DOI 10.31891/2307-5732-2024-337-3-20
УДК 004.94

КОЗЛОВСЬКИЙ ОЛЕКСАНДР
Херсонський національний технічний університет
https://orcid.org/0009-0006-1864-1107
e-mail: oleksandr.v.kozlovskyi@gmail.com
ЖАРІКОВА МАРИНА
Херсонський національний технічний університет
https://orcid.org/0000-0001-6144-480X
e-mail: marina.jarikova@gmail.com

МОДЕЛЮВАННЯ ТА РОЗРОБКА КОНЦЕПЦІЇ БАГАТОРІВНЕВОГО ФРЕЙМВОРКУ ДЛЯ СИСТЕМИ КІБЕРБЕЗПЕКИ

У даній роботі представлено концепцію та архітектуру багаторівневого фреймворку для створення та налаштування моделі безпеки на основі даних в системі кібербезпеки, що дозволить забезпечити стійкість, стабільність та неперервність роботи систем у випадку кібератак та інших інцидентів безпеки.

Ключові слова: *кібербезпека, Машинне навчання, нейронна мережа, патерн безпеки, кібератака, модель безпеки, інцидент безпеки.*

KOZLOVSKYI OLEKSANDR, ZHARIKOVA MARYNA
Kherson National Technical University

MODELING AND DEVELOPMENT OF A CONCEPT OF A MULTI-LAYERED FRAMEWORK FOR CYBERSECURITY SYSTEM

In this article, the concept and architecture of a multi-layered framework for developing and configuring a data-driven security model in a cybersecurity system, was described. The initial layer involves collecting security data, which forms a bridge between security issues in the cyber-infrastructure and appropriate data-driven solutions. Collecting security patterns or insights from security data and building an appropriate data-driven model is crucial for making the security system automated and intelligent. Security data preparation layer is responsible for providing training and learning data from various sources for the resultant model. Both data quality and quantity determine the ability to solve a security problem, effective data pre-processing, cleaning, and normalization can play a significant role to build an effective security model. The ML based security modeling layer is the main step where insights and knowledge are extracted from the prepared data for further model composing. For this purpose, several ML methods, such as feature engineering, data clustering and classification can be used, as well as DL methods based on recurrent or convolutional neural networks. At this level, the model learns to classify and predict threats, as well as detect anomalous behavior using classification and regression methods. Gradual learning and dynamism layer is concerned with finalizing of the resultant security model. On this step, the security model is being updated by incorporating the latest DD security patterns to improve efficiency. All modules can be applied either together or separately, depending on the specific security issue. Thus, the multilayered framework will allow to build a security model to ensure the resilience and stability of security system under cyberattacks and other security incidents.

Key words: *Cybersecurity, Machine learning, Neural network, Security pattern, Cyberattack, Security model, Security incident.*

Постановка проблеми

Через зростаючу залежність від цифровізації різноманітні інциденти безпеки, такі як несанкціонований доступ, атака зловмисного ПЗ (програмне забезпечення), атака нульового дня, витік даних, відмова в обслуговуванні або фішинг за останні роки зросли експоненціально. Наприклад, у 2010 році світовій спільноті було відомо менше 50 мільйонів унікальних екземплярів зловмисного ПЗ. До 2012 року їх кількість збільшилася приблизно на 100 мільйонів, а в 2019 році спільноті стало відомо про понад 900 мільйонів екземплярів, і ця кількість, ймовірно, зростатиме, згідно зі статистикою інституту AV-TEST у Німеччині [1].

Кіберзлочинність і атаки можуть спричинити руйнівні фінансові втрати, а також вплинути на організації та фізичних осіб. За оцінками, один витік даних коштує 8.19 млн доларів у США і 3.9 млн доларів в середньому, а щорічні збитки світової економіки від кіберзлочинності становлять 400 млрд доларів. За даними організації Juniper Research [2], кількість витоків даних потроїться протягом наступних 5 років. Таким чином, важливо, щоб організації ухвалили та впровадили потужні заходи щодо безпеки в інформаційному просторі, щоб зменшити втрати.

