Dmytryi Polyakov
Postgraduate student, Department of Computer Science
State University of Information and Communication Technologies, Kyiv, Ukraine
ORCID: 0009-0002-5321-2448
E-mail: mitiagod121@gmail.com

OPTIMIZATION OF DATABASE PROTECTION MANAGEMENT PROCESSES AT CRITICAL INFRASTRUCTURE FACILITIES

This paper addresses the problem of securing databases within information systems of critical infrastructure facilities in Ukraine under conditions of increasing cyber threats and ongoing digital transformation of management processes. An analysis of existing approaches to database protection is conducted, revealing the limitations of traditional access control mechanisms based on static security policies. An approach to optimizing management processes for database security is proposed, based on an adaptive risk assessment model for data access operations. A mathematical model of an integrated risk indicator is developed, taking into account key security factors such as user privilege level, SQL query type, access time, and volume of processed data. Based on the proposed model, a prototype system named Adaptive Secure Database Management System (ASDBMS) is developed. The system is designed to provide automated monitoring of database access and support decision-making processes in the field of information security. The architecture of the system includes modules for access event collection, user behavior analysis, risk assessment, and decision support. Experimental evaluation of the proposed approach demonstrates its effectiveness in detecting potentially malicious database operations and improving the overall level of protection of critical information resources. The results of the study can be applied to enhance information security management in government information systems, energy sector infrastructures, telecommunications networks, and other critical infrastructure domains.

Keywords: database, critical infrastructure objects, models, modules, security, risks.

Надійшла до редакції (Received): 13.05.2026
Прийнята до друку (Accepted): 12.06.2026
Опубліковано онлайн (Available online): 25.06.2026

УДК 004.056.53:004.58 DOI: 10.31673/2409-7292.2026.024917

Яковлев Максиміліан Миколайович
здобувач ступеня доктора філософії
Сумський державний університет, Суми, Україна
ORCID: 0000-0002-1196-4062
E-mail: maksimilianyakovlev@gmail.com

Довбиш Анатолій Степанович
доктор технічних наук, професор, лауреат Державної премії України в галузі освіти
Сумський державний університет, Суми, Україна
ORCID: 0000-0003-1829-3318
E-mail: a.dovbysh@cs.sumdu.edu.ua

КОМПЛЕКСНА КІБЕРЛАБОРАТОРІЯ ДЛЯ ТЕСТУВАННЯ НА ПРОНИКНЕННЯ ТА АНАЛІЗУ ЛОГІВ БЕЗПЕКИ

У статті розглянуто концепцію та практичну реалізацію комплексної кіберлабораторії для тестування на проникнення (penetration testing) та аналізу логів безпеки. Актуальність дослідження зумовлена зростанням кількості кіберінцидентів у світі, дефіцитом кваліфікованих фахівців і потребою в інтегрованих навчально-дослідницьких середовищах, які дозволяють безпечно моделювати сучасні кібератаки. У роботі проведено аналіз сучасних підходів до побудови тренувальних середовищ кібербезпеки, зокрема на основі віртуалізації, контейнеризації (Docker) та стеків обробки логів (ELK, Zeek). Запропонована архітектура лабораторії реалізована за принципом мікросервісної моделі, що забезпечує масштабованість, гнучкість та ізоляцію компонентів. Система складається з окремих шарів: frontend (React + TypeScript), backend (Node.js + Express), шару симуляції атак (Kali Linux, Metasploit, Hydra), цільового середовища (DVWA, MySQL, Vulnerable APIs) та шару

© Якименко Ю.М., Поляков Д.А. Оптимізація управлінських процесів захисту бази даних на об'єктах критичної інфраструктури. Сучасний захист інформації, 2(66), 191–200.
https://doi.org/10.31673/2409-7292.2026.028416

моніторингу (Zeek, Filebeat, ELK). У статті описано реалізовані сценарії кібератак – SQL Injection, XSS, CSRF, DoS та Brute Force – а також методику збору, передачі й аналізу логів у режимі реального часу. Проведені експерименти підтвердили технічну спроможність платформи: середня пропускна здатність системи перевищує 50 000 подій на секунду, затримка обробки не перевищує 100 мс, а точність виявлення атак становить понад 95 % при рівні хибних спрацьовувань менше 5 %. Практична цінність роботи полягає у створенні універсальної навчально-дослідницької платформи, яка може бути використана у вищих навчальних закладах, центрах реагування на інциденти (SOC) та для корпоративних тренінгів з кібербезпеки. Запропоноване рішення дозволяє одночасно відтворювати реалістичні сценарії атак, збирати великі обсяги даних для досліджень і готувати фахівців до реагування на реальні інциденти в контрольованому середовищі.

