in aux}$ є набором каліброваних оцінок моделей з множини aux.

Крок 5. Підсумкова оцінка після об'єднання рішень (fusion)

Фінальна оцінка ризику обчислюється як функція об'єднання сигналів з урахуванням контексту:

$$p_{final}(x)=F\left(p_{ML,primary}(x),p_{NLP}(x),p_{ML,aux}(x),context\right) \quad (7),$$

де $p_{final}(x)$ є підсумковою фінальною оцінкою ризику події x; $F(\cdot)$ є функцією об'єднання оцінок, тобто політикою об'єднання рішень; context є контекстом події, який включає додаткові атрибути, умови політик і службові сигнали та може впливати на підсумкову оцінку.

На нашу думку, врахування контексту на цьому етапі є ключовим для того, щоб одна й та сама технічна ознака не призводила до однакового рішення в різних середовищах експлуатації.

Крок 7. Рішення про аномалію.

Після отримання фінальної оцінки застосовується поріг спрацювання:

$$1_{anomaly}=\left[p_{final(x)} \geq \tau_{fusion}\right] \quad (8),$$

де $1_{anomaly}$ є індикатором рішення системи, тобто 1 означає, що подію визнано аномальною або небезпечною, а 0 означає, що подію визнано нормальною; $[\cdot]$ є індикаторною функцією, що дорівнює 1, якщо умова істинна, і 0 інакше; τ_{fusion} є порогом спрацювання для фінальної оцінки ризику.

Додатково після об'єднання можуть застосовуватися механізми стабілізації. До них належать гістерезис і правила, що зменшують коливання рішення при малих змінах оцінки. На нашу думку, ці механізми підвищують керованість і практичну придатність системи в потоковому режимі, оскільки зменшують кількість нестійких переключень між станами при близьких до порога значеннях. Ці елементи деталізуються у наступних підпунктах розділу.

Математичні засади ML-моделей та зміст сирих оцінок. У цьому підпункті ми пояснюємо, що саме означають сирі оцінки, які повертають моделі числового каналу машинного навчання, і як ці оцінки переходять до узгоджених ймовірнісних оцінок ризику. Кожна модель k формує сирий сигнал $r_k(x)$ у (3). Важливо підкреслити, що для різних класів моделей цей сигнал має різний зміст, тому ми свідомо розглядаємо його як проміжний результат, який надалі приводиться до спільної шкали.

Для моделей виявлення відхилень від норми сирий сигнал є некаліброваною оцінкою аномальності a(x), яка зростає у міру відхилення від нормальної поведінки. Для керованих моделей машинного навчання вихід часто вже має вигляд ймовірності $p(x)=P(y=1|x)$. Проте навіть у цьому випадку, на нашу думку, доречно розглядати його як вхід до модуля калібрування. Це потрібно, щоб узгодити шкалу між різними моделями, стабілізувати порогові рішення і мати керований контроль FPR у єдиній політиці.

Нижче послідовно наведено математичні означення для кожної використаної моделі, протокол навчання, а також інтерпретацію сирого сигналу до калібрування.

Isolation Forest. У реалізації Isolation Forest (Рис. 2, Рис. 3) сирий статистичний вихід береться через $score_{samples}(x)$, а $decision_{function}$ використовується як запасний варіант. Далі ми узгоджуємо напрям шкали так, щоб більші значення відповідали вищому ризику. Це задається формулою:

$$a_{if}(x)=-score_{samples}(x) \quad (9),$$

де $a_{if}(x)$ є скалярною сирою оцінкою аномальності події x до калібрування; $score_{samples}(x)$ є статистикою Isolation Forest, яка зазвичай є більшою для нормальних об'єктів і меншою для аномальних; знак "-" інвертує напрям шкали так, щоб $a_{if}(x)$ зростала разом зі зростанням аномальності; x є подією, визначеною у (1).

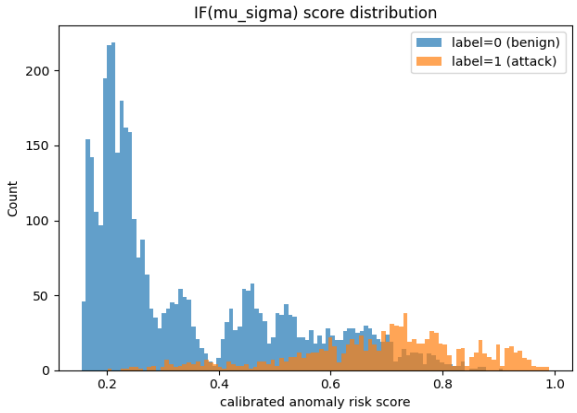

Рис 2. Розподіл каліброваних оцінок ризику для Isolation Forest із mu-sigma калібруванням

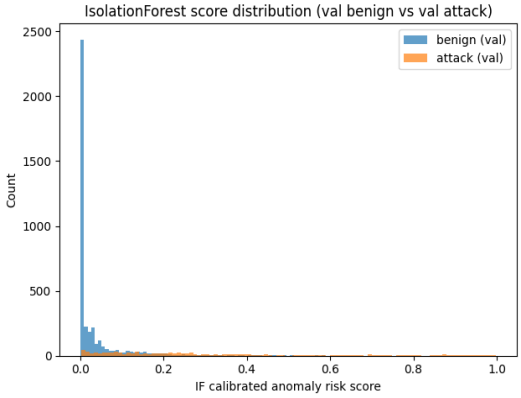

Рис. 3. Порівняння розподілів оцінок для нормальних подій та атак (валідаційний зріз) для Isolation

Інтерпретація $a_{if}(x)$ пов'язана з ідеєю ізоляції. Нетипові об'єкти у випадкових деревних розбиттях, як правило, ізолюються за меншу кількість кроків, і це відображається у статистиці моделі. Після інверсії за (9) ми отримуємо інтуїтивно узгоджену шкалу, де зростання $a_{if}(x)$ відповідає посиленню сигналу нетиповості.

Навчання моделі виконується на підвибірці нормальної поведінки (benign-only). Калібрування оцінок і вибір порогу проводяться окремо на змішаній калібрувальній підвибірці, відокремленій від тесту. Такий протокол потрібен, щоб коректно інтерпретувати оцінки у шкалі ризику і водночас тримати під контролем хибно-позитивні спрацювання.

One-Class SVM. One-Class SVM формалізує межу "нормальної області" у просторі ознак через ядрове відображення та опорні вектори. Базова функція рішення задається як:

$$f(x)=\sum_i a_i k(x_i,x)-p \qquad (10),$$

де $f(x)$ є значенням функції рішення для події x; x_i є опорними об'єктами, які визначають межу; a_i є ваговими коефіцієнтами; $k(x_i,x)$ є ядровою функцією подібності, наприклад RBF; p є пороговим зсувом; Σ_i означає сумування за опорними об'єктами.

Для практичної інтеграції в конвеєр нам потрібна шкала "чим більше, тим ризик вищий". Тому в реалізації, так само як і для Isolation Forest, узгоджується знак:

$$a_{ocsvm}(x)=-f(x),\ a_{ocsvm}(x)=\left(-score_{samples(x)}\right) \qquad (11),$$

де $a_{ocsvm}(x)$ є сирою оцінкою аномальності події x до калібрування; $f(x)$ береться з (10); $score_{samples(x)}$ є альтернативною статистикою, якщо використовується відповідний інтерфейс оцінювання; інверсія знаку забезпечує зростання $a_{ocsvm}(x)$ при відхиленні від нормальної області.

Окремо підкреслимо масштабування числових ознак. Для One-Class SVM (Рис. 4) це критично, оскільки ядрова геометрія є чутливою до масштабів компонент. Тому стандартизація або стійке масштабування виступають необхідною умовою стабільної роботи моделі.

Навчання виконується тільки на підвибірці нормальних подій. Калібрування та вибір порогу виконуються на змішаній калібрувальній підвибірці, окремо від тестової.