Відповідно до дослідження європейського проекту інноваційних технологій cyberSANE, національна безпека країни залежить від того, чи бізнес структури, уряд або окремі громадяни мають доступ до програм та інструментів, які є високозахищеними, а також чи є здатність своєчасно виявляти та усувати кіберзагрози. Таким чином, ефективне виявлення різноманітних інцидентів безпеки, як раніше помічених, так і невідомих, та інтелектуальний захист відповідних систем від кібератак є ключовим питанням, яке потрібно терміново вирішувати.

Аналіз досліджень та публікацій

Кібербезпека – це комплекс процесів, практичних дій і технологічних рішень, які допомагають захистити важливі системи й дані від несанкціонованого доступу [3]. Вона досягається завдяки ряду традиційних рішень в сфері кібербезпеки, таких як брандмауери, аутентифікація користувачів і контроль доступу, системи шифрування тощо. Проблема полягає у тому, що інциденти безпеки зазвичай стаціонарно вирішуються декількома досвідченими аналітиками, коли управління даними виконується не системно.

Однак зі зростанням кількості інцидентів безпеки в різних формах, такі традиційні рішення зіткнулися з обмеженнями у вирішенні кіберзагроз. У результаті створюються численні комплексні атаки, які дуже швидко поширюються в Інтернеті.

Кіберзагроза - це наявні та потенційно можливі чинники, що створюють небезпеку життєво важливим функціям системи в інформаційному просторі, справляють негативний вплив на стан інформаційної безпеки організації та захист її об'єктів [4].

Інциденти безпеки - це фактичні події, що призводять до пошкодження даних чи порушення певних процесів. Водночас безпекові ризики є потенційними обставинами, що здатні призвести до порушення функціонування системи, втрати даних та інших негативних наслідків. Для запобігання ризикам важливо проводити аналіз та порівняння відповідних ризиків.

Розглянемо дві потенційні загрози такі як фішинг та несанкціонований віддалений доступ. Перша загроза базується на використанні зловмисником соціальної інженерії, коли враховується людський фактор як спосіб отримання доступу до даних, а отже пріоритет загрози буде вищим. Друга загроза базується на використанні технічних засобів або шкідливих програм для пошуку вразливостей у системі та в подальшому доступі до її ресурсів. Стратегія захисту від фішингу включатиме навчання персоналу та використання вивчених моделей ML (Machine Learning) для сканування і аналізу вмісту повідомлень, в той час як несанкціонований доступ можна попередити за допомогою встановлення та налаштування брандмауерів.

Патерни безпеки - це набір рекомендацій та система заходів по забезпеченню безпеки в інформаційних системах. Поширеним патерном є “Принцип найменшого привілею”, що дозволяє обмежувати права та доступ до ресурсів у відповідності до ролі. Ролі можуть поділятися на менеджера, адміністратора, розробника тощо. Кожна роль має свій набір дозволів та обмежень, які визначаються відповідно до рівня доступу, необхідного для виконання обов'язків цієї ролі. Ключовими елементами патернів безпеки є опис конкретної проблеми, конкретні кроки для запобігання або зменшення ризиків, поради щодо використання та приклади інтеграції в аналогічні системи безпеки.

Модель безпеки – це сукупність правил та функцій, які визначають, як система забезпечує захист від потенційних загроз і зберігає конфіденційність, цілісність та доступність своїх ресурсів. Основними компонентами моделі безпеки в контексті інформаційного простору є: **контроль доступу** (регулює доступ до мереж та систем, впроваджуючи механізм аутентифікації, авторизації та обліку), **шифрування даних** (кодування даних за допомогою алгоритмів та ключів шифрування), **моніторинг мережі** (безперервний аналіз журналів активності у мережі за допомогою системи моніторингу для виявлення аномальної поведінки) та **план реагування** (план дій на випадок інциденту безпеки для запобігання порушенню функціонування системи).