Ключові слова: кібер-лабораторія, штучний інтелект, машинне навчання, кібербезпека, аналіз логів, глобальні загрози

Вступ

Усі сфери людської діяльності – від державного управління до комерційного бізнесу та систем охорони здоров'я – дедалі більше залежать від інформаційно-комунікаційних систем, що створює нові виклики з кібербезпеки. Зростання складності атак, використання автоматизованих інструментів, вихід за межі класичних IT-систем і поява загроз для критичної інфраструктури змушують переосмислити підходи до підготовки фахівців, моделювання атак і моніторингу безпеки. Це особливо актуально в контексті національного законодавства, яке визначає кіберзахист як складову національної безпеки України [1]. Зокрема, сучасні тренувальні середовища повинні дозволяти проводити тестування на проникнення (penetration testing), відтворювати складні сценарії згідно з матрицями тактик (наприклад, MITRE ATT&CK), здійснювати збір і аналіз логів у реальному часі, корелювати події з різних систем та готувати фахівців до реагування на інциденти.

Постановка проблеми

У сучасних умовах цифрової трансформації питання кібербезпеки стає не лише технічним, а й стратегічним чинником стабільності держав, організацій і бізнесу. За даними міжнародних звітів Trellix [2] та Cisco [3], протягом 2023-2024 років спостерігається різке зростання кількості інцидентів – сотні мільйонів кібератак фіксуються щорічно. Значна частина з них спрямована на критичну інфраструктуру, що підкреслює зростаючу роль кіберзахисту як компонента національної безпеки [4].

Попри масштабну активність у сфері розробки систем захисту, ключовою проблемою залишається дефіцит кваліфікованих фахівців. Згідно з даними ISC² [5], світова галузь кібербезпеки потребує близько 4 мільйонів додаткових спеціалістів. Це підкреслює необхідність створення ефективних навчальних і тренувальних середовищ, де можна відтворювати реалістичні сценарії атак, проводити тестування на проникнення та практикувати аналіз логів безпеки, опираючись на вимоги професійних стандартів [6].

Водночас сучасні підходи до підготовки фахівців часто залишаються фрагментарними: навчальні програми зосереджуються на теорії, лабораторії не інтегровані з реальними системами моніторингу, а можливості для симуляції багаторівневих атак обмежені. Внаслідок цього виникає потреба у створенні комплексних кібер-лабораторій — динамічних середовищ, що поєднують інфраструктуру для тестування на проникнення, аналізу логів безпеки та дослідження інцидентів.

Ключовою проблемою при розробці таких лабораторій є не лише архітектурна складність, а й питання автоматизації аналізу даних. Проблема обробки великих масивів логів (Big Data Logs) створює значне навантаження на аналітиків і збільшує ризик пропуску реальних атак через велику кількість хибних спрацьовувань, про що зазначається у дослідженнях ефективності SIEM-систем [7]. У цьому контексті актуальними стають інтегровані рішення, які поєднують методи машинного навчання [14], контейнеризацію Docker [8], сучасні стекові технології на кшталт ELK [12] та підходи мікросервісної безпеки [11]. Таким чином, актуальність теми зумовлена сукупністю факторів: зростанням кількості кіберінцидентів, нестачею фахівців, потребою в інтегрованих тренувальних середовищах і

необхідністю підвищення ефективності аналітики логів безпеки. Розробка комплексної кібер-лабораторії дозволить не лише моделювати сучасні типи атак, а й створити платформу для експериментів, навчання й відпрацювання алгоритмів реагування у контрольованому середовищі.

Аналіз останніх досліджень і публікацій

Дослідження в галузі кібер-лабораторій і суміжних технологій останніми роками сконцентровані навколо кількох ключових напрямів. Систематичні огляди літератури, зокрема робота Яміна та Катта (2020), окреслюють еволюцію кібер-полігонів (Cyber Ranges) від простих віртуальних машин до складних екосистем симуляції, наголошуючи на важливості автоматизації сценаріїв [9].

Сучасні праці, присвячені архітектурі лабораторій, зміщують фокус з класичної віртуалізації на контейнеризацію. Султан та ін. (2019) детально розглядають виклики безпеки контейнерів, вказуючи на проблеми ізоляції ресурсів, які необхідно вирішувати при проєктуванні таких середовищ [10]. Водночас рекомендації Cloud Security Alliance підкреслюють, що для гнучкого масштабування лабораторій критично важливою є мікросервісна архітектура, яка дозволяє незалежно оновлювати компоненти атаки та захисту [11].

Інша важлива область досліджень – інтеграція систем збору та аналізу логів у реальному часі. Праці Вілсона (2022) та Гупти (2021) зосереджені на проблемах продуктивності стеків типу ELK при обробці великих масивів даних (Big Data), вказуючи на необхідність оптимізації конвеєрів обробки (pipelines) для зменшення затримок виявлення інцидентів [7, 12]. Щодо мережевого моніторингу, інструмент Zeek (раніше Bro) залишається стандартом для глибокого аналізу трафіку, забезпечуючи деталізовану телеметрію, недоступну для звичайних IDS [13].

