https://doi.org/10.31891/2219-9365-2026-86-46
УДК 621.391

ПРОДЕУС Максим
Хмельницький національний університет
https://orcid.org/0009-0002-2968-4648
e-mail: mprodeus99@ukr.net
НІЧЕПОРУК Андрій
Хмельницький національний університет
https://orcid.org/0000-0002-7230-9475
e-mail: andrey.nicheporuk@gmail.com
ВОЗНА Наталія
Західноукраїнський національний університет
https://orcid.org/0000-0002-8856-1720
e-mail: nvozna@ukr.net

ДВОКАНАЛЬНА ІНТЕЛЕКТУАЛЬНА МЕРЕЖЕВА СИСТЕМА ВИЯВЛЕННЯ ВТОРГНЕНЬ НА ОСНОВІ ГЛИБОКОГО НАВЧАННЯ

У статті запропоновано двоканальну інтелектуальну систему виявлення мережевих вторгнень, що базується на гібридній архітектурі глибокого навчання та механізмі активних приманок. Система розглядає задачу виявлення вторгнень як задачу класифікації з двома незалежними джерелами доказів: підсистемою аналізу мережевого трафіку та підсистемою поведінкових приманок, результати яких агрегуються у єдиний висновок про стан мережі. Підсистема аналізу трафіку реалізована як ансамбль двох паралельних каналів: нейронної мережі CNN+LSTM, що виявляє складні часові залежності у послідовностях мережевих потоків, та алгоритму Random Forest, що забезпечує стабільну класифікацію поодиноких аномалій. Фінальний вектор ймовірностей формується як зважена комбінація виходів обох каналів. Підсистема приманок генерує скалярний сигнал достовірності загрози на основі стану розгорнутих honeypot-агентів у мережі та асиметрично коригує результати класифікатора: спрацювання приманки суттєво підвищує ймовірності класів атак, тоді як її відсутність вносить лише незначний коригуючий вплив. Вхідний простір ознак формується з агрегованих даних мережевих потоків за протоколами NetFlow, sFlow та IPFIX; для відбору найінформативніших атрибутів застосовується метод рекурсивного виключення на базі Random Forest у поєднанні з аналізом кореляції Спірмена. Систему навчено та протестовано на наборі даних CIC-IDS-2017. Отримані результати показали, що пропонована гібридна система досягла точності 99 %, а також показників повноти та точності на рівні 98 %, що перевищує показники ізольованих моделей CNN+LSTM (97 %) та Random Forest (96 %). Встановлено, що інтеграція сигналу підсистеми приманок дозволяє ескалювати приховані загрози, які класифікатор трафіку початково відносить до невизначених, та суттєво знижує кількість хибно негативних спрацювань у випадках цілеспрямованих атак на внутрішні сегменти мережі.

Ключові слова: інтелектуальна мережева система виявлення вторгнень, комп'ютерні атаки, приманки, глибоке навчання.

PRODEUS Maxim, NICHEPORUK Andrii
Khmelnytskyi National University
VOZNA Nataliia
West Ukrainian National University