Модель безпеки, що навчена на видобутих інсайтах та безпекових патернах, може бути більш практичною. Інсайти - це висновки або уявлення, що виникають в результаті аналізу даних. Щоб вирішити проблему, потрібно розробити гнучкі та ефективні механізми безпеки, які можуть реагувати на загрози, і оновлювати патерни безпеки, щоб зменшувати негативні наслідки. Щоб досягти цієї мети, необхідно аналізувати велику кількість відповідних даних про безпеку, отриманих з різних джерел, таких як мережеві або системні джерела, та виявляти інсайти в автоматизованому режимі з мінімальним втручанням людини.

Аналіз даних та створення правильних моделей для захисту від інцидентів безпеки виходить за межі простого набору функціональних вимог і знань про загрози чи вразливості. Для ефективного навчання моделі безпеки слід використовувати методи ML, такі як маркування, кластеризація даних, класифікація або методи DL (Deep Learning) на основі нейронної мережі тощо. Ці методи ML здатні знаходити аномалії чи зловмисну поведінку, а також DD (Data-Driven) патерни безпеки, щоб прийняти інтелектуальне рішення.

Для побудови моделі безпеки можна використати фреймворк. Незважаючи на те, що існує велика кількість фреймворків для систем безпеки, вони все ж відрізняються в багатьох аспектах. В контексті кібербезпеки, фреймворк - це набір інструкцій, стандартів та патернів, розроблених для побудови моделі безпеки, її навчання та конфігурації. Ми розглянемо три широко відомих фреймворки для системи кібербезпеки, а саме: NIST, ISO 27001 та COBIT. Кожен з них зосереджений на різних сферах, має різні переваги та недоліки.

NIST – це набір стандартів, що зосереджується на вимірюванні завершеності контролю та приведенні засобів захисту кібербезпеки у відповідність до цілей організації. Основними функціями NIST є ідентифікація загроз, реагування, захист та відновлення системи [5]. Перевагами є те, що NIST побудований на попередніх версіях стандартів, доступний безкоштовно і працює з багатьма регуляторними директивами. Недоліками є відображення відомих результатів та відсутність чітких пунктів дій для протидії загрозам.

ISO 27001 – це набір стандартів, що зосереджується на створенні програм управління безпекою. Стандарт має такі функції як: розробка політики конфіденційності, контроль доступу, управління активами та організація інформаційної безпеки. Перевагами фреймворку ISO 27001 [6] є те, що він є найбільш визнаним міжнародним стандартом IT-безпеки, і більшість вимог щодо відповідності (регуляторних директив) побудовані на його основі. До недоліків можна віднести високу вартість сертифікації, відсутність посібника для інтеграції, і застарілість (не оновлювався з 2013 року).

COBIT - це набір стандартів, що зосереджується на ефективному управлінні безпеки у IT-процесах. COBIT охоплює основні IT-процеси, такі як: планування, організація, розробка, імплементація тощо. Він застосовує передові практики та підходи щодо управління цими процесами, а також інфраструктурою,

ресурсами та даними в системі безпеки [7]. Основні цілі можна розділити на чотири категорії, а саме: планування, впровадження, моніторинг та оцінка. Перевагами фреймворку є те, що він відповідає більшості регуляторних директив, зосереджений на управлінні IT процесами, а також може бути інтегрований разом з іншими стандартами. Недоліком є відсутність певних компонентів кібербезпеки, таких як контроль доступу, а також висока складність реалізації.

Вище перераховані фреймворки застосовуються у конкретних сферах для протидії кіберзагрозам та забезпеченні стабільності. Втім застарілість, відсутність чітких директив для досягнення цілей та зосередженість на конкретній області не дозволяють цим фреймворкам бути універсальним рішенням. Концепція багаторівневого фреймворку дозволить створити модель безпеки у відповідності з актуальними даними та патернами, що добуватиме чіткі патерни з класифікованого масиву даних, а застосування методів машинного навчання дозволить приймати рішення, керовані даними.

Метою дослідження є розробка концепції фреймворку для створення моделі безпеки, на основі набору інцидентів безпеки та відомих патернів. Розробка моделі безпеки на базі LSTM архітектури з допомогою багаторівневого фреймворку.