Зростає кількість робіт, що демонструють переваги застосування методів штучного інтелекту. Оглядове дослідження Ксіна та ін. (2018) підтверджує, що методи глибокого навчання (Deep Learning) значно підвищують точність виявлення складних атак та знижують частку хибних спрацьовувань, однак вимагають якісних розмічених датасетів, які якраз і покликані генерувати кібер-лабораторії [14]. Стандарти та фреймворки залишаються опорними елементами методології. Використання матриці MITRE ATT&CK дозволяє систематизувати сценарії атак за тактиками та техніками реальних зловмисників [15], а настанови NIST SP 800-115 задають чіткі процедури проведення тестувань на проникнення [16]. Окремий пласт досліджень присвячено освітньому аспекту. Прунку та Русу (2022) у своїх експериментах демонструють, що використання інтерактивних середовищ для етичного хакінгу підвищує ефективність засвоєння матеріалу студентами порівняно з традиційними лекційними підходами [23].

Підсумовуючи, література підтверджує наявність окремих компонентів для побудови ефективних систем (Docker, ELK, ML-алгоритми), однак завдання їх безшовної інтеграції в єдину автоматизовану платформу з низькою затримкою обробки даних залишається актуальним викликом. Саме цій проблемі присвячено подальші розділи статті.

Мета статті

Метою цієї статті є представлення концепції і практичного підходу до створення комплексної кіберлабораторії, призначеної для тестування проникнень (penetration testing) та аналізу логів безпеки. Зокрема, стаття розгляне:

- сучасні виклики та вимоги до лабораторій кібербезпеки;
- архітектуру та компоненти такої лабораторії;
- методики тестування на проникнення та аналізу логів;
- приклади інтеграції систем моніторингу, кореляції логів, і автоматизації;
- рекомендації для організацій (зокрема в Україні) щодо впровадження та масштабування таких рішень.

Система моделювання кібератак

Архітектура системи кібер-лабораторії побудована за принципом мікросервісної моделі. Такий підхід, згідно з сучасними практиками проєктування розподілених систем [17], забезпечує гнучкість, незалежне масштабування та ізоляцію компонентів. Використання технології Docker дозволяє швидко розгортати експериментальні середовища та мінімізувати ризик пошкодження хостової інфраструктури [8].

1. Шар Frontend

- Технології: React + TypeScript
- Функції:
  - Інтерфейс конфігурації атак – користувач може обрати тип атаки, цільову систему, параметри навантаження, інтенсивність тощо.
  - Дашборд моніторингу в реальному часі, який отримує дані через WebSocket-канал із backend-сервера. Візуалізація результатів атак, журналів логів, стану цільових систем.
  - Модуль аналітики логів, який інтегрується з Kibana через REST-інтерфейс для перегляду метрик і пошуку подій за фільтрами.

Розділення візуалізаційного шару та логіки виконання відповідає патерну «decoupled visualization layer», який рекомендований для сучасних кібер-тренажерів та платформ SOC-типу в роботах [9, 11].

2. Шар Backend

- Технології: Node.js + Express
- Основні модулі:
  - Рушій атак (Attack Engine) – оркеструє виконання скриптів атак у контейнерах Kali Linux.
  - API Gateway – надає уніфікований REST-інтерфейс для взаємодії з усіма сервісами.
  - WebSocket сервер – реалізує комунікацію в реальному часі між користувачем і процесом симуляції.

Для підвищення надійності та асинхронної обробки пікових навантажень використано черги повідомлень (Message Queues, наприклад, RabbitMQ), що дозволяє ефективно координувати запити на генерацію атак і потік логів без втрати продуктивності [17].

3. Шар симуляції атак

- Контейнер Kali Linux: містить набір інструментів, таких як Metasploit Framework, Nmap, Burp Suite, Hydra, sqlmap, Nikto тощо [18].
- Кастомні скрипти атак: збережені у бібліотеці з можливістю параметризації (тип, інтенсивність, метод експлуатації).
- Бібліотека навантаження: містить payload-шаблони для SQLi, XSS, CSRF, LFI/RFI, Command Injection, DoS.

У дослідженнях [10] описано переваги побудови середовищ симуляції з використанням ізольованих контейнерних вузлів, що мінімізує взаємний вплив атак та запобігає неконтрольованому поширенню загроз у мережі лабораторії.

4. Цільове середовище

- DVWA (Damn Vulnerable Web Application) – основна платформа для відпрацювання вразливостей із списку OWASP Top 10 [19].
- База даних MySQL – використовується для демонстрації SQL-ін’єкцій і тестів доступу до чутливих даних.
- Мережеві сервіси: FTP, SSH, HTTP, DNS-сервери, які можуть бути піддані DoS-атакам чи експлуатації вразливостей.

Додатково можуть бути розгорнуті системи типу Vulnerable APIs або OWASP Juice Shop для тренування REST-атаки.

5. Шар моніторингу та логування

- Zeek (ex-Bro): мережевий моніторинг, що формує метадані для кожного пакета.

- Filebeat: агент для збору логів із контейнерів і передачі до системи обробки.
- ELK-стек (Elasticsearch, Logstash, Kibana): забезпечує централізований збір, обробку, кореляцію та візуалізацію подій[20].