DUAL-CHANNEL INTELLIGENT NETWORK INTRUSION DETECTION SYSTEM BASED ON DEEP LEARNING

The article proposes a dual-channel intelligent system for detecting network intrusions based on a hybrid deep learning architecture and an active honeypot mechanism. The system treats intrusion detection as a classification problem with two independent sources of evidence: a network traffic analysis subsystem and a behavioral honeypot subsystem, whose results are aggregated into a unified conclusion about the network state. The traffic analysis subsystem is implemented as an ensemble of two parallel channels: a CNN+LSTM neural network that captures complex temporal dependencies in network flow sequences, and a Random Forest algorithm that ensures stable classification of isolated anomalies. The final probability vector is formed as a weighted combination of the outputs from both channels. The honeypot subsystem generates a scalar threat confidence signal based on the state of deployed honeypot agents in the network and asymmetrically adjusts the classifier results: a honeypot trigger significantly increases the probabilities of attack classes, while its absence introduces only a minor corrective effect. The input feature space is formed from aggregated network flow data based on the NetFlow, sFlow, and IPFIX protocols. Recursive feature elimination based on Random Forest, combined with Spearman correlation analysis, is used to select the most informative attributes. The system was trained and tested on the CIC-IDS-2017 dataset. The obtained results showed that the proposed hybrid system achieved an accuracy of 99%, as well as precision and recall of 98%, outperforming the standalone CNN+LSTM (97%) and Random Forest (96%) models. It was established that integrating the honeypot subsystem signal enables the escalation of hidden threats initially classified as uncertain by the traffic classifier and significantly reduces the number of false negatives in cases of targeted attacks on internal network segments.

Keywords: intelligent network intrusion detection system, computer attacks, honeypots, deep learning.

Стаття надійшла до редакції / Received 02.04.2026
Прийнята до друку / Accepted 28.04.2026
Опубліковано / Published 31.05.2026

ПОСТАНОВКА ПРОБЛЕМИ У ЗАГАЛЬНОМУ ВИГЛЯДІ ТА ЇЇ ЗВ'ЯЗОК ІЗ ВАЖЛИВИМИ НАУКОВИМИ ЧИ ПРАКТИЧНИМИ ЗАВДАННЯМИ

Зростання інтенсивності та складності кіберзагроз у сучасних комп'ютерних мережах вимагає розробки ефективних автоматизованих систем виявлення вторгнень, здатних виявляти як відомі, так і нові типи атак у режимі реального часу. Класичні підходи, засновані на сигнатурному аналізі та традиційних алгоритмах машинного навчання, демонструють суттєві обмеження при роботі з багатоетапними та замаскованими атаками, де мережевий профіль зловмисного трафіку максимально наближений до легітимного. Це обумовлює необхідність розробки гібридних архітектур, що поєднують просторовий та часовий аналіз мережевих потоків із механізмами додаткового підтвердження загроз, зокрема на основі підсистем активних приманок.

На сьогоднішній день у науковій літературі приділяється значена увага цій проблемі, шляхи вирішення якої реалізуються як із застосуванням класичних методів машинного навчання, так і архітектур глибокого навчання. У роботі [1] для побудови легковагової системи виявлення вторгнень застосовано навчання з учителем із відбором ознак на основі інформаційного приросту, а верифікація на UNSW-NB15 підтвердила придатність підходу до роботи в реальному часі. Автори [2] порівняли п'ять алгоритмів машинного навчання і встановили, що Random Forest досягає 90,2 % при бінарній класифікації, проте лише 70,8 % при мультикласовій, що вказує на принципові обмеження класичних підходів. Для подолання цих обмежень у [3] запропоновано послідовну DNN з відбором ознак через Extra Tree Classifier, яка скоротила розмірність із 43 до 8 атрибутів і досягла точності 97,93 %. Принципово відмінний підхід реалізовано у [4], де трафік перетворювався на RGB-зображення для класифікації архітектурою Vision Transformer із точністю до 100 % на наборі Edge-IIoTset. Огляд [5] систематизував еволюцію систем від сигнатурних методів до інтеграції LLM, зафіксувавши, що жоден із підходів наразі не забезпечує повноцінної надійності для промислового розгортання. Таким чином, попри значний прогрес, залишаються невирішеними проблеми масштабування на реальне середовище, балансу точності й обчислювальних витрат та інтерпретованості складних моделей.

Двоканальна інтелектуальна мережева система виявлення вторгнень на основі глибокого навчання

Запропонована система виявлення мережевих вторгнень розглядається як задача класифікації з двома незалежними джерелами доказів – підсистемою аналізу трафіку та підсистемою поведінкових приманок, результати яких агрегуються у єдиний висновок про стан мережі.

