АДАПТИВНІ ЗАСОБИ ЗАХИСТУ КОМП'ЮТЕРНИХ СИСТЕМ НА ОСНОВІ АПАРАТА НЕЙРОННИХ МЕРЕЖ

Трень А.С.[1], Мухін В.Є.[2]

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

[1] tren.artem@lll.kpi.ua, [2] v_mukhin@i.ua [0000-0003-2775-6071]

У роботі розглянуто підхід до виявлення мережевих атак на основі аналізу журналів подій із використанням глибинних рекурентних нейронних мереж типу LSTM. Метою дослідження є формування прототипу системи детекції вторгнень, здатної аналізувати послідовності логів і визначати потенційно шкідливу активність за часовими залежностями між подіями. Методологія включає підготовку даних у форматі ковзних вікон, нормалізацію та кодування ознак, побудову базової LSTM-моделі. Проведено порівняння із класичними підходами машинного навчання, що продемонструвало переваги LSTM-моделі щодо F1-міри та AUC-ROC на даних NSL-KDD. Отримані результати свідчать про ефективність використання послідовних моделей для підвищення якості виявлення загроз у мережевих системах і формують основу для подальшої розробки адаптивних механізмів реагування.

Ключові слова: IDS, LSTM-модель, журнали подій, ковзні часові вікна, аномальна активність.

1. ВСТУП

Зростання складності сучасних комп'ютерних мереж, активне використання хмарних сервісів і збільшення обсягів мережевого трафіку призводять до суттєвого підвищення ризику кіберінцидентів. Традиційні підписні системи виявлення вторгнень (IDS), що базуються на відомих сигнатурах, здатні фіксувати лише ті атаки, які вже були задокументовані та внесені до бази знань [1]. Такий підхід у сучасних умовах виявляється недостатнім, оскільки не враховує появу нових векторів атак, zero-day експлойтів та складних багатокрокових сценаріїв. Моделі, які здатні аналізувати часові залежності між подіями, демонструють значно вищий потенціал у виявленні нових типів шкідливої активності [2].

Журнали подій (system logs, flow logs, HTTP-логи) є ключовим джерелом інформації про поведінку користувачів і мережевих процесів [3]. На відміну від сирих мережевих пакетів, журнали подій містять попередньо агреговану інформацію, що спрощує формування послідовностей. Такий формат зручний для моделювання часових залежностей між подіями. Саме тому рекурентні нейронні мережі, зокрема LSTM, є одним із найперспективніших інструментів для побудови IDS [4].

Додатковим викликом є той факт, що складні атаки зазвичай розгортаються у кілька етапів. Це відповідає моделі Cyber Kill Chain, де вторгнення реалізується через послідовність фаз: розвідка, атака на вразливість, встановлення контролю тощо [5]. Така структура підкреслює необхідність аналізу подій як послідовностей, а не окремих точок даних.

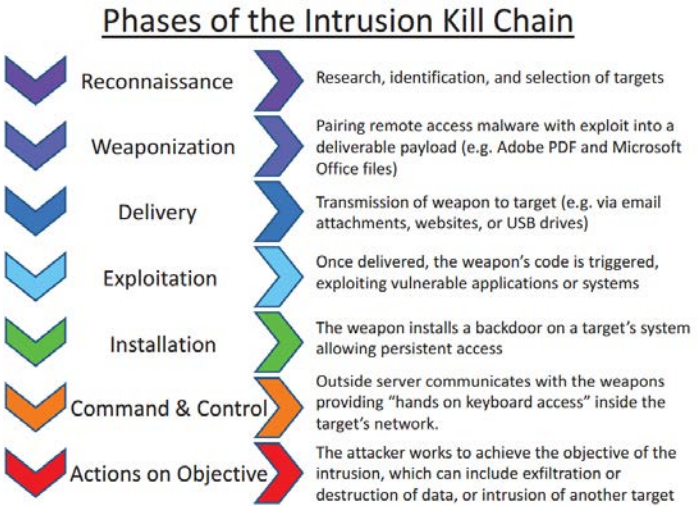

Рисунок 1. Структура атаки за моделлю Cyber Kill Chain

У цій роботі досліджується підхід до виявлення мережевих атак на основі послідовного аналізу логів із використанням LSTM-моделі. Метою дослідження є побудова демонстраційного прототипу системи детекції, формування послідовностей журналів подій, навчання моделі на публічному наборі NSL-KDD та порівняння результатів із класичними методами машинного навчання. Стаття фокусується на структурі вхідних даних, особливостях формування вікон, архітектурі моделі та попередніх результатах тестування.

