УДК 004.492

ЗАСТОСУВАННЯ АЛГОРИТМІВ МАШИННОГО НАВЧАННЯ ДЛЯ ДЕТЕКЦІЇ ШКІДЛИВОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ЧЕРЕЗ АНАЛІЗ PE-ЗАГОЛОВКІВ

А. В. Хандрос[1,а], В. М. Ткач[1,б]

[1] Навчально-науковий Фізико-технічний інститут

Анотація

Враховуючи зростання кіберзагроз, розробка методів штучного інтелекту для виявлення шкідливого програмного забезпечення є критичною. Ми розглядаємо різні техніки, такі як статичний аналіз, хешування та класифікація, для ідентифікації потенційно шкідливих файлів. Основні результати включають розробку надійної моделі, здатної виявляти шкідливі програми з високою точністю, а також обговорення викликів, пов'язаних з обфускацією та поліморфізмом шкідливого ПЗ. Робота підкреслює потенціал машинного навчання як важливого інструменту у сфері кібербезпеки.

Ключові слова: Штучний інтелект, машинне навчання, кібербезпека, шкідливе програмне забезпечення, виявлення шкідливого програмного забезпечення, автоматизоване виявлення, алгоритми машинного навчання, захист від шкідливих програм.

Вступ

Шкідливе програмне забезпечення–це категорія кіберзагроз, яка охоплює програми та код, розроблені для нанесення шкоди користувачам або системам. Ці програми можуть порушувати нормальну роботу систем, красти конфіденційні дані або поширювати інші види шкідливих програм через інтернет-підключені пристрої. Шкідливе програмне забезпечення часто маскується під законні додатки або поширюється через вразливості веб-сервісів, включно з серверами додатків та базами даних, де зберігається важлива інформація.

Серед найпоширеніших видів шкідливого програмного забезпечення можна виділити:

- **Віруси:** самореплікуючі програми, що інфікують файли, змінюючи їх для включення копії вірусу.
- **Черв'яки:** самостійні шкідливі програми, які розповсюджують себе через мережі, експлуатуючи вразливості в операційних системах.
- **Троянські коні:** маскують себе як легітимне програмне забезпечення, але виконують шкідливі дії на комп'ютері жертви, такі як викрадення даних або установка інших шкідливих програм.
- **Spyware :** приховано збирає інформацію про користувача або організацію без їх відома.
- **Ransomware :** блокує доступ до системи або файлів користувача і вимагає викуп за їх розблокування.
- **Rootkits:** забезпечують повний і прихований доступ до комп'ютера, ускладнюючи виявлення і видалення шкідливих програм.

[а]hanart-ipt24@lll.kpi.ua
[б]v.tkach@kpi.ua

1. Методи виявлення шкідливих програм на основі PE-заголовків

Основним методом виявлення шкідливих програм на основі PE-заголовків (Portable Executable) є аналіз структури виконуваних файлів Windows, які включають .exe, .dll, .sys та інші формати. PE-заголовки містять важливу метаінформацію, яка дозволяє ідентифікувати потенційні загрози на основі аномалій у їх структурі. Методи статичного аналізу, включаючи розгляд викликів Windows API та загальних атрибутів файлів, забезпечують засоби для класифікації та ідентифікації шкідливих файлів [1, 2].

Аналіз заголовків PE-файлів зокрема зосереджується на ідентифікації «упакованих» файлів, що часто використовуються для приховування шкідливого коду. Техніки, як PHAD (PE file Header Analysis-based packed file Detection), розроблені для розрізнення між звичайними та модифікованими заголовками, що є важливим для виявлення сучасних шкідливих програм [3].

Для аналізу шкідливого ПЗ на основі PE-заголовків використовуються різноманітні методи машинного навчання, які аналізують метадані файлу, включаючи таблиці імпорту та експорту. Зокрема, алгоритми як випадкові ліси, градієнтний бустинг та нейронні мережі демонструють високу ефективність у класифікації та ідентифікації шкідливих патернів.

Глибоке підсилювальне навчання відкриває нові можливості для раннього виявлення шкідливого ПЗ, особливо вимагачів. Використання цієї техніки до-

зволяє моделям навчатися на основі реакції системи на зловмисні дії, адаптуючись до нових загроз і стратегій обходу [4].

2. Основні проблеми виявлення шкідливих програм на основі PE-заголовків

При виявленні шкідливих програм на основі PE-заголовків (Portable Executable), аналітики кібербезпеки часто стикаються з рядом складнощів. Ці заголовки, які містять ключову інформацію про виконувані файли у системах Windows, можуть бути маніпульовані або змінені шкідливими програмами для уникнення виявлення.