ISSN 2663 – 4023

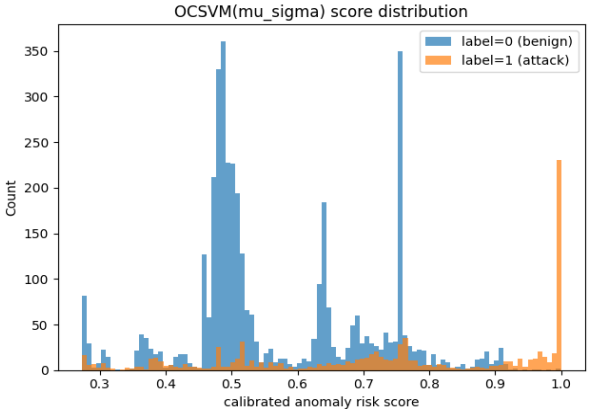

Рис. 4. Розподіл каліброваних оцінок ризику для One-Class SVM із mu-sigma калібруванням

Dense Autoencoder. Dense Autoencoder (щільний автоенкодер) навчається відтворювати нормальну поведінку через мінімізацію похибки реконструкції. Сирий сигнал аномальності задається середньоквадратичною похибкою реконструкції:

$$a_{ae}(x)=\frac{1}{d}\sum_{j=1}^{d}\left(x_j-\hat{x}_j\right)^2 \quad (12),$$

де $a_{ae}(x)$ є сирою оцінкою аномальності події x до калібрування; d є розмірністю вектора числових ознак, тобто кількістю компонент у x_{num}; x_j є компонентою вхідного вектора j; $\hat{x}_j$ є компонентою реконструйованого вектора j; $\Sigma_{j=1}^{d}$ означає сумування за всіма компонентами.

Зміст (12) інтуїтивний. Модель, навчена на нормальних даних, відтворює типові структури та залежності у x_{num} з малою похибкою. Для нетипових подій похибка реконструкції зростає, і це природно трактувати як посилення сигналу відхилення від норми.

Навчання автоенкодера можна записати як мінімізацію очікуваної похибки реконструкції на нормальній вибірці:

$$\min_{\theta} E_{x\sim D_{normal}}\left[\frac{1}{d}\sum_{j=1}^{d}\left(x_j-\hat{x}_j(\theta)\right)^2\right] \quad (13),$$

де θ є параметрами автоенкодера; D_{normal} є розподілом або вибіркою нормальних подій; $\Sigma[\cdot]$ є математичним сподіванням за даними нормальної поведінки; $\hat{x}_j(\theta)$ є реконструкцією, що залежить від параметрів θ.

Калібрування виконується у два етапи. Спочатку оцінюються параметри розподілу для тренувальних даних нормальних подій і виконується початкова нормалізація. Далі на калібрувальній підвибірці обирається робочий варіант калібрування, який дає найкращу узгодженість шкали і потрібний контроль помилкових спрацювань (Рис. 5, Рис. 6).

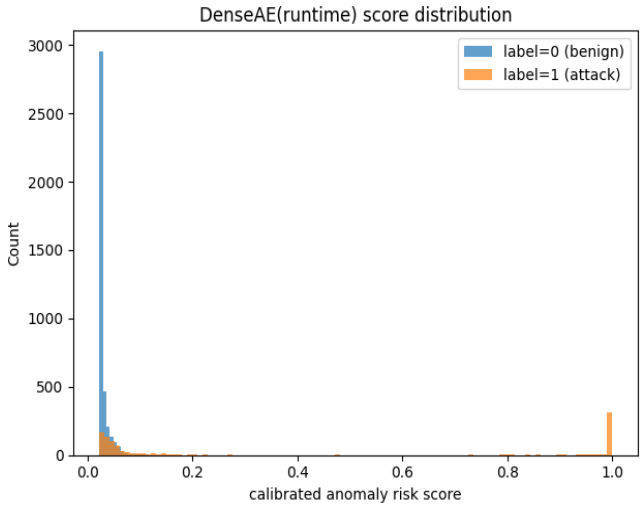

Рис. 5. Розподіл каліброваних оцінок ризику для Dense Autoencoder

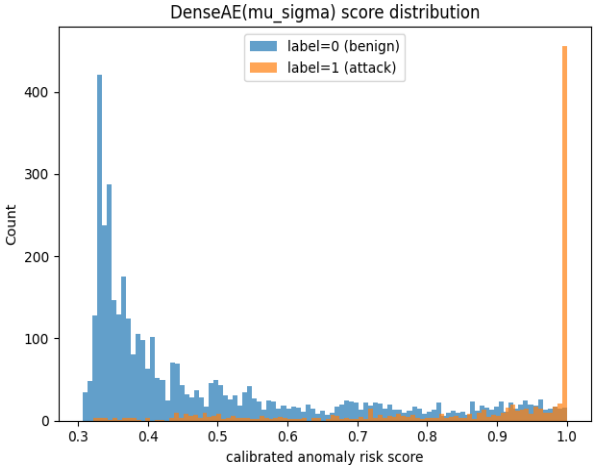

Рис. 6. Розподіл каліброваних оцінок ризику для Dense Autoencoder із mu-sigma

LSTM-AE. LSTM-AE узагальнює реконструкційний підхід для часових послідовностей. Модель відтворює фрагмент динаміки нормальної поведінки, а сирий сигнал визначається середньоквадратичною похибкою реконструкції для послідовності:

$$a_{lstm}(x)=\frac{1}{T*d}\sum_{t=1}^{T}\sum_{j=1}^{d}\left(x_{t,j}-\hat{x}_{t,j}\right)^2 \quad (14),$$

де $a_{lstm}(x)$ є сирою оцінкою аномальності послідовності, пов'язаної з подією x, до калібрування; T є довжиною часового вікна; d є кількістю ознак у кожному часовому кроці; $x_{t,j}$ є значенням j-тої ознаки на кроці t; $\hat{x}_{t,j}$ є реконструйованим значенням; подвійне сумування охоплює і час, і ознаки.

Як і Dense Autoencoder, LSTM-AE навчається тільки на даних нормальних подій. Його практична перевага, на нашу думку, полягає у здатності моделювати саме нормальну динаміку, тому порушення часових залежностей можуть проявлятися навіть тоді, коли окремий статичний зріз ознак не виглядає явно підозрілим (Рис. 7, Рис. 8).

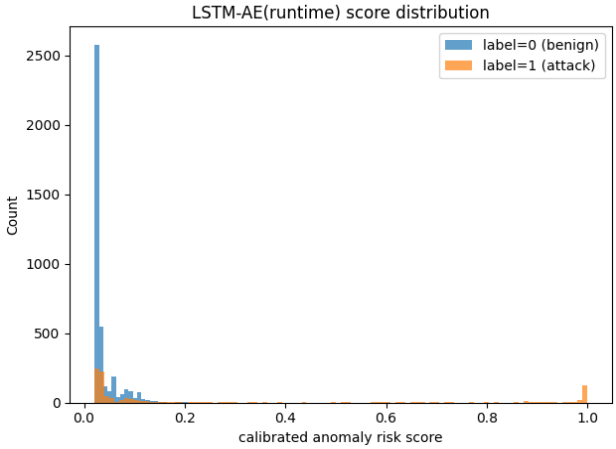

Рис. 7. Розподіл каліброваних оцінок ризику для LSTM-AE

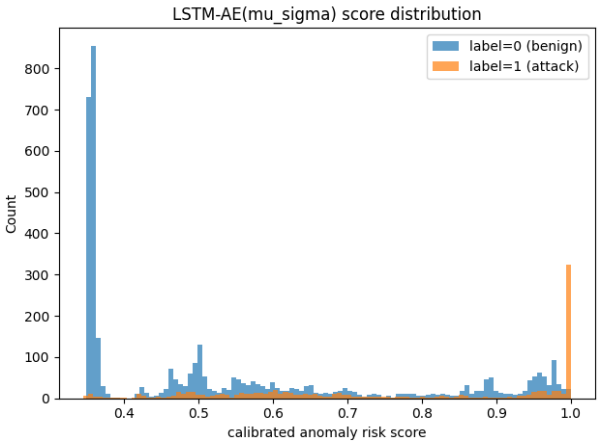

Рис. 8. Розподіл каліброваних оцінок ризику для LSTM-AE із mu-sigma калібруванням

Керовані числові моделі. Керовані числові моделі формують основний канал оцінювання ризику, оскільки навчаються на розміченому змішаному трафіку і безпосередньо оптимізують розділення класів y=0 та y=1. На відміну від моделей відхилення від норми, які відтворюють портрет нормальної поведінки, керовані моделі наближають умовну ймовірність атаки за наявними ознаками.

Нехай маємо навчальну вибірку $\{(x_i,y_i)\}_{i=1}^{n}$, де x_i є подією або її числовим поданням, а $y_i \in \{0,1\}$ є міткою класу. Мета керованого навчання полягає у побудові параметризованої функції $p_\theta(x)$, яка відображає подію у значення в [0,1] і інтерпретується як оцінка умовної ймовірності $P(y=1|x)$.