Виклад основного матеріалу

Щоб прийняти інтелектуальне рішення у вихідній системі безпеки, необхідно розуміти проблеми захисту та природу відповідних даних безпеки, вміти проводити їх широкий аналіз. З цією метою нижче описаний фреймворк не лише враховує методи машинного навчання для побудови моделі безпеки, але також бере до уваги поступове навчання та динамізм, щоб підтримувати модель в актуальному стані та генерувати своєчасну реакцію, яка може бути більш ефективною та раціональною для надання очікуваних послуг.

На рисунку 1 показано схему фреймворку, що включає кілька модулів перетворення необроблених безпекових даних, а також створення та навчання моделі для цільової системи безпеки. Далі ми коротко обговоримо принцип роботи фреймворку.

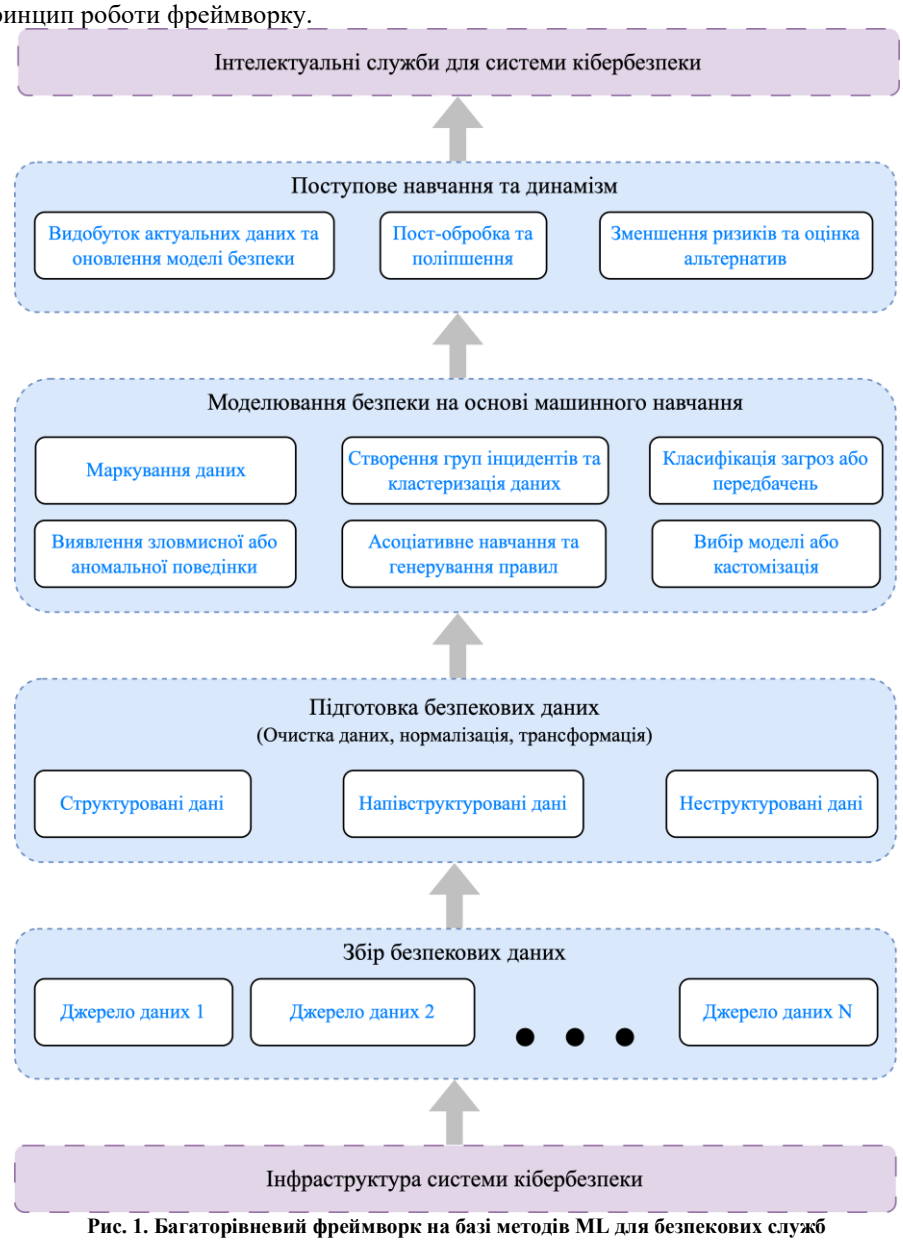

Рис. 1. Багаторівневий фреймворк на базі методів ML для безпекових служб

На першому рівні відбувається **збір цільових даних безпеки**, який формує сполучну ланку між проблемами в інфраструктурі системи безпеки та відповідними рішеннями цих проблем на основі даних. Дані безпеки - це будь-яка інформація, що стосується безпеки певної мережі, системи чи організації. Фактично такими даними слугують різноманітні журнали активності в мережі чи БД (базі даних), дані з моніторингових систем, дані про використання ресурсів тощо. Якість і кількість зібраних даних визначають можливість та ефективність вирішення проблеми безпеки відповідно до поставленої мети. Таким чином, питання полягає в тому, як зібрати цінні та унікальні дані для створення DD моделі безпеки.

Загальний етап збору та керування даними з різних джерел залежить від конкретної безпекової проблеми та проекті в межах підприємства чи організації. Джерела даних можна класифікувати за кількома категоріями, такими як мережеві, хостові і гібридні [8].

У рамках мережевої інфраструктури система безпеки може оперувати різними безпековими даними, такими як журнали систем виявлення загроз (IDS), журнали брандмауера (система захисту комп'ютерної мережі), дані мережевого трафіку, пакетні дані, honeypot дані (дані з сервісів-приманок для зловмисників) тощо. Наприклад, аналізуючи дані про IP-адреси та їх активність, можна визначити, чи є певна IP-адреса зловмисною чи ні. У галузі кібербезпеки мережеве джерело вважається основним джерелом подій безпеки для аналізу.