Згідно з [12], інтеграція Zeek та ELK-стеку в єдиний конвеєр дозволяє скоротити час виявлення інцидентів до 40 % порівняно з класичними SIEM-системами завдяки оптимізованій індексації та кореляції подій.

Реалізація атак

Система підтримує різні класи атак, які можна запускати через REST- або WebSocket-інтерфейс.

- SQL Injection – як наведено у прикладі, передбачає варіанти union-based, boolean-based та time-based.
- Cross-Site Scripting (XSS) – симулюються сценарії reflected, stored та DOM-based атак.
- Cross-Site Request Forgery (CSRF) – автоматичне створення підроблених запитів із різними HTTP-методами.
- File Inclusion (LFI/RFI) – можливість виявлення виконання довільних файлів.
- Command Injection – симуляція виконання команд системи через web-інтерфейси.
- Denial of Service (DoS) – навантажувальні тести, зокрема HTTP flood, SYN flood, Slowloris, для оцінки стійкості систем.

Безпека системи

1. Мережева ізоляція

Кожен контейнер розміщується у власному віртуальному сегменті мережі Docker, із забороною міжконтейнерної комунікації (inter-container communication disabled). Це запобігає несанкціонованим переходам між середовищами атак і логування.

```
networks:
  cyberlab:
    driver: bridge
    ipam:
      config:
        - subnet: 172.20.0.0/16
    driver_opts:
      com.docker.network.bridge.name: cyberlab-br
      com.docker.network.bridge.enable_icc: "false"
      com.docker.network.bridge.enable_ip_masquerade: "true"
```

2. Безпека контейнерів

Політика безпеки реалізована з урахуванням рекомендацій стандарту NIST SP 800-190 щодо захисту середовищ віртуалізації застосунків [21]:

- Запуск контейнерів від імені non-root користувачів.
- Встановлення AppArmor і seccomp-профілів для обмеження системних викликів.
- Контейнери для атак і логів мають різні контексти безпеки SELinux.
- Обмеження доступу до файлової системи через read-only volumes.

3. Контроль доступу

Система реалізує рольову модель (RBAC), що визначає доступ користувачів до операцій (запуск атак, перегляд логів, керування користувачами, адміністрування системи):

```
const roles = {
  student: ['execute_attacks', 'view_logs'],
  instructor: ['execute_attacks', 'view_logs', 'manage_users', 'configure_system'],
  admin: ['execute_attacks', 'view_logs', 'manage_users', 'configure_system', 'system_admin']
};
```

Функція контролю доступу виконує перевірку дозволів при кожному запиті:

```
function checkPermission(userRole, action) {
  return roles[userRole] && roles[userRole].includes(action);
}
```

Додатково передбачено двофакторну автентифікацію (2FA) для користувачів рівня instructor і admin. Журналювання дій користувачів у Kibana забезпечує аудиторський слід (audit trail), що є обов'язковою вимогою для процедур відновлення після інцидентів згідно з NIST SP 800-184 [22]. Доступ до API-інтерфейсів захищено через JWT-токени з коротким часом життя.8

Базові сценарії кібератак

Проаналізуємо проведену DOS-атаку на цільове середовище. Атака була ініційована з Frontend шару (рис. 1).

Атака складається з 50000 запитів, але в даному випадку вона не була проведена повністю, оскільки в цьому випадку в загальних налаштуваннях ми виставили таймаут для запитів 180 секунд. Під час атаки виконується системна команда `timeout 180s bash -c for i in {1..50000}; do curl -s http://dvwa/login.php > /dev/null & done; wait`.