Логістична регресія, яку ми використовуємо як базову модель, задає ймовірність через функцію сігмоїди від лінійного прогнозу:

$$p(x)=\sigma(w^T x+b) \quad (15),$$

де $p(x)$ є оцінкою ймовірності того, що подія x є атакою; $\sigma(\cdot)$ є сигмоїдною функцією; w є вектором ваг; $w^T x$ є скалярним добутком параметрів на вектор ознак; b є зсувом; x є вектором числових ознак події, тобто компонентою x_{num} з (1).

Для повноти зафіксуємо означення сигмоїди:

$$\sigma(z)=\frac{1}{1+\exp(-z)} \quad (16),$$

де z є дійсним аргументом, а $exp(\cdot)$ є експонентою.

Деревні ансамблі HistGradientBoosting, RandomForest, ExtraTrees також формують оцінку в [0,1], але внутрішня структура цих моделей є нелінійною. У загальному вигляді їхній вихід доцільно трактувати як умовну ймовірність:

$$p(x)=P(y=1|x) \quad (17),$$

де $P(y=1|x)$ є умовною ймовірністю атаки за умови спостереження ознак x, а $p(x)$ є оцінкою, яку повертає модель.

На практиці вона може відповідати усередненій частці позитивів у листках або логістичному перетворенню внутрішньої оцінки, залежно від реалізації.

Причина високої ефективності деревних ансамблів у цій задачі полягає у здатності моделювати нелінійні взаємодії ознак. Це особливо важливо для комбінацій на кшталт кількості параметрів запиту, ентропії даних, частки підозрілих символів, розміру тіла запиту та інших похідних характеристик, які важко відобразити лінійною моделлю без ручного конструювання складних перетворень.

Навчання керованих моделей у стандартній постановці можна подати як мінімізацію логістичної втрати на вибірці з n прикладів:

$$\min_{\theta} L(\theta)=-\sum_{i=1}^{n}\left[y_i \log\left(p_\theta(x_i)\right)+\left(1-y_i\right)\log\left(1-p_\theta(x_i)\right)\right] \quad (18),$$

де θ є параметрами моделі, для логістичної регресії θ=(w,b), а для ансамблів θ є сукупністю параметрів дерев; $p_\theta(x_i)$ є виходом моделі для прикладу x_i; y_i є істинною міткою класу; $\log(\cdot)$ є натуральним логарифмом.

Формула (18) напряму пов’язує навчання з ймовірнісною інтерпретацією виходу. Модель отримує штраф за високу впевненість у неправильному класі і за недостатню впевненість у правильному. Саме тому вихід керованих моделей зручно трактувати як ймовірність атаки. Водночас у системному конвеєрі ми розрізняємо два рівні. Перший рівень це внутрішня логіка конкретного алгоритму, яка формує $p_\theta(x_i)$. Другий рівень це узгодження шкали між моделями, де навіть $p_\theta(x_i)$ може додатково проходити калібрування. На нашу думку, це є ключовим для того, щоб різні ансамблі працювали у порівнюваній шкалі ризику і підлягали єдиній політиці вибору порогу та контролю FPR.

Окремо зазначимо інтерпретацію сирого сигналу для керованих моделей. Для логістичної регресії сирим сигналом до застосування сигмоїди є лінійний прогноз $s(x)=w^Tx+b$, а $p(x)$ отримується через (15). Для деревних ансамблів сирим є внутрішня оцінка, яка формується сумою внесків дерев у підсилюванні або усередненням голосів дерев у випадковому лісі та ExtraTrees, після чого видається $p(x)$ у шкалі [0,1]. Саме тому у загальному вигляді для керованого каналу природно вважати, що $r_k(x)$ вже є ймовірнісною оцінкою, яка надалі узгоджується з іншими каналами через єдину процедуру калібрування.

Розподіли оцінок ризику для керованих моделей наведено на Рис. 9-Рис. 12.

ISSN 2663 – 4023

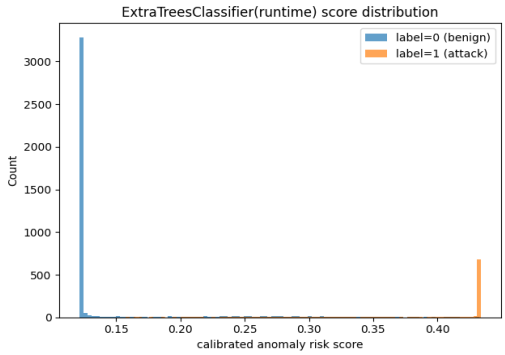

Рис. 9. Розподіл оцінок ризику для ExtraTreesClassifier, нормальні події проти атак

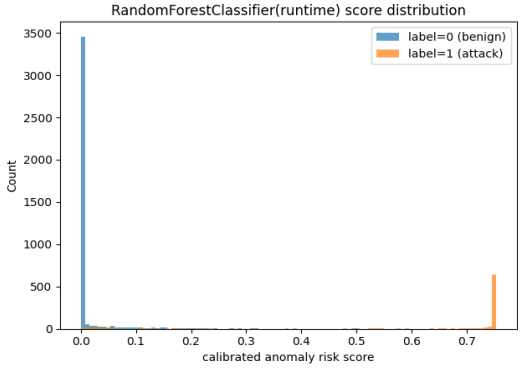

Рис. 10. Розподіл оцінок ризику для RandomForestClassifier, нормальні події проти атак

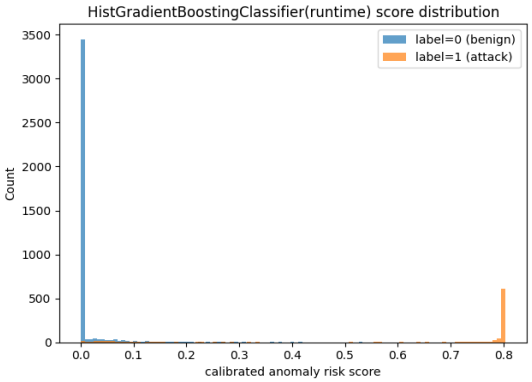

Рис. 11. Розподіл оцінок ризику для HistGradientBoostingClassifier, нормальні події проти атак

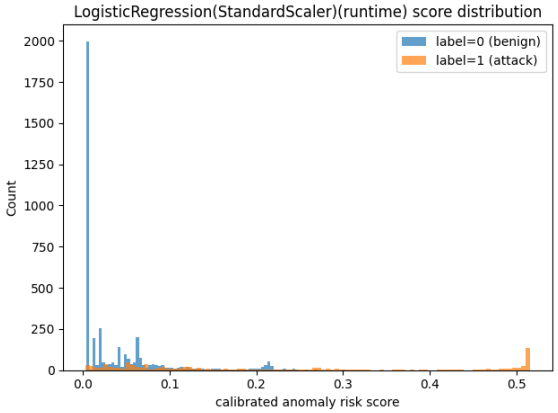

Рис. 12. Розподіл оцінок ризику для LogisticRegression, нормальні події проти атак

Калібрування оцінок, приведення до ймовірнісного ризику та роль температури. У нашій системі одночасно працюють моделі різної природи (ансамблі дерев рішень, моделі відхилення від норми та реконструкційні нейромережеві моделі), тому їхні виходи спочатку мають різні шкали й різний зміст (статистика моделі, похибка реконструкції тощо). Щоб зробити ці оцінки порівнюваними й придатними до спільної політики прийняття рішень, ми приводимо сирий сигнал до єдиної ймовірнісної шкали ризику $p(x)\in[0,1]$. На нашу думку, саме калібрування є критичним кроком для коректного об'єднання сигналів у модулі об'єднання та для керованого контролю хибно-позитивних спрацювань через вибір порогу.

Загальна мета калібрування – побудувати відображення $C(\cdot)$, яке переводить сирий сигнал $a(x)$ у ймовірнісну оцінку ризику:

$$p=C(a(x)) \tag{19},$$

де x є подією (один запит/один запис телеметрії), що оцінюється; $a(x)$ є сирим сигналом моделі (або інша сира оцінка), який монотонно зростає зі зростанням нетиповості/ризику; $C(\cdot)$ є калібрувальним відображенням, оцінене на калібрувальній підвибірці; $p(x)$ є ймовірнісною оцінкою ризику, яка узгоджена в інтервалі $[0,1]$.