У категорії хостів дані збираються з комп'ютерів організації, де джерелами даних можуть бути журнали операційної системи, журнали доступу до бази даних, журнали веб-серверу, журнали електронної пошти, журнали активності додатку та інші.

Збір даних як з мережевого, так і з хостового комп'ютера вважається гібридною категорією. Загалом, на рівні збору даних мережева активність, активність бази даних, активність додатків і активність користувачів можуть бути можливими джерелами подій безпеки в контексті кібербезпеки.

Рівень **підготовки необроблених даних безпеки** з різних джерел відповідає за підготовку навчальних та тренувальних даних для майбутньої моделі. Однак не всі зібрані дані сприяють процесу побудови моделі безпеки [9]. Надлишкові дані, перехоплені за допомогою сніферів (програм, що аналізують трафік), необхідно видалити з масиву. Крім того, дані можуть бути зашумлені, мати відсутні або пошкоджені значення, містити атрибути різних типів.

Висока якість даних є необхідною умовою для досягнення високої точності в DD моделі. Точність моделі формується, коли модель навчається на основі пари вхідних-вихідних даних, де навчальний алгоритм використовує ці пари для встановлення зв'язку між ними, щоб навчити модель передбачати вихідні значення на основі вхідних. В зв'язку з цим може знадобитися процедура очищення даних, обробки відсутніх або пошкоджених значень. Крім того, характеристики або атрибути безпекових даних можуть бути різних типів, наприклад безперервні, дискретні або символьні [10]. Дискретні атрибути мають кінцеві значення, вони можуть бути числовими, а також можуть бути в категоріальній формі. Ці атрибути мають скінченний або зліченно нескінченний (величезний) набір значень. Неперервні атрибути мають плаваючий тип (значень може бути безліч від 2 до 3, наприклад 2.567…), тобто вони охоплюють певний діапазон. Символьні атрибути є якісними, тобто на відміну від числових, вони можуть описувати властивості даних (розмір, колір тощо). Окрім чіткого розуміння типів даних, атрибутів та допустимих операцій з ними, необхідно попередня обробка даних та їх атрибутів для перетворення в цільовий тип.