2.1. Технічні виклики

Один з основних технічних викликів полягає у обфускації та поліморфізмі шкідливого ПЗ, що ускладнює точне визначення зловмисного коду через статичний аналіз PE-заголовків. Шкідливі програми можуть динамічно змінювати свої PE-заголовки, що робить використання традиційних сигнатурних методів менш ефективними.

2.2. Помилкові позитивні та негативні результати

Додаткову складність вносять високі рівні помилково позитивних та помилково негативних результатів. Часто системи виявлення шкідливих програм на основі PE-заголовків можуть неправильно класифікувати легітимні програми як шкідливі, що призводить до великої кількості хибних спрацьовувань. З іншого боку, деякі справжні загрози можуть залишатися невиявленими, оскільки вони ефективно маскують свою присутність.

2.3. Потреба в адаптації та оновленні методів

Динамічна природа сучасних шкідливих програм вимагає від аналітичних систем неперервного оновлення та адаптації. Використання методів машинного навчання та розширене моделювання загроз можуть допомогти підвищити точність виявлення, адаптуючи системи до нових шкідливих патернів та стратегій обходу виявлення.

Ці виклики підкреслюють складність задачі виявлення шкідливих програм на основі PE-заголовків і підтверджують необхідність вдосконалення існуючих технологій для забезпечення надійної кібербезпеки.

3. Впровадження алгоритму машинного навчання для класифікації за PE-заголовками

З метою підвищення ефективності систем виявлення шкідливих програм, впровадження алгоритмів машинного навчання, які аналізують PE-заголовки, може значно покращити точність класифікації шкідливих файлів. Особливий інтерес становить розробка класифікаційних моделей, що базуються на аналізі характеристик PE-файлів, зокрема, за допомогою методів машинного навчання, таких як аналіз асоціацій у викликах API, який є ключовим у визначенні шкідливої поведінки файлів [1, 5].

Використовуючи методи аналізу з наукової роботи, можемо створити наступний загальний алгоритм впровадження:

1. **Збір та обробка даних:** На першому етапі, дані, що містять PE-заголовки файлів, збираються та підготовлюються для аналізу. Важливо забезпечити, що ці дані містять достатньо інформації про типи шкідливих та безпечних файлів для ефективного навчання моделі. Для тренування моделей використовується відкритий набір даних Ember, який містить значну кількість метаданих шкідливих та безпечних файлів, що дозволяє проводити глибокий аналіз зловмисного ПЗ [6].
2. **Вибір алгоритму:** Алгоритми машинного навчання, такі як випадкові ліси, градієнтний бустинг чи нейронні мережі, вибираються на основі їх здатності ефективно класифікувати великі обсяги даних і їх адаптивності до нових, раніше невідомих шкідливих програм.
3. **Тренування моделі:** Використовуючи вибрані алгоритми, модель тренується на підготовлених даних. Процес навчання включає в себе валідацію та налаштування гіперпараметрів для забезпечення оптимальної продуктивності моделі.
4. **Тестування та валідація:** Після тренування моделі проводиться її тестування на незалежному наборі даних для оцінки точності та здатності до узагальнення. Це важливо для підтвердження, що модель правильно ідентифікує шкідливі файли без значної кількості помилок.
5. **Імплементація та моніторинг:** Завершальний етап полягає в імплементації тренованої моделі у виробниче середовище та постійному моніторингу її продуктивності. Необхідно регулярно оновлювати модель, щоб вона залишалася ефективною у змінних умовах та нових типах шкідливих програм.

4. Аналіз різних алгоритмів

У рамках дослідження було застосовано кілька алгоритмів машинного навчання для класифікації шкідливого програмного забезпечення за допомогою аналізу PE-заголовків. Аналізуючи результати, можна відзначити наступне [рис.1]:

- **Decision Tree** та **Random Forest** алгоритми показали високу точність (Accuracy: 0.99) та F1 Score (0.99), що свідчить про їх високу ефективність у класифікації шкідливих програм. Відносно мала кількість помилок першого (FP) та другого (FN) роду сприяє високій точності та надійності цих моделей.
- **K-Nearest Neighbors (KNN)** також продемонстрував порівнянні результати з точністю