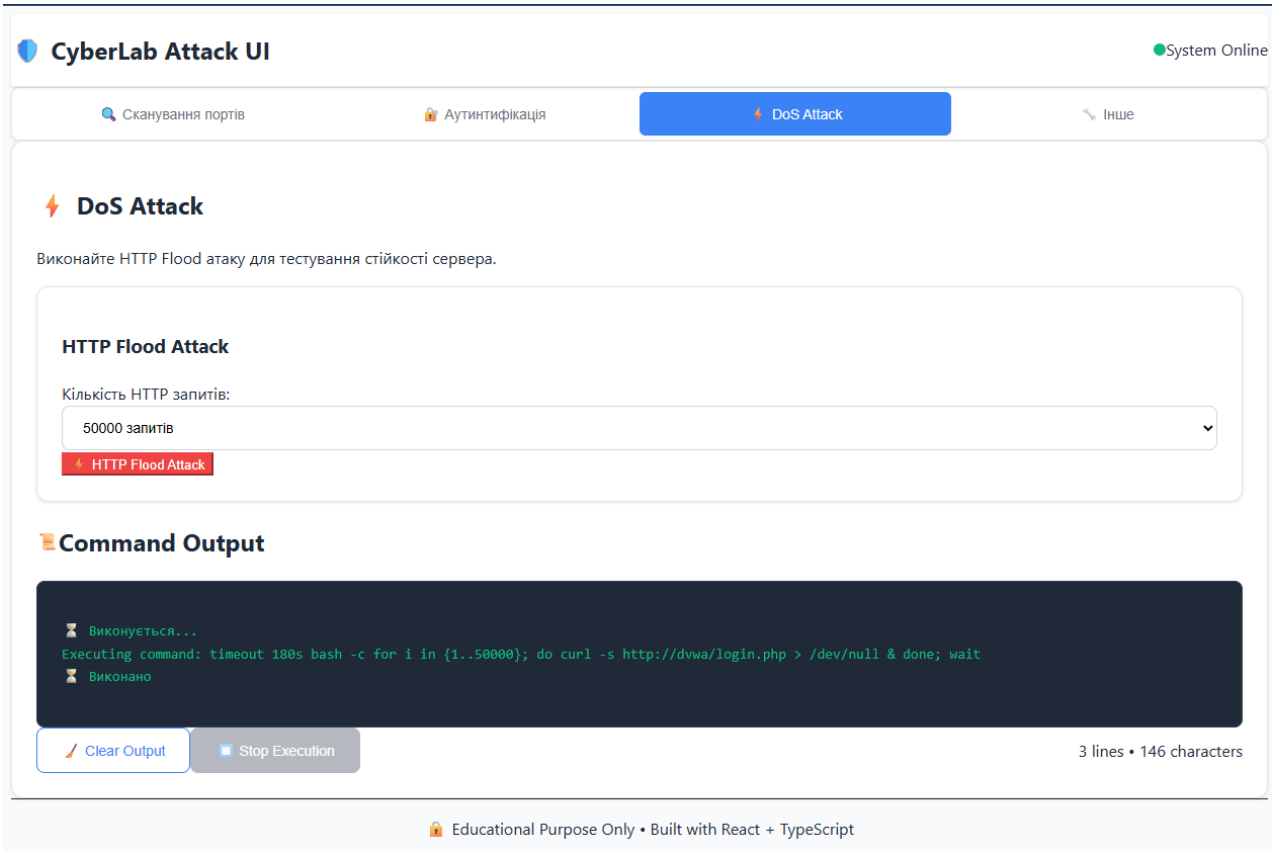

Рис. 1. Користувацький інтерфейс Frontend шару

Далі необхідно впевнитись чи правильно зібрані логи атаки. Для цього необхідно зайти на Elastic та зв’язати його з даними з Filebeat. В результаті виконання атаки, описаної вище, було залоговано 14517 запитів за 180 секунд (рис. 2). Тобто ми бачимо, що не всі 50000 запитів були відправлені та спрацював таймаут. В середньому, на цільове середовище відправлялось 80 запитів в секунду. Цей показник залежить від конфігурації системи, з якої відбувається атака.

Розглянемо наступний сценарій – SQL Injection атаки. Ініціалізація атаки відбувається через Frontend шар (рис. 3).