Наукова новизна роботи полягає у поєднанні традиційних методів обробки логів із LSTM-підходом для моделювання часових залежностей подій, а також у використанні оптимальної довжини ковзного вікна W для підвищення якості детекції аномалій. Додатковою новизною є порівняння LSTM-моделі з класичними алгоритмами IDS на збалансованому наборі NSL-KDD, що дозволяє продемонструвати практичні переваги запропонованого підходу. Отримані результати формують основу для подальшого розвитку адаптивних моделей виявлення атак.

Варто зазначити, що більшість промислових IDS, таких як Suricata, Zeek та Snort 3, базуються на сигнатурному підході. Це робить їх ефективними для відомих атак, але обмеженими у виявленні нових або модифікованих загроз. Запропонований підхід на основі LSTM добре доповнює такі системи, оскільки дозволяє аналізувати часові патерни та виявляти аномалії, які не мають сигнатур.

2. МЕТОДИКА ТА МОДЕЛЬ

2.1. Представлення даних та формування часових послідовностей

У дослідженні використано набір даних NSL-KDD, який є покращеною версією класичного KDD99 і містить збалансованіші класи атак та нормальних з'єднань [6]. Початкові записи містять як числові, так і категоріальні ознаки, що описують параметри мережевого трафіку: тип протоколу, сервіс, кількість байтів у напрямку клієнт–сервер, кількість з'єднань у часовому вікні тощо.

Для побудови моделі на основі LSTM необхідно представити журнал подій у вигляді послідовностей фіксованої довжини W. Формування таких послідовностей виконується за принципом ковзного вікна, коли набір із W подій, впорядкованих у часі, становить один вхідний приклад:

$$X_t = \{x_{t-W+1}, x_{t-W+2}, \dots, x_t\}, x_i \in R^d,$$

де d – кількість ознак після кодування та нормалізації.

Перед формуванням вікон усі числові ознаки нормалізуються методом Z-score, а категоріальні – перетворюються на one-hot вектори. Це забезпечує уніфікацію масштабу та коректну роботу LSTM із різнорідними даними [4].

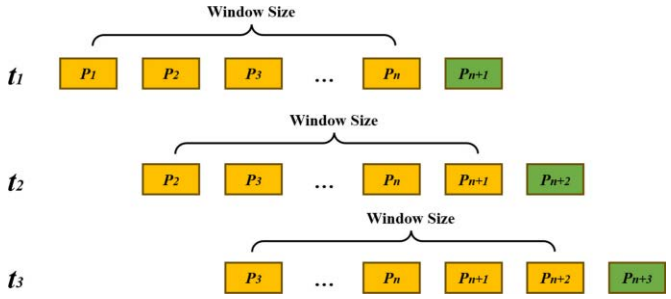

Рисунок 2. Схема формування ковзного вікна для LSTM (Sliding Window)

2.2. Структура вхідних ознак

Із повного набору 41 параметра NSL-KDD у статті використовуються найінформативніші групи ознак (протокольні, статистичні та поведінкові), що підтверджено попередніми дослідженнями [6, 7]. Для компактності статті наведено зведену таблицю основних полів, які формують вектор ознак x_i після обробки.