Необроблені дані також можуть бути різних видів, наприклад структуровані, напівструктуровані, неструктуровані тощо. Якщо структуровані дані - це вже готове джерело кількісних фактів чи спостережень, то неструктуровані дані - це інформація, яку ще потрібно класифікувати певним чином. Напівструктуровані дані - це дані, які є комбінацією обох попередніх типів, тобто їх не можна категоризувати звичним способом. Зазвичай такі дані мають певні сталі характеристики або властивості: наприклад, містять теги (інформація, що описує зміст даних), які можна проаналізувати. Таким чином, нормалізація та перетворення можуть бути корисними операціями для організації даних у структурованому вигляді. У деяких випадках можуть бути корисними методи обробки людської мови, в залежності від типу і характеристик даних, наприклад, текстового вмісту.

Оскільки якість та кількість даних визначають можливість вирішення проблеми безпеки, ефективна попередня обробка та управління даними можуть відіграти важливу роль у створенні ефективної моделі безпеки для безпекових сервісів.

Рівень **моделювання безпеки на основі машинного навчання** - це основний етап, на якому з даних видобуваються висновки (інсайти) та знання, що готує фундамент для навчання моделі. Розглянемо декілька пов'язаних модулів.

Модуль *маркування безпекових даних* в основному відповідає за перетворення необроблених даних в інформативні ознаки, які ефективно представляють проблему безпеки для DD моделей. Маркування даних – це процес ідентифікації необроблених даних (зображень, текстових файлів) та додавання однієї або кількох значущих та інформативних міток для забезпечення контексту, щоб ML модель могла на них навчатися. Наприклад, мітки можуть вказувати, чи є на фотографії дерево чи автомобіль, які слова використовуються в текстовому файлі тощо.

В цей модуль може бути залучено декілька методів обробки даних, таких як нормалізація ознак , вибір ознак з урахуванням підмножини доступних ознак відповідно до їх кореляцій або важливості в моделюванні[11]. Кожна нормалізована ознака визначається за формулою:

$$x_{i,norm} = \frac{x_i - x_{min}}{xmin_{max}}$$

де $x_{i,norm}$ - нормалізований елемент ознаки, x_{min} - найменший елемент ознаки, x_{max}.- найбільший елемент, x_i.- ненормалізований елемент. Нормалізація даних дозволить представити дані у вигляді набору значень в діапазоні від 0 до 1, що спростить процес навчання.

Іншим важливим модулем є *кластеризація даних безпеки*, яка включає групування даних безпеки зі схожими характеристиками, які можна використовувати для вирішення кількох проблем кібербезпеки, таких як виявлення аномалій, порушення політики конфіденційності тощо. Елементи всередині кластера повинні бути схожими один на одного, та відмінними від будь-яких елементів, що увійшли у інші кластери.

Кластеризація відбувається на основі алгоритму k-середніх , якщо кількість кластерів відома, або g-середніх, коли кількість кластерів визначається автоматично. В основі роботи алгоритму лежить створення центрів кластерів, навколо яких будуть розташовуватися елементи. Центри кластерів можуть створюватися довільно, або в конкретній точці, після чого близько розташовані елементи будуть формувати кластер. На кожній ітерації, позиція центру кластера буде розраховуватися як середнє арифметичне всіх елементів, що належать кластеру. Процес оновлення центрів кластерів повторюється поки кластери не стабілізуються, тобто поки зміни у належності елементів до кластерів і центрів кластерів не стануть мінімальними або зміни від однієї ітерації до іншої незначними. Процес кластеризації дозволить сформувати розділені набори елементів за чіткими критеріями, що спростить їх аналіз та використання.

Модуль *виявлення аномальної поведінки*, як правило, відповідає за виявлення відхилень від нормальної поведінки (девіації), при цьому для виявлення зловмисної поведінки або аномалій також можуть використовуватися методи аналізу на основі кластеризації. Наприклад існує декілька кластерів з центрами, та набором елементів навколо центрів. Чим ближче елемент кластеру знаходиться до центру, тим більш типовим цей елемент є для кластеру. Відповідно чим далі - тим він є менш типовим. Задавши порогове значення відстані для елементів кластеру, тобто створивши межі в яких елементи будуть вважатися типовими, ми помітимо, що деякі елементи перевищують дану відстань. Такі елементи є аномальними або підозрілими.