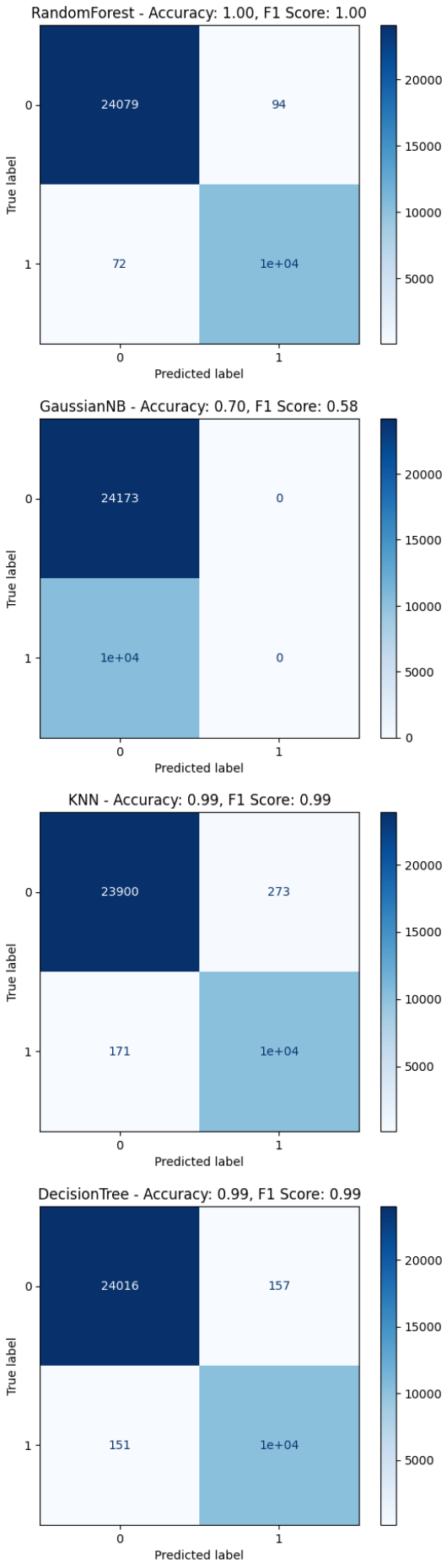

Рис. 1. Матриці невідповідності

та F1 Score як 0.99. Це підкреслює потенціал алгоритму у задачах, де важливим є точне і швидке розпізнавання шкідливих елементів.

- Навпаки, **Gaussian Naive Bayes** показав значно гірші результати з точністю 0.70 та F1 Score 0.58. Це може бути пов'язано з припущеннями, властивими цьому алгоритму, такими як незалежність ознак, що часто не є випадком у реальних даних.

Ці результати вказують на те, що ансамблеві методи, такі як Random Forest та алгоритми, засновані на інстансах, як KNN, можуть бути більш ефективними для задач виявлення шкідливого ПЗ порівняно з простішими ймовірнісними моделями. Враховуючи ці висновки, подальші дослідження можуть бути спрямовані на оптимізацію параметрів для ансамблевих моделей та дослідження альтернативних підходів для покращення результатів класифікації в складних сценаріях розпізнавання шкідливих програм.

Висновки

В роботі демонструється значний потенціал машинного навчання у виявленні шкідливих програм через аналіз PE-заголовків. Використання розширених алгоритмів дозволяє точно класифікувати шкідливі файли, знижуючи ризик помилкових позитивних результатів. Важливо поєднувати ці технології з експертними знаннями для розробки комплексних оборонних стратегій, що захищають цифрові активи від сучасних кіберзагроз.

Перелік використаних джерел

1. An intelligent PE-malware detection system based on association mining / Y. Ye, D. Wang, T. Li, D. Ye, Q. Jiang // Journal in Computer Virology. — 2008. — Т. 4, № 4. — С. 323—334.
2. *Namita, Prachi.* PE file-based malware detection using machine learning // Proceedings of International Conference on Artificial Intelligence. — Springer, 2021. — С. 117—130.
3. PE file header analysis-based packed PE file detection technique (PHAD) / Y. Choi, I. Kim, J. Oh, J. Ryou // 2008 International Symposium on Communications and Information Technologies. — IEEE. 2008. — С. 645—650.
4. Deep Learning for Ransomware Detection: A Novel Approach Using Reinforcement Learning / X. Deng, M. Cen, M. Jiang, M. Lu // Journal of Cybersecurity and Threat Intelligence. — 2024. — Т. 7, № 1. — С. 34—56.
5. *Abdessadki I., Lazaar S.* A new classification based model for malicious PE files detection // International Journal of Computer Network and Information Security. — 2019. — Т. 11, № 6. — С. 1—10.
6. *Anderson H. S., Roth P.* EMBER: An Open Dataset for Training Static PE Malware Machine Learning Models. — 2018. — arXiv: 1804.04637 [cs.CR].