Нехай задано мережу, що описується скінченною множиною вузлів $N = \{n_1, n_2, .., n_N\}$, де кожен n_i представляє окремий хост, сервер або мережевий пристрій. Мережевий трафік спостерігається на рівні потоків, таким чином, що мережевий потік f визначається кортежом $f = (src, dst, p_{src}, p_{dst}, \pi, \tau_s, \tau_e)$, де $src, dst \in N$ визначають адреси джерела та призначення, $p_{src}, p_{dst} \in \mathbb{N}$ мережеві порти, $\pi \in P = \{TCP, UDP, ICMP, \dots\}$ – транспортний протокол, а $\tau_s < \tau_e \in \mathbb{R}_+$ – часові межі потоку. Множина класів трафіку визначається як $\mathcal{A} = \{a_0\} \cup \mathcal{A}_{atk}$, де a_0 – позначає нормальний трафік, а $\mathcal{A}_{atk} = \{DoS, Probe, \dots\}$ – множина класів атак.

Задамо функцію вилучення ознак $\phi: \mathcal{F} \to \mathbb{R}^d$, яка перетворює сирий потік f у числовий вектор ознак $x = \phi(f) \in \mathbb{R}^d$, де d – кількість ознак. Після попередньої обробки — нормалізації, видалення надлишкових атрибутів та зменшення розмірності — отримується очищений вектор $x' \in \mathbb{R}^{d'}$, де $d' < d$. Навчальна вибірка має вигляд $\mathcal{D} = \{(x'_i, y_i)\}_{i=1}^{M}$, де $y_i \in \mathcal{A}$ – мітка класу.

Підсистема аналізу трафіку на основі очищеного вектора ознак x' формує вектор ймовірностей класів $p_{net}(\tau) = (p_{net}(a_0|\tau), p_{net}(a_1|\tau), \dots) \in [0,1]^{|\mathcal{A}|}$. Розрахунок цього вектора базується на ансамблюванні двох паралельних обчислювальних каналів:

1) канал глибокого навчання (CNN+LSTM): аналізує послідовність потоків для виявлення складних часових залежностей, генеруючи вектор ймовірностей $p_{DL} \in [0,1]^{|\mathcal{A}|}$;

2) канал статистичного аналізу (Random Forest): здійснює класифікацію поточного вектора незалежно від контексту для ідентифікації поодиноких аномалій, генеруючи вектор $p_{RF} \in [0,1]^{|\mathcal{A}|}$.

Фінальний вектор підсистеми аналізу трафіку обчислюється як зважена комбінація виходів обох каналів:

$$\mathbf{p}_{net}(\tau) = \alpha \cdot \boldsymbol{p}_{DL}(\tau) + (1 - \alpha) \cdot \boldsymbol{p}_{RF}(\tau) \quad (1)$$

де ваговий коефіцієнт $\alpha \in (0.5,1)$ підбирається на валідаційній вибірці та відображає більший вплив CNN+LSTM порівняно з RF при збереженні стабілізуючого внеску лісу. CNN+LSTM є сильнішим детектором для складних паттернів – slow-and-low атак, поступових відхилень, послідовних проб – тоді як RF краще справляється з поодинокими аномальними потоками, характерними для port scanning та brute force, де часовий контекст менш важливий. Саме ця комплементарність двох підходів обумовлює вибір м'якого голосування замість жорсткого: усереднення ймовірностей зберігає відтінки невпевненості обох моделей, тоді як жорстке голосування зводить їх до бінарних передбачень, втрачаючи інформацію про ступінь впевненості. Таким

чином отриманий вектор відображає впевненість системи у приналежності трафіку до певного класу атак виключно на основі аналізу мережевих пакетів.