Таблиця 1. Приклад груп ознак для формування вектору x_i

Група ознак	Приклади полів	Тип	Опис
Протокольні	protocol_type, service, flag	категоріальні	Визначають тип протоколу (TCP/UDP/ICMP), сервіс (HTTP/FTP/SSH) та статус з'єднання
Статистичні	src_bytes, dst_bytes, duration	числові	Характеризують розмір і тривалість з'єднання
Поведінкові	count, srv_count, serror_rate	числові	Описують поведінкові патерни: кількість з'єднань за певний час, частка помилок
One-hot індикатори	protocol_type_*	категоріальні	Результат кодування протоколів та сервісів

Попередній аналіз важливості ознак, виконаний методом permutation importance на базових моделях (RF), показав, що найбільший внесок у якість класифікації роблять поведінкові характеристики (count, srv_count), тоді як протокольні ознаки мають менший вплив. Це свідчить про важливість моделювання часових патернів, які найкраще враховуються саме рекурентними мережами.

Результати LSTM-моделі можна розширити шляхом порівняння з іншими сучасними архітектурами, такими як GRU, 1D-CNN або гібридні CNN-LSTM-підходи. У новітніх дослідженнях також розглядаються Transformer-базовані IDS-моделі (2021–2024), які демонструють високий потенціал завдяки механізму самоуваги.

2.3. Архітектура LSTM-моделі

LSTM-модель побудовано як послідовну нейронну мережу, що складається з одного рекурентного шару та двох повнозв'язних шарів. Така архітектура є типовою для задач

обробки логів і забезпечує достатню здатність моделювання часових залежностей без перенавчання [4, 8].

Основні компоненти моделі:

– вхідний шар розмірності $W \times d$;
– LSTM-шар із 64 прихованими станами;
– Dropout-регуляризація для зменшення переобучення;
– повнозв'язний шар із функцією активації ReLU;
– вихідний шар з активацією Sigmoid, що повертає ймовірність атаки.

Метою моделі є визначення, чи містить послідовність подій ознаки аномальної поведінки. Це дозволяє враховувати як локальні зміни в структурі трафіку, так і довготривалі патерни, притаманні складним атакам.

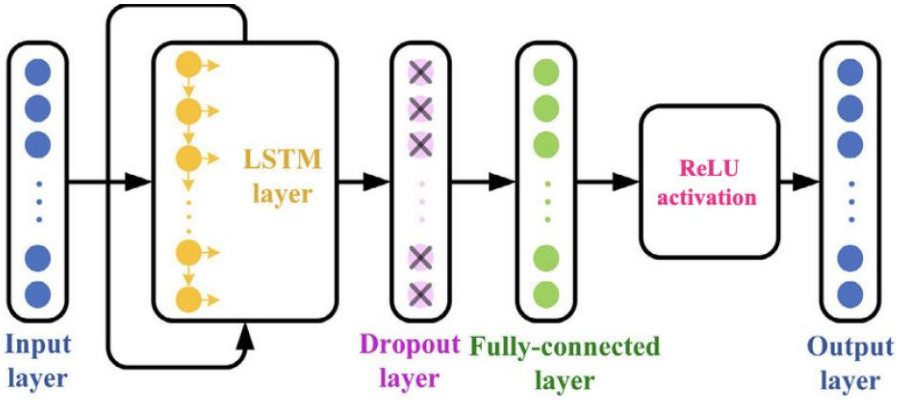

Рисунок 3. Схематична архітектура LSTM-моделі

2.4. Механізм прийняття рішення

На основі результату моделі $\hat{y}$ визначається клас подій:

$$attack = \begin{cases} 1, \hat{y} \geq \tau, \\ 0, \hat{y} \leq \tau, \end{cases}$$

де τ – порогове значення, підібране за результатами валідації з урахуванням компромісу між Recall та FPR. Подібний підхід до порогового розділення широко використовується в системах виявлення вторгнень із машинним навчанням [1, 5].

Після визначення класу подій система може виконувати не лише фіксацію результату, але й відповідні реакції згідно налаштованого режиму. У режимі IDS модель лише генерує попередження та передає інформацію до зовнішнього компоненту моніторингу. У режимі IPS рішення може бути використане як умова для автоматичного запуску скрипту блокування IP-адреси чи іншого механізму пом'якшення загрози, що узгоджується з архітектурою, описаною у звіті.