У сфері кібербезпеки *класифікація або прогнозування атак* розглядається як один із найважливіших модулів, який відповідає за створення моделі прогнозування на основі методів ML, а саме класифікації та регресії.

Метод класифікації - це алгоритм, який використовується для присвоєння об'єктам або елементам однієї з попередньо визначених категорій на основі їх характеристик або ознак. Прикладом використання цього методу є вбудований у більшість поштових клієнтів фільтр спаму. Застосувавши алгоритм Баєса, ми перевіримо кожен елемент повідомлення та визначимо чи воно належить до спам-вибірки. Якщо кількість елементів, що відмічені як загрозливі, перевищує порогове значення, то повідомлення буде додано до відповідної спам-категорії. Тобто результатом виконання буде розміщення об’єкта або елемента в певному класі або категорії. Для вирішення подібних задач класифікації також застосовується дерево рішень, метод опорних векторів, нейронні мережі тощо.

Метод регресії - це алгоритм, який описує залежність між двома змінами для того, аби прогнозувати значення для нових наборів вхідних даних. Головною відмінністю від попереднього методу є те що, в результаті ми отримуємо дійсне безперервне число. Для вирішення задач регресії застосовується лінійна регресія, дерево регресії та нейронні мережі.

Модуль *вивчення асоціацій та генерації правил* може зіграти роль у створенні системи безпеки, яка містить кілька правил IF-THEN які визначають тактику протидії атакам. Ці правила є звичайною перевіркою умови - якщо вона задовільнена, то виконується певний програмний алгоритм (послідовність дій). Метою даного алгоритму, є пошук відношень між даними. Таким чином, цей модуль дозволить виявляти взаємозв'язки або асоціації між наявними ознаками у заданому наборі даних.

Рівень **поступового навчання та динамізму** пов’язаний із завершенням формування підсумкової моделі безпеки шляхом включення додаткових можливостей відповідно до потреб. Це можливо шляхом подальшої обробки даних в кількох модулях.

Наприклад, модуль *пост-обробки та вдосконалення* відповідає за спрощення видобутих знань згідно з конкретними вимогами, включаючи лише знання, притаманні специфічній галузі. Оскільки моделі класифікації або прогнозування атак, засновані на методах ML, сильно залежні від навчальних даних, їх навряд чи можна узагальнити, що може бути важливим для деяких випадків. Щоб усунути такі обмеження, цей модуль використовує знання про галузь у вигляді таксономії (об’єднання в групу на основі притаманних ознак) чи онтології (області знань) для покращення кореляції (співвідношення або залежності) загроз у системі безпеки.

Модуль *оновлення моделі безпеки* залучає найновіші DD патерни безпеки для підвищення продуктивності. Добуті знання, розглянуті на попередньому рівні, ґрунтуються на статичному початковому масиві даних з урахуванням загальних закономірностей в наборах даних. У багатьох випадках такі дані можуть містити різні закономірності, які суперечитимуть існуючим знанням. В такому випадку можна застосувати концепцію Recency Miner [12], що вилучає нові безпекові закономірності та правила, які мають більше шансів бути значущими, для прогнозування ризиків та загроз. Замість обробки всіх безпекових даних

знову, динамічне оновлення даних на основі рекурентної нейронної мережі дозволить видалити застарілі дані та додати нові. Це може зробити кінцеву безпекову модель динамічною.

Модулі, розглянуті вище, можуть бути застосовані як окремо, так і разом залежно від цільових проблем безпеки.

Наведемо приклад застосування багаторівневого фреймворку для створення моделі безпеки. На першому етапі ми використаємо SIEM (Security information and event management) систему, як джерело даних. Ця система дозволяє обробляти великі обсяги журналів, аудиторських записів, мережевого трафіку та іншої інформації для виявлення аномальних подій та інцидентів безпеки.

Наступним кроком буде підготовка масиву даних для навчання, що буде складатися з журналів мережевої активності усіх користувачів в рамках організації. Масив даних буде розділений на два набори: навчальний та тестувальний. З метою покращення якості вихідної моделі буде залучено структуровані та марковані дані зловмисної мережевої діяльності, такі як Stuxnet, Havex та TRISIS / TRITON. Таким чином модель зможе розпізнавати подібну аномальну поведінку, що вказуватиме на потенційну загрозу.