Паралельно з підсистемою аналізу трафіку функціонує підсистема приманок. Нехай у мережі розгорнуто K приманок, що утворюють множину $\mathcal{H} = \{h_1, h_2, \dots, h_K\}$, де кожна приманка h_k характеризується своїм розташуванням в мережевій топології (зона периметру, DMZ, внутрішня зона) та типом (мережевий сервіс або файловий артефакт). У момент часу τ стан підсистеми приманок описується бінарним вектором активації:

$$\mathbf{b}(\tau) = \big(b_1(\tau), b_2(\tau), \dots, b_K(\tau)\big) \in \{0,1\}^K \tag{2}$$

де $b_K(\tau) = 1$ означає, що приманка h_k зафіксувала тригерну подію в межах часового вікна $[\tau - \Delta\tau, \tau]$ і $b_K(\tau) = 0$ в протилежному випадку. Значення $\Delta\tau$ визначається емпірично як характерний часовий масштаб атак, що підлягають виявленню (припускатимемо значення в діапазоні від 30 до 120 сек).

Підсистема приманок формує скалярний сигнал достовірності загрози:

$$\mathrm{s}_{hp}(\tau) = \frac{1}{K}\sum_{k=1}^{K} w_k \cdot b_k(\tau), \tag{3}$$

де $w_k \in \mathbb{R}_+$ визначає ваговий коефіцієнт k-ї приманки, що відображає її інформативність залежно від розташування та типу; приманки у внутрішніх сегментах мережі, менш доступних для зовнішнього сканування, отримують вищу вагу, оскільки будь-яке їх спрацювання є сильнішим свідченням проникнення.

Фінальний висновок системи формується як зважена комбінація двох сигналів. Введемо функцію коригування : $[0,1]^{|\mathcal{A}|} \times [0,1] \to [0,1]^{|\mathcal{A}|}$, яка застосовує до вектора p_{net} асиметричне коригування залежно від значення s_{hp}:

$$\mathrm{p}_{final}(c|\tau) = p_{net}(c|\tau) \cdot \Big(1 + \beta \cdot s_{hp}(\tau)\Big), \text{для } c \in \mathcal{A}_{atk} \tag{4}$$

$$\mathrm{p}_{final}(a_0|\tau) = p_{net}(a_0|\tau) \cdot \Big(1 - \gamma \cdot s_{hp}(\tau)\Big), \text{для } c \in \mathcal{A}_{atk} \tag{5}$$

де $\beta > 0$ – коефіцієнт підсилення при наявності тригера приманки, а $\gamma \in (0,1)$ – коефіцієнт послаблення нормального класу, причому $\beta \gg \gamma$, що відображає асиметричний характер впливу: спрацювання приманки є сильним свідченням атаки та суттєво підвищує ймовірності класів загроз, тоді як відсутність спрацювання є лише слабким аргументом на користь нормального стану і вносить незначний штраф.

Після нормалізації p_{final} на одиницю остаточний клас визначається як:

$$\hat{y}(\tau) = arg\max_{c \in \mathcal{A}} \mathrm{p}_{final}(c|\tau) \tag{6}$$

Таким чином процес функціонування інтелектуальної мережевої системи виявлення вторгнень на основі глибокого навчання можна подати кроками:

1. Збір мережевого трафіку з усіх сегментів мережі через протоколи NetFlow, sFlow та IPFIX і передача сирих потоків до колектора.
2. Вилучення вектора ознак $x = \phi(f) \in \mathbb{R}^d$ з кожного потоку та попередня обробка: видалення інфраструктурно-специфічних атрибутів, відбір d' найінформативніших ознак через RFE, нормалізація до діапазону [0,1].
3. Паралельна класифікація обробленого вектора x' двома незалежними моделями: CNN+LSTM отримує послідовність з T останніх потоків і формує вектор ймовірностей $\boldsymbol{p}_{DL}$; Random Forest аналізує x' незалежно і формує $\boldsymbol{p}_{RF}$.
4. Ансамблеве об'єднання результатів через зважене м'яке голосування відповідно до виразу 1
5. Паралельно з кроками 3-4 підсистема приманок формує бінарний вектор активації (вираз 2) на основі тригерних подій, зафіксованих honeypot-агентами у часовому вікні $[\tau - \Delta\tau, \tau]$ та обчислює скалярний сигнал s_{hp} відповідно до виразу 3.
6. Асиметричне коригування вектора p_{net} сигналом s_{hp}: ймовірності класів атак підсилюються з коефіцієнтом β, ймовірність нормального класу послаблюється з коефіцієнтом γ << β, після чого вектор нормалізується до p_{final} (вирази 4-5);
7. Визначення фінального класу ŷ відповідно до виразу 6.

