***Клименко Д. О., магістрант,***
***Чижмотря О. В., ст. викладач***
*Державний університет «Житомирська політехніка»*

**ОСОБЛИВОСТІ ІНТЕГРАЦІЇ АЛГОРИТМІВ МАШИННОГО НАВЧАННЯ ДЛЯ ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ ВЕБСАЙТІВ**

Широко використовувані сервери, такі як Apache, Nginx або IIS, є основою для обслуговування величезних обсягів трафіку. Але, зростання складності та кількості кібератак, включаючи DDoS, SQL-ін'єкції та автоматизовані боти, створює потребу у впровадженні передових рішень для забезпечення безпеки. Інтеграція алгоритмів машинного навчання у вебсервери дозволяє ефективно виявляти та реагувати на загрози, які обходять нові методи захисту.

У даній статі було проаналізовано особливості інтеграцій алгоритмів машинного навчання у вебсистеми для забезпечення захисту. Було проаналізовано основні деталі кожного етапу впровадження, визначено виклики процесу та зроблено висновки.

У процесі інтеграції важливо звертати увагу на технічні аспекти системи. Вебсервери генерують великі обсяги лог-файлів, що містять інформацію про запити, включаючи IP-адреси, часові мітки, HTTP-методи, параметри запитів та типи пристроїв [1]. Ці дані необхідно автоматизовано збирати та передавати в алгоритми аналізу. Використання модулів збору даних, таких як Logstash чи Fluentd, забезпечує можливість інтеграції потоків логів безпосередньо у системи машинного навчання.

Зібрані дані нерідко зберігаються у форматі, який не може бути оброблений алгоритмом. Тому потрібно попередньо обробляти отриману інформацію. Процес включає в себе очищення даних від пустих значень, дубльованих записів, хибні метрики, визначення загальний особливостей, нормалізація та приведення до уніфікованого форматі всіх отриманих записів. Для релазіації часто використовують такі інструменти як LUA-скрипти для Nginx або Python-плагіни.

Також алгоритми машинного навчання можуть бути інтегровані напряму у сервер як локальні модулі або використовуватися як віддалені сервіси [2]. Локальна інтеграція дозволяє алгоритмам працювати у межах самого сервера, наприклад, за допомогою модулів на кшталт ModSecurity для Apache. Такий підхід забезпечує миттєвий аналіз трафіку, але обмежений продуктивністю серверного обладнання. Віддалений аналіз, навпаки, передбачає передачу даних у хмарні сервіси, такі як AWS WAF. Це забезпечує вищу масштабованість і можливість використання потужних алгоритмів, але може створювати затримки в обробці запитів.

Ефективність роботи таких алгоритмів підвищується у поєднанні з базами даних кіберзагроз, наприклад OWASP [3]. Використання шаблонів атак допомагає ідентифікувати відомі загрози та швидко адаптувати системи до нових викликів.

Для обробки даних у реальному часі часто застосовуються такі алгоритми, як глибокі нейронні мережі, що дозволяють виявляти складні шаблони загроз, або автоенкодери, які аналізують дані для пошуку небезпечних відхилень. Регресійні моделі використовуються для прогнозування навантаження на сервер, а алгоритми кластеризації, наприклад як K-Means, допомагають сегментувати трафік для подальшого аналізу. Гібридні підходи, що поєднують кілька методів, підвищують точність і надійність системи.

Одним із головних викликів інтеграції є забезпечення високої продуктивності алгоритмів при роботі з великими обсягами даних. Для цього використовуються оптимізації, такі як попередня фільтрація, що дозволяє зменшити навантаження на сервер. Важливою проблемою є також хибнопозитивні спрацьовування, які можуть заважати нормальній роботі сервера, блокуючи легітимний трафік.

Отже, інтеграція алгоритмів машинного навчання у вебсервери значно підвищує ефективність протидії сучасним кіберзагрозам. Це дозволяє забезпечити адаптивність і точність у виявленні загроз, недоступних для традиційних методів. Незважаючи на технічні виклики, такі рішення сприяють надійному захисту та масштабованості вебресурсів.

**Список використаних джерел**

1. Integration of Machine Learning with Cybersecurity. ResearchGate. URL: https://www.researchgate.net/publication/374515411_Integration_of_Machine_Learning_with_Cybersecurity_Applications_and_Challenges.
2. Machine Learning Algorithms for Detecting and Preventing Cyber Threats. OxJournal | Academic education journal. URL: https://www.oxjournal.org/machine-learning-algorithms-for-detecting-and-preventing-cyber-threats.
3. Loshin P. What is OWASP. Search Software Quality. URL: https://www.techtarget.com/searchsoftwarequality/definition/OWASP.