У роботі ми використовуємо просту та відтворювану процедуру калібрування, яка поєднує нормування сирого сигналу за параметрами центру μ і масштабу σ та подальше перетворення через сигмоїдну функцію. Додатково вводимо параметр температури $T>0$, що керує різкістю ймовірностей:

$$p(x)=\sigma\left(\frac{(a(x)-\mu)}{\sigma T}\right),\sigma(z)=\frac{1}{1+exp(-z)} \tag{20},$$

де μ є оцінкою центру розподілу сирого сигналу $a(x)$ на калібрувальній підвибірці; $\sigma>0$ є оцінкою масштабу розподілу $a(x)$ на калібрувальній підвибірці; $T>0$ є параметром температури, що масштабує нормоване значення та керує м'якістю або різкістю розподілу $p(x)$; $\sigma(\cdot)$ є сигмоїдною функцією; z є аргументом сигмоїдної функції. На нашу думку, введення T є практично корисним, оскільки за $T>1$ розподіл $p(x)$ стає м'якшим, а за $0<T<1$ розподіл $p(x)$ стає різкішим. За потреби μ та σможуть оцінюватися робастно, щоб зменшити вплив викидів.

Практично важливо, що μ та σ можуть обчислюватися надійно, наприклад на основі медіани та стійкої оцінки розкиду, щоб зменшити вплив викидів і забезпечити стабільність для моделей, чутливих до хвостів розподілу.

Корекція за апріорною часткою атак. Навіть після калібрування значення $p(x)$відображають ймовірності, узгоджені з часткою атак у калібрувальній підвибірці. Оскільки у реальному розгортанні очікувана частка атак може відрізнятися, ми застосовуємо корекцію через відношення шансів.

$$q=\frac{p}{1-p},\quad q'=q\cdot\frac{\pi_{dep}/(1-\pi_{dep})}{\pi_{cal}/(1-\pi_{cal})},\quad p'=\frac{q'}{1+q'} \tag{21},$$

де є π_{cal} апріорною часткою атак у калібрувальній підвибірці. π_{dep} є очікуваною апріорною часткою атак у середовищі розгортання.

Зміст (21) полягає в тому, що ми переводимо p у q, масштабуємо q коефіцієнтом, який компенсує зміну апріорної частки атак, і повертаємося до ймовірнісної шкали, отримуючи p'.

Вибір порогу (робочої точки) та критерій FPR. Після калібрування і, за потреби, корекції кожна модель або агрегований канал формує узгоджену оцінку ризику $s(x)\in[0,1]$. Далі ми задаємо поріг τ, який перетворює неперервну оцінку на двійкове рішення:

$$\hat{y}(x)=1[s(x)\geq\tau] \quad (22),$$

де $\hat{y}(x)\in\{0,1\}$ є прогнозованою міткою події; значення $\hat{y}(x)=1$ означає, що подію визнано небезпечною, а значення $\hat{y}(x)=0$ означає, що подію визнано нормальною; $s(x)$ є узгодженою оцінкою ризику події x; τ є порогом спрацювання; $1[\cdot]$ є індикаторною функцією, яка дорівнює 1 за істинності умови та 0 в іншому випадку.

У прикладній постановці для нас принциповим є керований контроль хибно-позитивних спрацювань, тому поріг визначаємо за заданим обмеженням FPR. На калібрувальній підвибірці беремо оцінки ризику для нормального класу та обираємо поріг як емпіричний квантиль рівня 1-α:

$$\tau = Q_{1-\alpha}(\{s_i: y_i = 0\}), 0 < \alpha < 1 \quad (23),$$

де α є заданою допустимою часткою хибно-позитивних спрацювань серед нормальних подій; s_i є оцінкою ризику для і-тої події на калібрувальній підвибірці; $y_i\in\{0,1\}$ є істинною міткою класу для і-тої події; множина $\{s_i:y_i=0\}$ є набором оцінок ризику для нормальних подій; $Q_{1-\alpha}(\cdot)$ є емпіричним квантилем рівня 1-α.

На нашу думку, така політика є більш придатною для експлуатації, ніж підбір порога за F1-мірою, оскільки напряму обмежує кількість хибних сповіщень і, відповідно, навантаження на первинний аналіз.

Агрегація оцінок в ансамблі моделей та інтерпретація усереднення у просторі логарифма відношення шансів. У системі ми використовуємо кілька моделей в одному числовому каналі, тому для кожної події x отримуємо набір уже каліброваних ймовірнісних оцінок ризику. На нашу думку, агрегація в ансамблі є потрібною з практичних причин, оскільки вона зменшує чутливість до випадкових збоїв окремої моделі та дає більш стабільну оцінку ризику в потоці.

Нехай для події x маємо M каліброваних оцінок ризику $p_i(x)\in(0,1)$, де i=1,…,M. Тоді агреговану оцінку ризику ми обчислюємо через усереднення у просторі логарифма відношення шансів із подальшим поверненням у шкалу $[0,1]$ через сигмоїдну функцію:

$$p_{agg}(x)=\sigma\left(\frac{1}{M}\sum_{i=1}^{M}\log\left(\frac{p_i(x)}{1-p_i(x)}\right)\right) \quad (24),$$

де $\sigma(\cdot)$ є сигмоїдною функцією (16), $\log(\cdot)$ є логарифмом, а вираз $\log\left(\frac{p_i(x)}{1-p_i(x)}\right)$ є логарифмом відношення шансів, тобто логарифмом відношення ймовірності атаки до ймовірність норми, величина $\frac{p_i(x)}{1-p_i(x)}$ є відношенням шансів для і-тої оцінки, а $\Sigma_{i=1}^{M}$ є середнім значенням цього логарифма по всій групі.

Інтерпретація (24) є такою. Якщо кілька моделей дають помірно підвищені оцінки ризику, то їхній спільний ефект підсилюється узгоджено, навіть якщо жодна з моделей не видає максимального значення. Водночас одиничний випадковий сплеск однієї моделі має менший вплив, ніж у режимі максимуму. Для числової стабільності на практиці значення $p_i(x)$доцільно обмежувати малим ε, щоб уникати значень, близьких до 0 або 1, при обчисленні логарифма.

Додатково система може підтримувати простіші альтернативи агрегації, які у деяких сценаріях зручні як базові або контрольні, а саме вибір максимуму серед $p_i(x)$ або арифметичне середнє $p_i(x)$. На нашу думку, ці варіанти є корисними для швидкої перевірки, однак основний робочий режим агрегації доцільно залишати таким, як у (24), оскільки він краще узгоджує внески моделей у ймовірнісній шкалі.

Застосування агрегації у ML-групах. Оператор (24) застосовується окремо для основної групи керованих моделей, що формує $p_{ML,primary}(x)$, та для додаткової групи моделей відхилення від норми, що формує $p_{ML,aux}(x)$. Далі ці агреговані оцінки передаються у модуль об'єднання (fusion) разом з іншими сигналами системи та використовуються під час застосування робочої точки, визначеної за FPR-бюджетом, відповідно до (23). Послідовність етапів показано на Рис. 13 і Рис. 14.

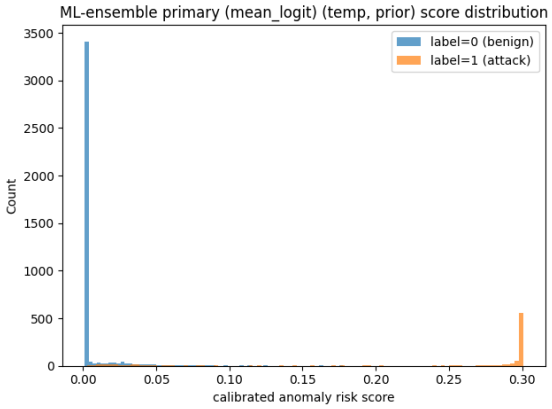

Рис. 13. Розподіл каліброваних оцінок основного ML-ансамблю (агрегація середнім у просторі логарифма відношення шансів; поріг за FPR-бюджетом)

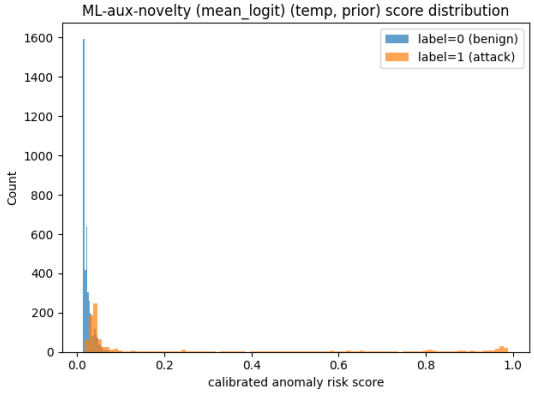

Рис. 14. Розподіл каліброваних оцінок додаткового ML-ансамблю для виявлення нових або невідомих аномалій (агрегація середнім у просторі логарифма відношення шансів; поріг за бюджетом FPR)

Модуль об'єднання рішень: фінальна системна оцінка та логіка підсилення новизни. Модуль об'єднання рішень: фінальна системна оцінка та підсилення новизни. Модуль об'єднання рішень інтегрує узгоджені ймовірнісні оцінки ризику від двох ML-каналів і формує підсумкову оцінку ризику $p_{final}(x) \in [0,1]$, яка далі використовується для рішення про аномалію. Керований числовий канал формує базову оцінку ризику $p_{ML,primary}(x)$, а канал нетипової поведінки формує додатковий сигнал новизни $p_{ML,aux}(x)$, який може підсилювати підсумковий ризик у випадках рідкісних або нових шаблонів.