Архітектура та функціонування підсистеми CNN+LSTM + RF

Вхідний простір ознак системи формується на основі агрегованих даних мережевих потоків, що збираються за протоколами NetFlow, sFlow або IPFIX. Кожен потік описується початковим вектором $x \in \mathbb{R}^d$, структура якого охоплює чотири семантичні групи: часові характеристики (тривалість, міжпакетні інтервали, показники jitter), об'ємні характеристики (кількість та розмір пакетів, інтенсивність трафіку); протокольні

ознаки (TCP-прапорці, типи протоколів, стани з'єднань) та агреговані поведінкові характеристики, що обчислюються у ковзному вікні (кількість з'єднань до конкретних сервісів чи адрес призначення).

Підсистема глибокого навчання реалізована як послідовна архітектура, де конволюційні шари виступають екстрактором просторових представлень, а рекурентний шар моделює часову динаміку потоків.

На вході CNN отримує вектор ознак $x' \in \mathbb{R}^{d'}$, інтерпретований як одновимірний сигнал, де ознаки розташовані у семантично впорядкованій послідовності: часові ознаки, потім об'ємні, потім прапорцеві. Таке впорядкування не є випадковим — воно забезпечує, що конволюційні фільтри застосовуються до семантично суміжних ознак, між якими існує природна кореляція. Перший конволюційний шар Conv1D з 64 фільтрами розміру 3 та активацією ReLU виявляє прості локальні паттерни: наприклад, аномально великий середній розмір пакету у поєднанні з малим міжпакетним інтервалом та підвищеним SYN-прапорцем є характерною ознакою SYN-flood атаки, яку один фільтр здатен виявити як єдиний структурний елемент. Операція MaxPooling зменшує розмірність вдвічі та забезпечує інваріантність до незначних зсувів у позиції ознак. Другий Conv1D шар зі 128 фільтрами будує ієрархічні комбінації з первинних паттернів, виявляючи складніші взаємодії між групами ознак. Після flatten-операції отримується вектор просторового представлення: $h_{CNN} \in \mathbb{R}^{n_{CNN}}$.

Для моделювання часової динаміки потоків CNN+LSTM отримує на вхід не окремий потік, а послідовність $X_{seq} = \{x'_{t-T+1}, \dots, x'_t\}$ з T послідовних потоків. Вихід конволюційних шарів після обробки кожного кроку послідовності передається в LSTM-шар зі 128 прихованими станами. LSTM реалізує гейтовий механізм управління пам'яттю: вентиль забування: $f_t = \sigma\big(W_f \cdot [h_{t-1}; x_t] + b_f\big)$ визначає, яка частина попереднього контексту залишається актуальною, вентиль входу $i_t = \sigma(W_i \cdot [h_{t-1}; x_t] + b_i)$ контролює надходження нової інформації, а вентиль виходу $o_t = \sigma(W_o \cdot [h_{t-1}; x_t] + b_o)$ формує поточний прихований стан $h_t = o_t \odot \tanh(c_t)$, де стан комірки $c_t = f_t \odot c_{t-1} + i_t \odot g_t$ накопичує довгострокову пам'ять про еволюцію трафіку.

Вихідний прихований стан $h_t \in \mathbb{R}^{128}$ після останнього кроку послідовності передається до повнозв'язного Dense-шару з активацією softmax, що формує вектор ймовірностей класів $p_{DL} \in [0,1]^{|\mathcal{A}|}$ зі сумою компонент, рівною одиниці.