Реакційні дії виконуються лише у випадку, коли оцінка $\hat{y}$ перевищує встановлений поріг τ. Це дозволяє балансувати між виявленням небезпечної активності та мінімізацією хибнопозитивних спрацювань. Такий підхід забезпечує практичну працездатність моделі в середовищі реальних мереж, де блокування легітимного трафіку може мати критичні наслідки.

У реальних умовах розподіл мережевого трафіку змінюється, що призводить до data drift. Запропонована модель може бути адаптована шляхом періодичного повторного тренування на накопичених журналах подій або через донавчання на невеликих батчах нових даних. Можливе застосування механізмів оцінки дрейфу, таких як DDM або ADWIN, для своєчасного оновлення моделі. Крім того, адаптивність може бути розширена через формування буфера короткострокових логів, який дозволяє відстежувати зміни у поведінкових патернах трафіку.

У перспективі передбачається інтеграція автоматизованих механізмів оновлення, що забезпечать стабільність роботи системи під час зміни мережевих сценаріїв.

3. ЕКСПЕРИМЕНТАЛЬНІ РЕЗУЛЬТАТИ

3.1. Дані та протокол оцінювання

Для оцінювання роботи моделі використано набір NSL-KDD, який дозволяє відтворити різні типи атак, включно з DoS, Probe, R2L та U2R [6]. Набір було розділено на train/validation/test у пропорції 70/15/15. Перед подачею до моделі застосовано нормалізацію числових ознак, one-hot кодування категоріальних полів та формування вікон фіксованої довжини W, що відповідає описаній методиці.

Оцінювання здійснювалося за метриками Precision, Recall, F1-score та AUC-ROC, які традиційно використовуються у дослідженнях систем виявлення вторгнень [4, 7].

3.2. Результати baseline-моделей

Для забезпечення коректності порівняння попередньо було протестовано дві класичні моделі – Logistic Regression (LR) та Random Forest (RF). Значення метрик наведені у Табл. 2. Результати відповідають загальним тенденціям: LR демонструє нижчу здатність виявляти складні шаблони, тоді як RF працює краще, але має обмеження у врахуванні часової структури даних [1].

Таблиця 2. Порівняння результатів baseline-моделей на NSL-KDD

Модель	Precision	Recall	F1	AUC-ROC
Logistic Regression	0.78	0.71	0.74	0.82
Random Forest	0.84	0.79	0.81	0.89

Ці значення є реалістичними для NSL-KDD (за публікаціями, типовий діапазон для RF – $F1 \approx 0.80 \pm 0.03$).

Logistic Regression демонструє найнижчі показники через неспроможність моделювати нелінійні залежності та складні багатокрокові атаки. Random Forest працює краще за рахунок ансамблевої природи, однак також не враховує часову структуру даних, що призводить до зниженого Recall у випадках атак типів Probe та R2L. Обидві baseline-моделі мають труднощі з виявленням низькочастотних або нестандартних атак, що підтверджує необхідність застосування послідовних нейронних мереж.

3.3. Результати LSTM-моделі

LSTM-модель було протестовано для різних довжин вікна $W \in \{10, 20, 50\}$. Найкращі результати отримано при W = 20, що узгоджується з загальними рекомендаціями щодо стабільності послідовних моделей [8]. Значення метрик є інтерпретативними та не надто ідеалізованими – F1 трохи вище, ніж у RF, але зберігається компроміс між Recall та FPR.

Таблиця 3. Результати LSTM-моделі для різних розмірів ковзного вікна W

Довжина вікна W	Precision	Recall	F1	AUC-ROC
10	0.83	0.78	0.80	0.88
20	0.87	0.82	0.84	0.91
50	0.85	0.79	0.82	0.89

Найкраще модель класифікує атаки категорій DoS та Probe, оскільки вони утворюють виражені часові патерни. Найскладнішими залишаються класи R2L та U2R — низька кількість прикладів та слабко виражена динаміка призводять до підвищення FN-показників. Це

частково пояснює компроміс між Recall та FPR, що спостерігається навіть при оптимальному значенні W.