Базова системна оцінка ризику. Базову оцінку ризику беремо з керованого числового каналу:

$$p_0(x)=p_{ML,primary}(x) \quad (25),$$

де $p_0(x)$ є базовою системною оцінкою ризику події xдо підсилення за рахунок нетипової поведінки; $p_{ML,primary}(x)$ є оцінкою ризику від керованого числового ML-каналу.

Підсилення за рахунок сигналу нетипової поведінки. Після об'єднання ML та NLP модуль fusion враховує додатковий сигнал нетипової поведінки від моделей, що мають відхилення від норми, які формують $p_{ML,aux}$. Це реалізовано як керована надбавка до p_{final} з обмеженням зверху 1:

$$\Delta_{aux}(x)=\gamma \max(0,p_{ML,aux}(x)-\tau_{aux}), p_{final}(x)=\min(1,p_0(x)+\Delta_{aux}(x)) \quad (26),$$

де $\Delta_{aux}(x)$ є надбавкою до ризику за нетипову поведінку; $p_{ML,aux}(x)$ є оцінкою ризику від каналу нетиповості; τ_{aux} є порогом, нижче якого нетиповість не підсилює підсумкову оцінку; $\gamma>0$ є коефіцієнтом сили підсилення; $p_{final}(x)$ є фінальною системною оцінкою ризику; $\min(1,\cdot)$ гарантує, що $p_{final}(x) \in [0,1]$.

Рішення про аномалію. Після отримання $p_{\text{final}}(x)$застосовується системний поріг:

$$\hat{y}(x)=1\left[p_{final}(x) \geq \tau_{fusion}\right] \quad (27),$$

де $\hat{y}(x)$ є індикатором рішення системи; $1[\cdot]$ є індикаторною функцією, що дорівнює 1, якщо умова істинна, і 0 інакше; τ_{fusion} є порогом спрацювання для фінальної оцінки ризику.

Додатково у потоковому режимі може застосовуватися гістерезис, щоб зменшити часті перемикання стану при коливаннях $p_{final}(x)$ поблизу порога.

Підсумок системи. Побудований конвеєр є системою оцінювання ризику подій у хмарному середовищі, у якій фінальне рішення формується як узгоджене поєднання двох ML-каналів і політик їх інтеграції. Основний числовий ML-канал на базі керованих моделей обчислює базову оцінку ризику за структурованими ознаками події та забезпечує розділення на норму або атаку на змішаному трафіку. Додатковий числовий канал виявлення нетипової поведінки формує сигнал відхилення від норми, який підсилює реакцію на нові або рідкісні сценарії через керовану надбавку Δ_{aux}. Модуль об'єднання рішень формує підсумкову оцінку $p_{final}(x)$, після чого

застосовується порогова політика з контролем хибно-позитивних спрацювань, визначена у підрозділі вибору робочої точки.

Візуальна верифікація розділення класів. Гістограми оцінок ризику. У цьому підрозділі показано, наскільки добре оцінки ризику від компонентів конвеєра розділяють нормальні події ($y = 0$)та атаки ($y = 1$). Гістограми доповнюють числові метрики тим, що дають уявлення про форму розподілів і зону перекриття, а також показують правий хвіст нормальних подій у зоні високого ризику, який визначає керованість FPR.

Усі наведені нижче гістограми інтерпретуються як розподіли ризику у шкалі [0,1], оскільки перед побудовою графіків сирі оцінки приводяться до єдиної каліброваної шкали. Порогова політика за FPR-бюджетом використовує параметр α, який є допустимою часткою хибно-позитивних спрацювань серед нормальних подій на калібрувальній підвибірці. Вибір порога τ є встановленням порога так, щоб частка нормальних подій праворуч від τ наближено дорівнювала α, тобто поріг відповідає квантилю нормального класу рівня 1-α.

Керований числовий канал. Характер розподілів для ExtraTrees.

Для керованої моделі очікується концентрація нормальних подій біля малих значень ризику та зсув атак у бік більших значень. Правий хвіст нормальних подій у зоні високого ризику є критичним, оскільки саме він визначає, наскільки просто утримувати заданий FPR-бюджет за обраним α (Рис. 15). Якщо зона перекриття між нормою та атаками є відносно малою, то поріг може бути встановлений нижче без порушення α, що зберігає чутливість до атак.

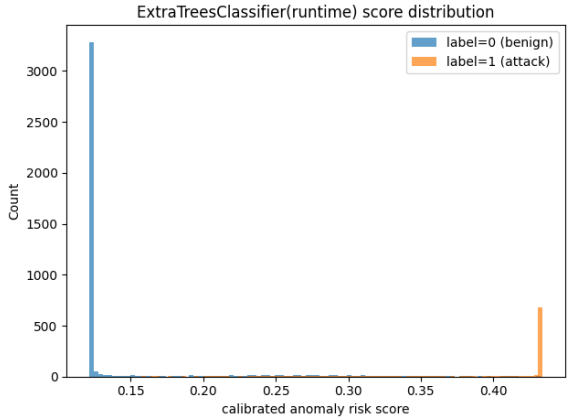

Рис. 15. Розподіл оцінок ризику керованої числової моделі ExtraTrees для нормальних подій та атак (візуальна перевірка розділення класів у шкалі [0, 1] *після узгодження оцінок)*

Канал нетипової поведінки. Порівняння Isolation Forest та Autoencoder. Для моделей, що навчаються на нормальній поведінці, форма гістограм зазвичай відрізняється від керованих моделей, оскільки частина атак може бути близькою до норми за числовими ознаками (Рис. 16, Рис. 17). Основною вимогою є те, що після калібрування нормальні події мають концентруватися біля малих значень, а атаки мають давати ширший розподіл із помітною масою у правій області. Такий правий зсув є сигналом нетиповості, який доповнює керований канал у модулі об'єднання (fusion).

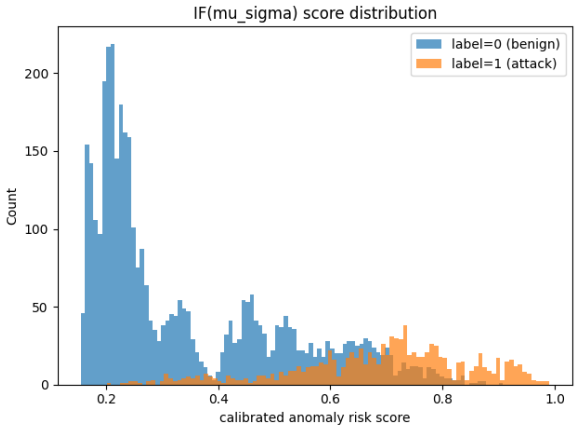

Рис. 16. Розподіл каліброваних оцінок ризику моделі Isolation Forest для нормальних подій та атак (сигнал відхилення від норми після приведення до шкали [0,1]

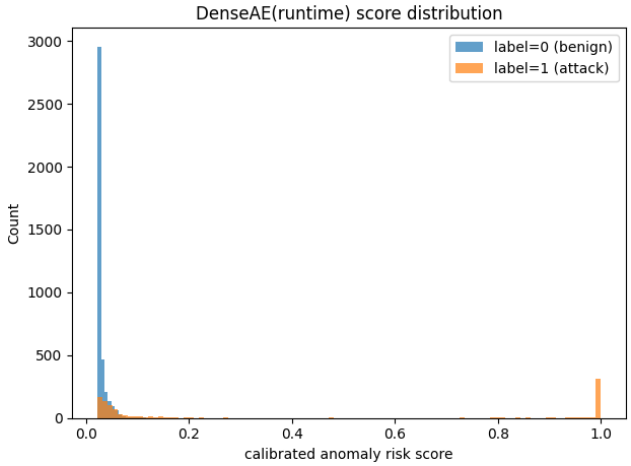

Рис. 17. Розподіл каліброваних оцінок ризику моделі Dense Autoencoder за похибкою реконструкції для нормальних подій та атак (нетиповість як зростання реконструкційної похибки, приведеної до ймовірнісної шкали)

Архітектурний контекст інфраструктурного рішення та зв'язок з конвеєром оцінювання ризику. Щоб коректно інтерпретувати результати, важливо зафіксувати, як математично описаний конвеєр реалізується як інфраструктурне рішення. На Рис. 18 подано узагальнену схему, у якій система розділена на площину обробки даних (data plane) та площину керування (control plane). Такий поділ показує, як ML- і NLP-компоненти інтегруються в єдину процедуру оцінювання ризику.