Навчання моделі виконується наскрізно методом зворотного поширення помилки через час (BPTT) для LSTM та стандартним backpropagation для CNN-шарів. Функція втрат є категоріальною крос-ентропією з L2-регуляризацією:

$$\mathcal{L}(\Theta) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c} \cdot \log\Big(\hat{y}_{i,c}(\Theta)\Big) + \lambda \cdot \|\Theta\|^2 \qquad (7)$$

де перший доданок штрафує невірні передбачення, а $\lambda \cdot \|\Theta\|^2$ обмежує норму ваг для запобігання перенавчанню. Оптимізація виконується алгоритмом Adam з параметрами $\beta_1 = 0.9$, $\beta_2 = 0.999$ та початковою швидкістю навчання $\eta = 10^{-3}$.

Оцінка ефективності. Для оцінки ефективності запропонованої архітектури було проведено серію експериментів на базі набору даних CIC-IDS-2017, який містить актуальні сценарії мережевих атак та відображає реалістичний розподіл трафіку. Процес підготовки даних включав фільтрацію ознак методом Recursive Feature Elimination та нормалізацію Min-Max, після чого вибірка була розділена у пропорції 80/20 для навчання та тестування відповідно. Основний фокус експериментальної оцінки зосереджено на підсистемі аналізу трафіку, тобто на ансамблі CNN+LSTM та Random Forest, оскільки саме вона несе головне класифікаційне навантаження і функціонує безперервно незалежно від стану приманок.

Підсистема приманок у даному експерименті реалізована у вигляді змодельованого сигналу, оскільки вихідний датасет CIC-IDS-2017 [6] не містить інформації про взаємодію з honeypot-артефактами. Для відтворення поведінки реальної підсистеми критичним внутрішнім вузлам мережі були призначені віртуальні honeypot-ідентифікатори, а тригери активації генерувалися детерміновано: вектор $\mathrm{b}(\tau)$ встановлювався у значення 1 для відповідних приманок синхронно з початком задокументованих інцидентів типів DoS та PortScan, що присутні у розмічених записах датасету. Такий підхід дозволяв оцінити механізм коригування та асиметричний вплив сигналу s_{hp} на фінальний вектор ймовірностей, однак не претендує на повноцінну симуляцію розподіленої honeypot-інфраструктури. Верифікація підсистеми приманок у реалістичних умовах є предметом окремого дослідження та потребує розгортання у живому мережевому середовищі.

Для функціонування механізму коригування було встановлено такі гіперпараметри: коефіцієнт підсилення $\beta = 1.5$, коефіцієнт послаблення $\gamma = 0.05$ та ваговий коефіцієнт ансамблю $\alpha = 0.7$. Значне перевищення β над γ реалізує задану асиметрію впливу: спрацювання приманки суттєво підвищує ймовірності класів атак у векторі p_{final}, тоді як її відсутність лише незначно коригує ймовірність нормального стану, не скасовуючи ескалацію у випадках, коли підсистема аналізу трафіку вже зафіксувала аномалію.

В межах порівняльного аналізу було протестовано три конфігурації: ізольована модель Random Forest як базовий рівень, ізольована нейронна мережа CNN+LSTM та запропонований гібридний ансамбль

CNN+LSTM+RF з інтегрованим сигналом підсистеми приманок (таблиця 1). Практична реалізація підсистеми CNN+LSTM виконана засобами бібліотеки TensorFlow з використанням інтерфейсу Keras. Навчання та тестування моделі проводилось на CPU Intel Core i7-10700 з 32 ГБ оперативної пам'яті без залучення графічного прискорювача, при розмірі батчу 256 та кількості епох до 50 з ранньою зупинкою за метрикою валідаційних втрат.