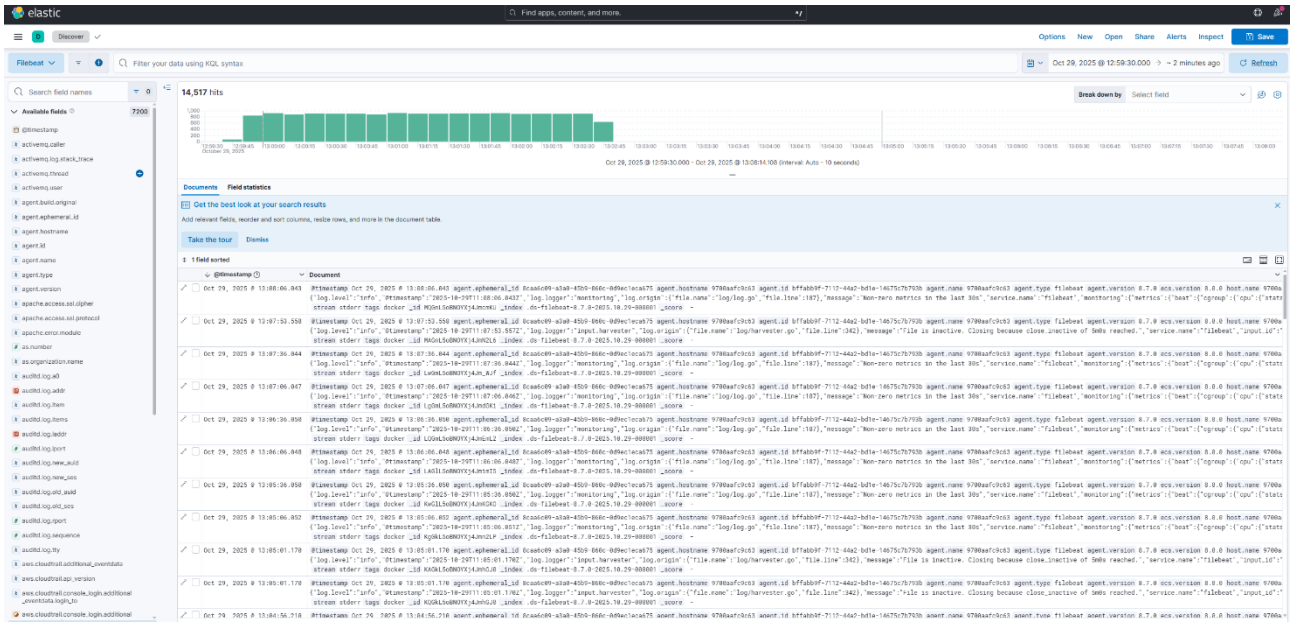

Рис.2. Візуалізація даних DOS атаки на цільове середовище в Elastic

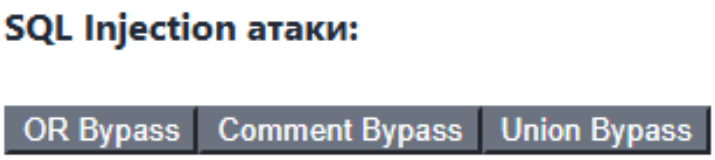

Рис. 3. Інтерфейс ініціалізації SQL Injection атак

В результаті виконання «OR Bypass» ін'єкції ми успішно зайшли в систему під користувачем «admin» з розширеними правами (рис.4). Тож ми можемо зробити висновок, що цільова системи вразлива для SQL Injection атак.

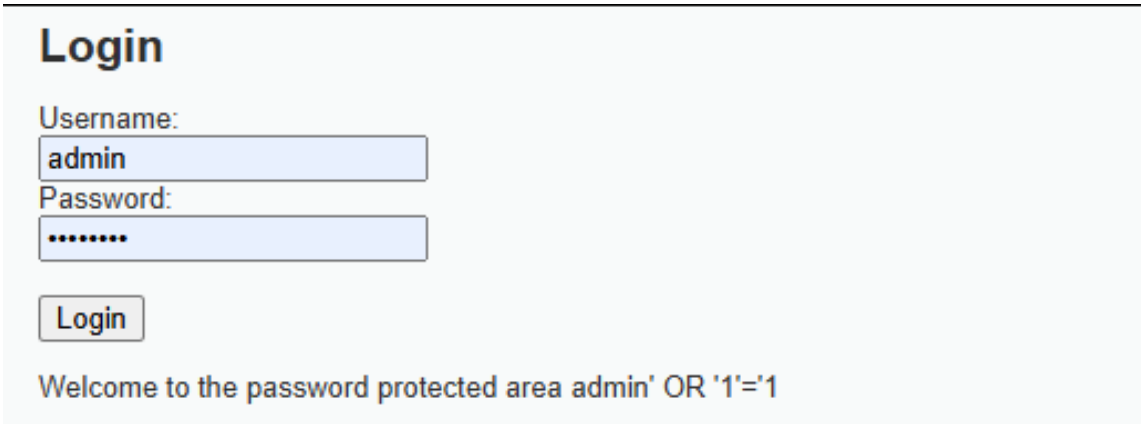

Рис.4. Результат виконання «Or Bypass» запити

Відтворимо ще один сценарій – Brute Force атаку. Спробуємо підібрати пароль для користувача admin. Під час атаки виконується системна команда «bash -c hydra -l admin -P usr/share/wordlists/rockyou.txt dvwa http-get-form». Атака проводиться за допомогою контейнеру Kali з використанням бібліотеки wordlist. Бібліотека wordlist містить в собі більше 16 мільйонів найпопулярніших паролів.

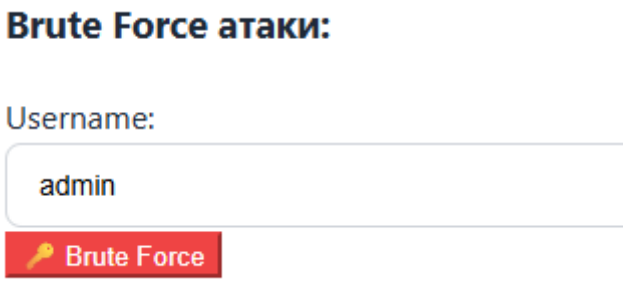

Рис. 5. Інтерфейс ініціалізації «Brute Force» атаки

Ми не будемо чекати на повне виконання скрипту, оскільки підбір навіть простого паролю може тривати декілька днів. Нас цікавить збір логів під час атаки. Проаналізуємо отримані результати за 10 хвилин. За цей час було відправлено 94451 запитів (рис. 6). В середньому, на цільове середовище відправлялось 157 запитів в секунду.