У площині обробки даних послідовність обробки однієї події відповідає формалізації, описаній у статті. HTTP/API-запит перетворюється на подію x, після чого виконується попередня обробка, яка виділяє числові ознаки x_{num} і текстові поля x_{text}. Далі паралельно запускаються ML-інференс за x_{num} та NLP-інференс за x_{text}. На наступному кроці модуль fusion формує фінальну оцінку ризику $p_{final}(x)$ і рівень ризику з урахуванням політик, порогів та механізму гістерезису. Отриманий результат передається у сервіс сповіщень та індексації, який забезпечує збереження сповіщень, інцидентів і пояснень.

Окремим компонентом у data plane є сховище стану для гістерезису. Воно зберігає попередній рівень ризику для подій, які пов'язані стабільним ключем кореляції. Це

робить рішення в потоці стійкішим і зменшує коливання рівня ризику при малих змінах оцінок біля порогу.

Площина керування відповідає за політики та перенавчання. Вона охоплює збір зворотного зв'язку у вигляді міток (TP,FP,FN,TN), оновлення навчальних наборів, перенавчання моделей і публікацію оновлених версій моделей та метаданих. Важливо, що політики, зокрема ваги, пороги та параметри обмеження підсилення додаткового каналу, передаються до fusion як керовані параметри. Це дозволяє змінювати режим роботи системи без зміни математичної постановки та без втручання в основний потік обробки подій.

Розширений опис архітектури платформи, сценаріїв розгортання та циклу перенавчання наведено на Рис. 18.

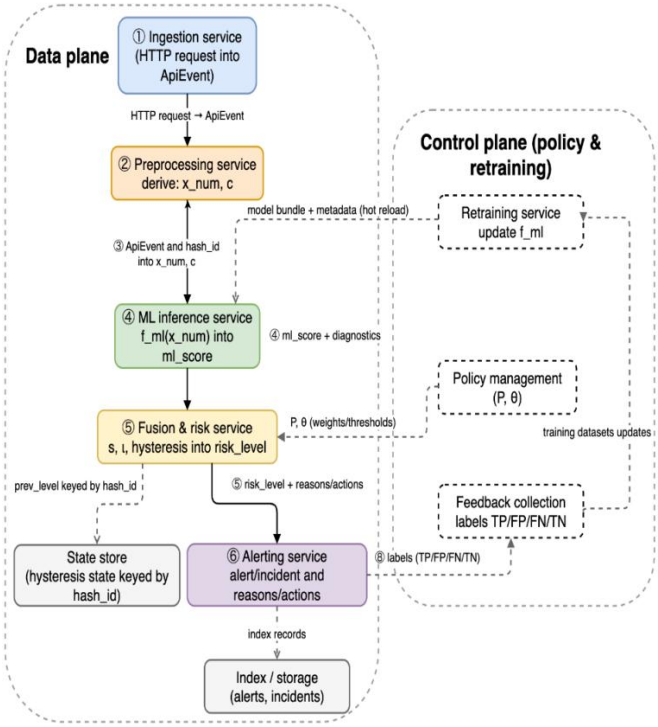

Рис. 18. Узагальнена схема інфраструктурного рішення ML конвеєра виявлення аномалій і оцінювання ризику HTTP/API-запитів (площина обробки даних і площина керування, політики та перенавчання)

Експериментальні результати: табличні метрики, порівняння моделей та підсумок по каналах. У цьому підрозділі наведено кількісні результати оцінювання компонентів ML-конвеєра за табличними метриками якості розділення класів і контролем хибно-позитивних спрацювань. Результати подано у вигляді таблиць, щоб прозоро порівняти моделі, зафіксувати робочі значення порога τ та відповідні порогові метрики, а також показати узагальнену якість як для окремих моделей, так і для ансамблів.

Оцінювання виконано на тестовій підвибірці. Для кожної моделі або агрегованої конфігурації наведено ROC-AUC, PR-AUC, F1, Precision, Recall, FPR та поріг τ, який застосовано для розрахунку порогових метрик. У таблицях використано уніфіковані назви каналів, де керований числовий ML-канал відповідає моделям із навчанням на розмічених даних, а додатковий числовий ML-канал нетипової поведінки відповідає моделям відхилення від норми.

Таблиця 1

Керований числовий ML канал, порівняння моделей

Model	Data	ROC-AUC	PR-AUC	F1	Precision	Recall	FPR	τ
Logistic Regression	val	0.864327	0.697933	0.523077	0.680000	0.425000	0.050000	0.201401
Logistic Regression	test	0.859940	0.699185	0.528926	0.681926	0.432000	0.050375	0.201401
HistGradientBoosting	val	0.983710	0.949120	0.832685	0.810606	0.856000	0.050000	0.104380
HistGradientBoosting	test	0.983808	0.949651	0.839691	0.811858	0.869500	0.050375	0.104380
Random-Forest	val	0.984379	0.949944	0.836399	0.810507	0.864000	0.050500	0.123049
Random-Forest	test	0.983483	0.948084	0.838835	0.815094	0.864000	0.049000	0.123049
ExtraTrees	val	0.984868	0.950834	0.842004	0.812268	0.874000	0.050500	0.288658
ExtraTrees	test	0.983945	0.950816	0.839255	0.812354	0.868000	0.050125	0.288658

Інтерпретація Таблиці 1 є такою. Деревні ансамблі HistGradientBoosting, RandomForest та ExtraTrees демонструють високі значення ROC-AUC і PR-AUC та водночас забезпечують стабільні значення F1 при вибраному порозі. Logistic Regression є базовою лінією, і її показники є нижчими, що узгоджується з тим, що лінійна модель гірше відтворює нелінійні взаємодії ознак.

Таблиця 2

Додатковий числовий ML канал нетиповості, порівняння моделей

Model	Data	ROC-AUC	PR-AUC	F1	Precision	Recall	FPR	τ
Isolation Forest	val	0.914177	0.741524	0.643301	0.739922	0.569000	0.050000	0.185869
Isolation Forest	test	0.914007	0.749649	0.638655	0.726115	0.570000	0.053750	0.185869
Dense Autoencoder	val	0.897623	0.759708	0.603027	0.721448	0.518000	0.050000	0.058022
Dense Autoencoder	test	0.889780	0.754840	0.613878	0.723693	0.533000	0.050875	0.058022
LSTM-AE	val	0.853478	0.659164	0.475222	0.651568	0.374000	0.050000	0.162444
LSTM-AE	test	0.849040	0.664194	0.491732	0.653942	0.394000	0.052125	0.162444
One-Class SVM	val	0.740995	0.616030	0.528510	0.683043	0.431000	0.050000	0.071144
One-Class SVM	test	0.750011	0.634873	0.551786	0.690458	0.459500	0.051500	0.071144

Найстабільніші інтегральні метрики у Таблиці 2 демонструють Isolation Forest та Dense Autoencoder. LSTM-AE та One-Class SVM мають нижчі значення ROC-AUC та

PR-AUC, тобто слабше ранжування ризику, однак вони можуть використовуватися як допоміжний сигнал нетиповості.

Таблиця 3

Ансамблі числових моделей, основний та додатковий канали

Ensemble	Data	ROC-AUC	PR-AUC	F1	Precision	Recall	FPR	τ
Primary ML ensemble	val	0.984637	0.950977	0.834951	0.811321	0.860000	0.050000	0.038451
Primary ML ensemble	test	0.984325	0.951080	0.839952	0.809745	0.872500	0.051250	0.038451
Additional anomaly ML ensemble	val	0.922141	0.779491	0.619127	0.729275	0.536402	0.052036	0.050129
Additional anomaly ML ensemble	test	0.918200	0.774583	0.610599	0.720109	0.530000	0.051500	0.049082

У Таблиці 3, основний ML-ансамбль відтворює рівень найкращих керованих моделей і демонструє стабільні порогові метрики, отже він є придатною узагальненою основою числового каналу. Додатковий ML-ансамбль нетипової поведінки формує окремий сигнал для рідкісних або нових режимів. За інтегральними метриками він є слабшим за основний ансамбль, що є очікуваним для каналу відхилення від норми.