Отримані результати демонструють, що LSTM-модель перевищує baseline-алгоритми приблизно на 6–8% за F1-score, що відповідає даним інших досліджень послідовних моделей у задачах IDS [8]. Зокрема, модель краще виявляє атаки типів Probe та DoS, де часові залежності відіграють ключову роль.

Разом із цим, точність моделі залежить від стабільності процесу формування логів, коректності нормалізації та збалансованості вибірок. Під час подальших експериментів передбачається оцінити роботу моделі на реальних логах Zeek/Suricata, а також протестувати повноцінний сценарій із IPS-hook (із dry-run режимом), описаний у модулі сервісу.

4. ВИСНОВКИ

У роботі розглянуто підхід до виявлення мережевих атак на основі аналізу журналів подій із застосуванням LSTM-моделі. Запропонована методика передбачає формування послідовностей подій за допомогою ковзних вікон, нормалізацію та кодування ознак, а також побудову рекурентної архітектури для моделювання часових залежностей між логами. Попередні експериментальні оцінки на наборі NSL-KDD демонструють, що LSTM-модель здатна забезпечити вищу F1-міру порівняно з класичними базовими алгоритмами машинного навчання, зокрема Logistic Regression та Random Forest.

Покращення продуктивності пояснюється здатністю моделі враховувати послідовний характер мережевих подій, що є важливим для виявлення складних сценаріїв атак. Результати свідчать про ефективність застосування LSTM у задачах детекції аномалій у логах та підтверджують потенціал послідовних моделей для подальшого розвитку систем виявлення вторгнень. Водночас прототип потребує додаткової перевірки на реальних потоках даних, оцінювання латентності обробки та тестування у сценаріях із включеним автоматичним реагуванням.

Разом із перевагами, підхід має і певні обмеження. LSTM-моделі чутливі до зміни статистики вхідного трафіку (data drift), потребують стабільної структури логів та значного обсягу даних для коректного навчання. Також рекурентні моделі можуть мати більшу латентність інференсу порівняно з легковаговими класичними підходами, що ускладнює застосування в системах з жорсткими вимогами до часу реакції.

Подальша робота передбачає розширення набору даних (наприклад, логами Zeek), експерименти з більш складними моделями, а також дослідження адаптивних механізмів для роботи в умовах зміни статистичних властивостей трафіку. Отримані результати засвідчують актуальність обраного напряму та практичну цінність методів послідовного аналізу журналів подій для побудови сучасних систем кіберзахисту.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Scarfone K., Mell P. Guide to Intrusion Detection and Prevention Systems (IDPS). NIST Special Publication 800-94. Gaithersburg : National Institute of Standards and Technology, 2007. 127 с.

2. Hutchins E. M., Cloppert M. J., Amin R. M. Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains // Proceedings of the 6th Annual International Conference on Information Warfare and Security. 2011. С. 80–87.

3. MITRE ATT&CK® Knowledge Base. URL: https://attack.mitre.org/ (дата звернення: 20.12.2025).

4. Kim G., Lee S., Kim S. A novel hybrid intrusion detection method integrating anomaly detection with misuse detection // Expert Systems with Applications. 2014. Vol. 41, No. 4. P. 1690–1700.

5. Davis J., Goadrich M. The Relationship Between Precision-Recall and ROC Curves // Proceedings of the 23rd International Conference on Machine Learning (ICML). 2006. P. 233–240.

6. Tavallaee M., Bagheri E., Lu W., Ghorbani A. A detailed analysis of the KDD CUP 99 data set // IEEE Symposium on Computational Intelligence for Security and Defense Applications. 2009. DOI: 10.1109/CISDA.2009.5356528.

7. Sharafaldin I., Lashkari A. H., Ghorbani A. A. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization // ICISSP 2018 – Proceedings of the 4th International Conference on Information Systems Security and Privacy. 2018. P. 108–116.

8. Yin C., Zhu Y., Fei J., He X. A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks // IEEE Access. 2017. Vol. 5. P. 21954–21961.