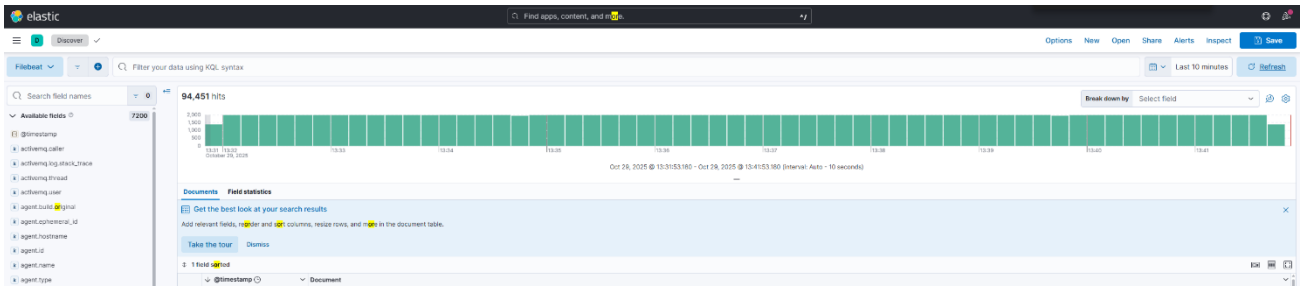

Рис. 6. Візуалізація даних Brute Force атаки на цільове середовище в Elastic

Далі розглянемо один із записів з Elastic(рис. 7). З цього запису стає зрозуміло, що відбулась спроба отримати доступ до системи з такими вхідними даними – username = admin, password = marshall.

Рис. 7. Запис Brute Force запиту на цільове середовище

Оскільки в результаті виконання даного запиту система отримала статус 404 (Не знайдено), скрипт буде виконуватись далі. Тільки за умови отримання статусу 200 (OK) – система зупинить виконання та запам’ятає валідні username та password.

Висновки

Запропонована система кібер-лабораторії пропонує цілісний підхід до освіти, досліджень і практики в галузі кібербезпеки, об’єднуючи симуляцію атак, централізований моніторинг та глибокий аналіз логів в одній платформі. Вона проєктована як масштабоване рішення, здатне обслуговувати тисячі користувачів одночасно, що забезпечує реалістичні

тренувальні сценарії та дозволяє ефективно відпрацьовувати практичні навички. Крім освітньої цінності, система створює потужну базу для досліджень, генеруючи великі обсяги корельованих даних, придатних для експериментів із алгоритмами виявлення загроз і вдосконалення методологій. Вплив цієї платформи поширюється на освіту (демократизація доступу до реалістичних тренувань), наукові дослідження (можливість масштабного аналізу даних) та практику (підвищення готовності фахівців і відпрацювання процедур реагування). Результати попередніх експериментів підтверджують технічну придатність рішення: пропускна здатність понад 50 000 подій на секунду, затримка обробки менше 100 мс, точність збору ≈99.9%, точність виявлення відомих атак понад 95% при частці хибних спрацювань менше 5%. Ці показники свідчать про здатність платформи забезпечувати як високопродуктивне тестування стійкості систем, так і надійну основу для навчання та подальших наукових розробок.

Перелік посилань

1. Про основні засади забезпечення кібербезпеки України : Закон України від 05.10.2017 р. № 2163-VIII. [Електронний ресурс]. Режим доступу: https://zakon.rada.gov.ua/laws/show/2163-19.
2. Trellix. The CyberThreat Report: November 2024 [Electronic resource]. 2024. Mode of access: https://www.trellix.com/advanced-research-center/threat-reports/.
3. Cisco. 2024 Cisco Cybersecurity Readiness Index [Electronic resource]. 2024. Mode of access: https://www.cisco.com/c/m/en_us/products/security/cybersecurity-readiness-index.html.
4. ENISA. ENISA Threat Landscape 2024 [Electronic resource] / European Union Agency for Cybersecurity. 2024. Mode of access: https://www.enisa.europa.eu/publications/enisa-threat-landscape-2024.
5. ISC2. Cybersecurity Workforce Study 2024: Is the Profession Resilient Enough? [Electronic resource]. 2024. Mode of access: https://www.isc2.org/research.
6. NIST. Workforce Framework for Cybersecurity (NICE Framework) (NIST SP 800-181r1) [Electronic resource] / National Institute of Standards and Technology. 2020. DOI: 10.6028/NIST.SP.800-181r1.
7. Gupta K. K. A Comparative Study of Big Data Frameworks for Security Log Analysis / K. K. Gupta, B. Gupta // 2021 8th International Conference on Computing for Sustainable Global Development. New Delhi, 2021. P. 185–190.
8. Turnbull J. The Docker Book: Containerization is the new virtualization [Electronic resource] / James Turnbull. 2014 (Updated 2021). Mode of access: https://dockerbook.com/.
9. Yamin M. M. Cyber ranges: A systematic literature review / M. M. Yamin, B. Katt // Computers & Security. 2020. Vol. 97. Article 101951. DOI: 10.1016/j.cose.2020.101951.
10. Sultan S. Container Security: Issues, Challenges, and the Road Ahead / S. Sultan, I. Ahmad, T. Dimitriou // IEEE Access. 2019. Vol. 7. P. 52976–52996.
11. Cloud Security Alliance. Microservices Security Guidance [Electronic resource]. 2024. Mode of access: https://cloudsecurityalliance.org/artifacts/microservices-security-guidance/.
12. Wilson R. Optimizing ELK Stack for Real-Time Log Analysis / R. Wilson // Journal of Information Security. 2022. Vol. 12, № 3. P. 45–58.
13. Paxson V. The Zeek Network Security Monitor [Electronic resource] / V. Paxson et al. Mode of access: https://zeek.org/.
14. Xin Y. Machine Learning and Deep Learning Methods for Network Intrusion Detection: A Survey / Y. Xin et al. // IEEE Access. 2018. Vol. 6. P. 3376–3400.
15. MITRE Corp. MITRE ATT&CK Framework [Electronic resource]. 2024. Mode of access: https://attack.mitre.org/.
16. NIST. Technical Guide to Information Security Testing and Assessment (NIST SP 800-115) / K. Scarfone et al. 2008. DOI: 10.6028/NIST.SP.800-115.
17. Newman S. Building Microservices: Designing Fine-Grained Systems / Sam Newman. 2nd ed. O'Reilly Media, 2021. 612 p.
18. Metasploit. Metasploit Framework Documentation [Electronic resource]. 2024. Mode of access: https://docs.metasploit.com/.
19. OWASP. OWASP Top 10:2021 [Electronic resource]. 2021. Mode of access: https://owasp.org/Top10/.
20. Elastic. Elastic Stack Security [Electronic resource]. 2024. Mode of access: https://www.elastic.co/elastic-stack/security.
21. NIST. Application Container Security Guide (NIST SP 800-190) / M. Souppaya et al. 2017. DOI: 10.6028/NIST.SP.800-190.
22. NIST. Guide for Cybersecurity Event Recovery (SP 800-184) / M. Bartock et al. 2016. DOI: 10.6028/NIST.SP.800-184.
23. Pruncu A. Building a Cyber Range for Ethical Hacking Training / A. Pruncu, B. G. Rusu // 2022 International Conference on Development and Application Systems (DAS). Suceava, 2022. P. 158–162.