Таблиця 1

Результати експериментів

Модель	Accuracy	Precision	Recall	F1-Score
RF	0,96	0,95	0,94	0,94
CNN + LSTM	0,97	0,97	0,96	0,96
Пропонована система	0,99	0,98	0,98	0,98

Аналіз отриманих даних підтверджує гіпотезу про те, що об'єднання просторового аналізу ознак через CNN та моделювання часових залежностей через LSTM дозволяє ефективніше ідентифікувати складні багатоетапні атаки порівняно з класичними методами машинного навчання. Додавання Random Forest як паралельної гілки забезпечило стабілізацію прогнозів на поодиноких аномальних потоках, де нейронна мережа демонструє знижену впевненість через відсутність достатнього часового контексту. Ключовим фактором покращення показників Recall та F1-Score стало впровадження сигналу s_{hp}. Завдяки високому значенню коефіцієнта β система успішно ескалювала приховані загрози, які підсистема аналізу трафіку початково класифікувала з низькою ймовірністю. Зокрема, у випадках цілеспрямованого сканування внутрішніх сегментів, де мережевий профіль атаки був максимально наближений до легітимного трафіку, саме тригер приманки дозволив скоригувати фінальний вектор p_{final} і перевести стан системи з рівня «невизначений» безпосередньо до «небезпечний». Водночас мале значення γ давало зменшення відсутність хибних негативів у випадках, коли атака відбувалася в обхід встановлених приманок.

ВИСНОВКИ З ДАНОГО ДОСЛІДЖЕННЯ І ПЕРСПЕКТИВИ ПОДАЛЬШИХ РОЗВІДОК У ДАНОМУ НАПРЯМІ

У роботі запропоновано гібридну інтелектуальну систему виявлення мережевих вторгнень, що поєднує ансамблеву архітектуру CNN+LSTM та Random Forest із механізмом асиметричного коригування на основі сигналу підсистеми активних приманок. Проведені експерименти на наборі даних CIC-IDS-2017 підтвердили, що запропонований підхід забезпечує точність класифікації на рівні 99 % та F1-score 98 %, перевищуючи показники як ізольованих нейромережевих моделей, так і класичних методів машинного навчання.

References

1. Mebawondu J. O., Alowolodu O. D., Mebawondu J. O., Adetunmbi A. O. Network intrusion detection system using supervised learning paradigm. Scientific African. 2020. Vol. 9. e00497. https://doi.org/10.1016/j.sciaf.2020.e00497

2. Clottey R. N., Yaokumah W., Appati J. K. Modelling and Evaluation of Network Intrusion Detection Systems Using Machine Learning Techniques. International Journal of Intelligent Information Technologies. 2021. Vol. 17, No. 4. P. 1–19. https://doi.org/10.4018/IJIIT.289971

3. Farhan M., Waheed ud din H., Ullah S., Hussain M. S., Khan M. A., Mazhar T., Khattak U. F., Hilali Jaghdam I. Network-based intrusion detection using deep learning technique. Scientific Reports. 2025. Vol. 15. Article 25550. https://doi.org/10.1038/s41598-025-08770-0

4. Zhou H., Zou H., Li W., Li D., Kuang Y. HiViT-IDS: An Efficient Network Intrusion Detection Method Based on Vision Transformer. Sensors. 2025. Vol. 25, No. 6. Article 1752. https://doi.org/10.3390/s25061752

5. Feng Y., Sakurai K. Network Intrusion Detection: Evolution from Conventional Approaches to LLM Collaboration and Emerging Risks. arXiv preprint. 2025. arXiv:2510.23313. https://doi.org/10.48550/arXiv.2510.23313

6. Sharafaldin I., Habibi Lashkari A., Ghorbani A. A., Toward generating a new intrusion detection dataset and intrusion traffic characterization, in: Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP 2018), SCITEPRESS, Funchal, Madeira, Portugal, 2018, pp. 108–116. doi:10.5220/0006639801080116.