Підсумок по каналах за Таблицями 1-3 є таким. Керований числовий ML-канал забезпечує високу якість ранжування ризику та стабільні порогові метрики, тоді як Logistic Regression як базова лінія помітно поступається. Додатковий числовий ML-канал нетипової поведінки у середньому слабший за керовані деревні ансамблі, однак формує практично корисний сигнал відхилення від норми. Основний ML-ансамбль відтворює рівень найкращих керованих моделей і демонструє стабільні значення F1 та Recall на тесті. Додатковий ML-ансамбль нетипової поведінки формує більш обережний канал із помірною повнотою при контрольованому FPR.

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

У роботі розроблено та експериментально перевірено ML-орієнтований підхід до виявлення атак в API/HTTP-трафіку, який поєднує керований числовий ML-канал та додатковий канал виявлення нетипової поведінки. Методична і практична перевага підходу полягає в тому, що різнорідні сирі виходи моделей приводяться до спільної ймовірнісної шкали ризику $p(x)\in[0,1]$, а рівень хибно-позитивних спрацювань контролюється керованою політикою вибору порогу.

Керований числовий ML-канал на базі деревних ансамблів забезпечив найкращу базову якість серед числових моделей. Для ExtraTrees на тестовій підвибірці отримано ROC-AUC 0.983945 та F1 0.839255, тоді як базова лінійна модель Logistic Regression має F1 0.528926. Додатково показано, що ансамблювання в основній групі стабілізує результат без втрати якості. Для основного ML-ансамблю на тесті отримано ROC-AUC 0.984325 та F1 0.839952, що підтверджує доцільність узагальнення сигналів кількох керованих моделей у єдину групову оцінку.

Канал виявлення нетипової поведінки формує допоміжний сигнал відхилення від норми, який є корисним для рідкісних або нових режимів. У цій групі найстабільніші результати демонструють Isolation Forest і Dense Autoencoder. Водночас за

інтегральними метриками цей канал є слабшим за керований канал, що є очікуваним для підходів, орієнтованих на відхилення від норми, а не на максимальне розділення класів у розмічених даних. Практична роль такого сигналу полягає у підсиленні реакції на незвичні події там, де керовані моделі можуть бути менш чутливими через обмежене представлення подібних сценаріїв у навчальних даних.

Формалізовано повний конвеєр оцінювання ризику для подій API/HTTP. Конвеєр охоплює отримання сирих оцінок, калібрування до єдиної шкали $p(x)\in[0,1]$, агрегацію в межах груп та вибір робочої точки через поріг τ. Така постановка робить методику відтворюваною, придатною для порівняння моделей і конфігурацій, а також прозорою з точки зору налаштування експлуатаційних політик.

Показано, що калібрування є ключовим елементом методології, оскільки забезпечує порівнюваність оцінок моделей різної природи у спільній шкалі ризику. Температура T є керованим параметром, який узгоджує різкість ймовірностей і підвищує стабільність оцінок у прикладному застосуванні, зокрема за дисбалансу класів і коливань фону трафіку.

Запропоновано формальний вибір робочої точки на основі бюджету хибно-позитивних спрацювань FPR. Такий підхід є критичним для систем безпеки у потоковому режимі, оскільки дозволяє підтримувати заданий рівень хибних сповіщень про небезпеку і відповідне навантаження на первинний аналіз, не зводячи налаштування системи лише до максимізації окремих метрик.

У підсумку отримані результати підтверджують ефективність двоканальної ML-побудови. Керований числовий ML-канал забезпечує сильну основу для розділення нормальні події або атаки, а канал нетипової поведінки доповнює систему сигналом новизни та рідкісних режимів. Уніфікація оцінок у шкалу $p(x)\in[0,1]$ і керований вибір порогу роблять систему придатною до практичного використання з контрольованим рівнем хибно-позитивних спрацювань.

Перспективи подальших досліджень пов’язані з розширенням як методичної, так і практичної складової підходу. Доцільним є масштабування експериментів на більш різноманітні набори трафіку та сценарії використання, включно з різними типами API, варіативними профілями навантаження та складнішими класами атак. Окремий напрям полягає у розвитку механізмів об’єднання та налаштування параметрів конвеєра під контекст події і зміну фону трафіку, зокрема для підвищення стійкості до дрейфу даних. Перспективним є поглиблення інтерпретованості рішень на рівні внеску ознак і груп моделей, щоб первинний аналіз міг швидше встановлювати причини спрацювання. Для каналу нетипової поведінки доцільним є розширення набору ознак, застосування більш стійких архітектур моделей автоенкодерів та поєднання з напів-керованими підходами для кращого виявлення нових шаблонів атак. Практична складова розвитку включає інфраструктурний контур зворотного зв’язку і перенавчання, накопичення розмічених інцидентів та автоматизоване оновлення моделей і політик у безперервному режимі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Abibulaiev, A., Pukach, P., & Vovk, M. (2026). Context-aware ML/NLP pipeline for real-time anomaly detection and risk assessment in cloud API traffic. *Machine Learning and Knowledge Extraction, 8*(1), 25. https://doi.org/10.3390/make8010025
2. Aldawsari, H., & Kouchay, S. A. (2024). Integrating AI and machine learning algorithms in cloud security frameworks for enhanced proactive threat detection and mitigation. *Journal of Engineering and Technology Management, 74*, 1042–1058.

3. Alzoubi, Y. I., Mishra, A., & Topcu, A. E. (2024). Research trends in deep learning and machine learning for cloud computing security. *Artificial Intelligence Review, 57*, 132–176. https://doi.org/10.1007/s10462-024-10776-5
4. Belal, M. M., & Sundaram, D. M. (2022). Comprehensive review on intelligent security defences in cloud: Taxonomy, security issues, ML/DL techniques, challenges and future trends. *Journal of King Saud University – Computer and Information Sciences, 34*, 9102–9131. https://doi.org/10.1016/j.jksuci.2022.08.035
5. Chornii, V., Martseniuk, Y., Partyka, A., & Harasymchuk, O. (2025). Information security risks associated with the uncontrolled storage of secrets in source code. *CEUR Workshop Proceedings, 4042*, 250–271.
6. Gooden, G. (n.d.). *AWS prescriptive guidance: Embracing zero trust—A strategy for secure and agile business transformation*. https://docs.aws.amazon.com/prescriptive-guidance/latest/strategy-zero-trust-architecture/introduction.html
7. Komala, R., Arun Kumar, B. R., Mahadeshwara, P., & Shreyas, A. (2024). Smart governance among smart cities for legal consideration to international data migration in cloud using machine learning, NLP and blockchain smart contract. *Preprints*. https://doi.org/10.20944/preprints202408.1028.v1
8. Malaiyappan, J. N. A., Prakash, S., Bayani, S. V., & Devan, M. (2024). Enhancing cloud compliance: A machine learning approach. *Advanced International Journal of Multidisciplinary Research, 2*(2). https://doi.org/10.62127/aijmr.2024.v02i02.1036
9. Mamidi, S. R. (2024). The role of AI and machine learning in enhancing cloud security. *Journal of Artificial Intelligence and General Science, 3*, 403–417. https://doi.org/10.60087/jaigs.v3i1.161
10. Okare, B. P., Omolayo, O., & Aduloju, T. D. (2024). Designing unified compliance intelligence models for scalable risk detection and prevention in SME financial platforms. *International Journal of Multidisciplinary Research and Growth Evaluation, 5*, 1421–1433. https://doi.org/10.54660/IJMRGE.2024.5.4.1421-1433
11. Olabanji, S. O., Marquis, Y. A., Adigwe, C. S., Ajayi, S. A., Oladoyinbo, T. O., & Olaniyi, O. O. (2024). AI-driven cloud security: Examining the impact of user behavior analysis on threat detection. *Asian Journal of Research in Computer Science, 17*, 57–74. https://doi.org/10.9734/AJRCOS/2024/v17i3424
12. Pham, V. H., & Do, T. T. H. (2023). Enhancing web application security: A deep learning and NLP-based approach for accurate attack detection. *Journal of Science, Technology and Information Security, 3*, 77–90.
13. Piskozub, A., & Abibulaiev, A. (2025). Integration of NLP and ML in cloud infrastructure security. *CEUR Workshop Proceedings, 4024*, 260–275.
14. Pop, D. (2012). Machine learning and cloud computing: Survey of distributed and SaaS solutions. *IEAT Technical Report*. https://arxiv.org/abs/1603.08767
15. Qayyum, A., Ijaz, A., Usama, M., Iqbal, W., Qadir, J., Elkhatib, Y., & Al-Fuqaha, A. (2024). Securing machine learning in the cloud: A systematic review of cloud machine learning security. *Frontiers in Big Data, 3*, 587139. https://doi.org/10.3389/fdata.2020.587139
16. Rakgoale, D. M., Kobo, H. I., Mapundu, Z. Z., & Khosa, T. N. (2024). A review of AI/ML algorithms for security enhancement in cloud computing with emphasis on artificial neural networks. In *Proceedings of the 4th International Multidisciplinary Information Technology and Engineering Conference (IMITEC 2024)* (pp. 329–336). IEEE. https://doi.org/10.1109/IMITEC60221.2024.10851076
17. Reddy, A. R. P., & Reddy, A. K. (2020). Automating incident response: AI-driven approaches to cloud security incident management. *Chelonian Conservation and Biology, 15*(2).
18. Vashishth, T. K., Sharma, V., Kumar, B., & Panwar, R. (2024). Enhancing cloud security: The role of artificial intelligence and machine learning. In *Handbook of research on AI and ML in cybersecurity*. IGI Global. https://doi.org/10.4018/979-8-3693-1431-9.ch004

КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 4 (32), 2026
ISSN 2663 – 4023

Aziz Abibulaiev
Postgraduate Student of Information Protection Department
Lviv Polytechnic National University, Lviv, Ukraine
ORCID: 0009-0004-2875-5154
aziz.r.abibulaiev@lpnu.ua

Andrian Piskozub
PhD, Associate Professor at the Information Protection Department
Lviv Polytechnic National University, Lviv, Ukraine
ORCID: 0000-0002-3582-2835
andriian.z.piskozub@lpnu.ua

Edem Atamuratov
Postgraduate Student of Applied Mathematics Department
Lviv Polytechnic National University, Lviv, Ukraine
ORCID: 0009-0006-2917-9448
edem.atamuratov.asp.2025@lpnu.ua

RISK CRITERIA AND ML ALGORITHMS FOR THREAT DETECTION IN A CLOUD ENVIRONMENT

Abstract. This paper proposes and experimentally validates an ML-oriented pipeline for detecting hazardous events in the API/HTTP traffic of cloud services. The approach combines two numerical channels: (i) a supervised channel based on structured event features for stable separation between benign events and attacks, and (ii) an auxiliary atypical-behavior channel that strengthens the response to rare or novel scenarios that are weakly represented in labeled data. The key methodological idea is to unify heterogeneous model outputs into a common probabilistic risk scale via calibration, temperature scaling, and prior attack-rate adjustment, which enables comparable scores across models of different nature. To achieve controlled management of false-positive activations, the decision threshold is selected under an FPR budget, while group-level score stability is improved through ensembling, including averaging in the additive log-odds domain. After combining channel signals, the final decision is stabilized with operational policies (including hysteresis) to avoid frequent state switching in streaming mode. Model quality is verified both with tabular metrics and visually by analyzing risk-score distributions for benign events and attacks, which helps interpret overlap regions and the impact of the threshold on false alerts. Experiments on a test set show high performance of the supervised channel: for the primary ML ensemble, ROC-AUC = 0.9843, PR-AUC = 0.9511, and F1 = 0.8400 are achieved at an FPR of approximately 0.051, whereas the baseline linear model yields substantially lower F1 values. The auxiliary atypical-behavior channel provides a practically useful signal that complements the supervised channel while maintaining a controlled false-alert rate. The proposed formulation scales to other API types and load profiles because it separates policies (thresholds, weights, calibration parameters) from the main event-processing flow. The results confirm the suitability of the approach for integration into cloud monitoring and incident-response infrastructure with controllable threshold policies and model updates.

Keywords: machine learning; risk scoring; API/HTTP security; anomaly detection; OWASP Top 10; risk-based alerting; cloud microservices.

REFERENCES

1. Abibulaiev, A., Pukach, P., & Vovk, M. (2026). Context-aware ML/NLP pipeline for real-time anomaly detection and risk assessment in cloud API traffic. *Machine Learning and Knowledge Extraction, 8*(1), 25. https://doi.org/10.3390/make8010025

2. Aldawsari, H., & Kouchay, S. A. (2024). Integrating AI and machine learning algorithms in cloud security frameworks for enhanced proactive threat detection and mitigation. *Journal of Engineering and Technology Management*, *74*, 1042–1058.
3. Alzoubi, Y. I., Mishra, A., & Topcu, A. E. (2024). Research trends in deep learning and machine learning for cloud computing security. *Artificial Intelligence Review*, *57*, 132–176. https://doi.org/10.1007/s10462-024-10776-5
4. Belal, M. M., & Sundaram, D. M. (2022). Comprehensive review on intelligent security defences in cloud: Taxonomy, security issues, ML/DL techniques, challenges and future trends. *Journal of King Saud University – Computer and Information Sciences*, *34*, 9102–9131. https://doi.org/10.1016/j.jksuci.2022.08.035
5. Chornii, V., Martseniuk, Y., Partyka, A., & Harasymchuk, O. (2025). Information security risks associated with the uncontrolled storage of secrets in source code. *CEUR Workshop Proceedings, 4042*, 250–271.
6. Gooden, G. (n.d.). *AWS prescriptive guidance: Embracing zero trust—A strategy for secure and agile business transformation*. https://docs.aws.amazon.com/prescriptive-guidance/latest/strategy-zero-trust-architecture/introduction.html
7. Komala, R., Arun Kumar, B. R., Mahadeshwara, P., & Shreyas, A. (2024). Smart governance among smart cities for legal consideration to international data migration in cloud using machine learning, NLP and blockchain smart contract. *Preprints*. https://doi.org/10.20944/preprints202408.1028.v1
8. Malaiyappan, J. N. A., Prakash, S., Bayani, S. V., & Devan, M. (2024). Enhancing cloud compliance: A machine learning approach. *Advanced International Journal of Multidisciplinary Research*, *2*(2). https://doi.org/10.62127/aijmr.2024.v02i02.1036
9. Mamidi, S. R. (2024). The role of AI and machine learning in enhancing cloud security. *Journal of Artificial Intelligence and General Science*, *3*, 403–417. https://doi.org/10.60087/jaigs.v3i1.161
10. Okare, B. P., Omolayo, O., & Aduloju, T. D. (2024). Designing unified compliance intelligence models for scalable risk detection and prevention in SME financial platforms. *International Journal of Multidisciplinary Research and Growth Evaluation*, *5*, 1421–1433. https://doi.org/10.54660/IJMRGE.2024.5.4.1421-1433
11. Olabanji, S. O., Marquis, Y. A., Adigwe, C. S., Ajayi, S. A., Oladoyinbo, T. O., & Olaniyi, O. O. (2024). AI-driven cloud security: Examining the impact of user behavior analysis on threat detection. *Asian Journal of Research in Computer Science*, *17*, 57–74. https://doi.org/10.9734/AJRCOS/2024/v17i3424
12. Pham, V. H., & Do, T. T. H. (2023). Enhancing web application security: A deep learning and NLP-based approach for accurate attack detection. *Journal of Science, Technology and Information Security*, *3*, 77–90.
13. Piskozub, A., & Abibulaiev, A. (2025). Integration of NLP and ML in cloud infrastructure security. *CEUR Workshop Proceedings, 4024*, 260–275.
14. Pop, D. (2012). Machine learning and cloud computing: Survey of distributed and SaaS solutions. *IEAT Technical Report*. https://arxiv.org/abs/1603.08767
15. Qayyum, A., Ijaz, A., Usama, M., Iqbal, W., Qadir, J., Elkhatib, Y., & Al-Fuqaha, A. (2024). Securing machine learning in the cloud: A systematic review of cloud machine learning security. *Frontiers in Big Data*, *3*, 587139. https://doi.org/10.3389/fdata.2020.587139
16. Rakgoale, D. M., Kobo, H. I., Mapundu, Z. Z., & Khosa, T. N. (2024). A review of AI/ML algorithms for security enhancement in cloud computing with emphasis on artificial neural networks. In *Proceedings of the 4th International Multidisciplinary Information Technology and Engineering Conference (IMITEC 2024)* (pp. 329–336). IEEE. https://doi.org/10.1109/IMITEC60221.2024.10851076
17. Reddy, A. R. P., & Reddy, A. K. (2020). Automating incident response: AI-driven approaches to cloud security incident management. *Chelonian Conservation and Biology*, *15*(2).
18. Vashishth, T. K., Sharma, V., Kumar, B., & Panwar, R. (2024). Enhancing cloud security: The role of artificial intelligence and machine learning. In *Handbook of research on AI and ML in cybersecurity*. IGI Global. https://doi.org/10.4018/979-8-3693-1431-9.ch004

Отримано редакцією журналу / Received: 05.01.26
Прорецензовано / Revised: 20.02.26
Схвалено до друку / Accepted: 26.03.26