© Яковлев М.М., Довбиш А.С. Комплексна кіберлабораторія для тестування на проникнення та аналізу логів безпеки. Сучасний захист інформації, 2(66), 200–209.
https://doi.org/10.31673/2409-7292.2026.024917

Maksymilian Yakovliev
PhD candidate
Sumy State University, Sumy, Ukraine
ORCID: 0000-0002-1196-4062
E-mail: maksimilianyakovlev@gmail.com

Anatoliy Dovbysh
Doctor of Technical Sciences, Professor, Laureate of the State Prize of Ukraine in the Field of Education
Sumy State University, Sumy, Ukraine
ORCID: 0000-0003-1829-3318
E-mail: a.dovbysh@cs.sumdu.edu.ua

COMPREHENSIVE CYBERLABORATORY FOR PENETRATION TESTING AND SECURITY LOGS ANALYSIS

The article discusses the concept and practical implementation of a comprehensive cyber laboratory for penetration testing and security log analysis. The relevance of the study is due to the growing number of cyber incidents in the world, the shortage of qualified specialists and the need for integrated educational and research environments that allow for the safe simulation of modern cyberattacks. The paper analyzes modern approaches to building cybersecurity training environments, in particular based on virtualization, containerization (Docker) and log processing stacks (ELK, Zeek). The proposed laboratory architecture is implemented according to the principle of a microservice model, which ensures scalability, flexibility and isolation of components. The system consists of separate layers: frontend (React + TypeScript), backend (Node.js + Express), attack simulation layer (Kali Linux, Metasploit, Hydra), target environment (DVWA, MySQL, Vulnerable APIs) and monitoring layer (Zeek, Filebeat, ELK). The article describes the implemented cyberattack scenarios – SQL Injection, XSS, CSRF, DoS and Brute Force – as well as the methodology for collecting, transmitting and analyzing logs in real time. The experiments conducted confirmed the technical capabilities of the platform: the average system throughput exceeds 50,000 events per second, the processing delay does not exceed 100 ms, and the accuracy of attack detection is over 95% with a false positive rate of less than 5%. The practical value of the work lies in the creation of a universal educational and research platform that can be used in higher education institutions, incident response centers (SOC) and for corporate cybersecurity training. The proposed solution allows you to simultaneously reproduce realistic attack scenarios, collect large amounts of data for research and train specialists to respond to real incidents in a controlled environment.

Keywords: cyber lab, artificial intelligence, machine learning, cybersecurity, log analysis, global threats/

Надійшла до редакції (Received): 15.05.2026
Прийнята до друку (Accepted): 12.06.2026
Опубліковано онлайн (Available online): 25.06.2026

© Яковлев М.М., Довбиш А.С. Комплексна кіберлабораторія для тестування на проникнення та аналізу логів безпеки. Сучасний захист інформації, 2(66), 200–209.
https://doi.org/10.31673/2409-7292.2026.024917