Модель оцінюється на основі тестових даних, а потім генерує рішення на основі нових даних. Якщо буде виявлена аномальна поведінка, то система запустить відповідні заходи захисту для протидії загрозі. Дані для навчання будуть організовані у вигляді часових рядів. Часовий ряд - це послідовність даних, що збираються або вимірюються у різні моменти часу. Кожне значення в часовому ряді відображає певну величину або властивість в певний момент часу. Часові ряди структуровані у хронологічному порядку.

На етапі моделювання ми використаємо рекурентну нейронну мережу (RNN) як метод DL для побудови моделі безпеки. Рекурентна нейронна мережа - це гілка нейронних мереж, що містить приховані вузли. Кожен вузол використовує вихідні дані попереднього вузла як свої вхідні дані. Таким чином інформація циркулює між вузлами в мережі. Основною метою RNN є обробка часових рядів і аналіз потоків даних. RNN володіє пам'яттю, що означає, що вона зберігає інформацію з попереднього досвіду і пізніше використовує її як вхідні дані для наступних вузлів [13].

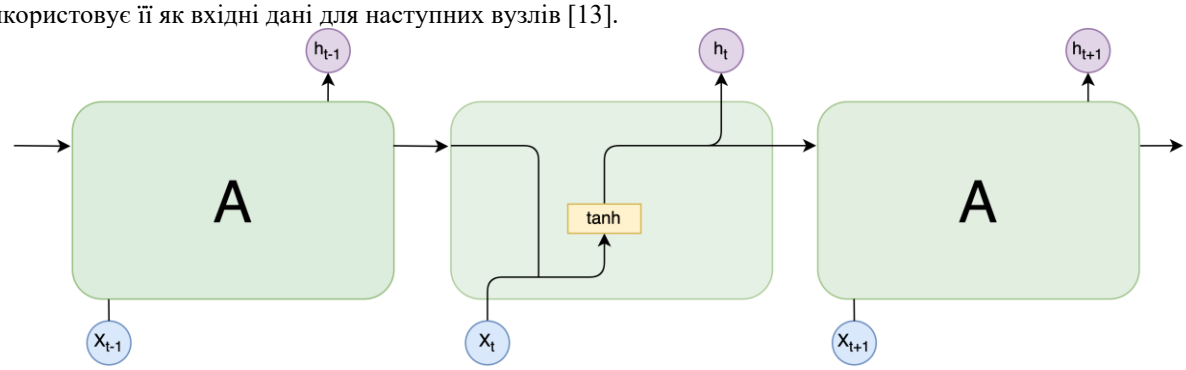

Рис. 2. Архітектура рекурентної мережі з одним шаром нейронної мережі

На рисунку 2 зображено базову рекурентну мережу, що складається з ланцюжку вузлів, які передають вихідні дані у наступний вузол. Символ x означає елемент послідовності, що надходить у якості вхідних даних у вузол, t - символізує порядковий номер ітерації або вузла, а h - це вихідні дані. Кожен вузол містить один шар нейронної мережі, що має функцію активації - гіперболічний тангенс. Функція активації гіперболічного тангенсу приймає вхідні значення і перетворює їх за формулою:

$$tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

де x - вхідне значення, e - число Ейлера. Гіперболічний тангенс на виході повертає значення в діапазоні від -1 до 1, тому він використовується для нелінійного перетворення вхідних значень, щоб зберегти або видобути корисну інформацію для наступних шарів мережі.

Базова рекурентна мережа має суттєвий недолік, а саме обмежена тривалість запам’ятовування попередньої інформації, через що ускладнюється процес навчання рекурентної мережі. З метою виправлення цього недоліку, німецькі вчені З. Хохрайтер та Ю. Шмідгубер створили так звану LSTM (Long Short-Term Memory) архітектуру, в основі якої лежить принцип запам’ятовування інформації з попередніх вузлів. LSTM архітектура організована достатньо складно, адже для побудови одного вузла використовується чотири шари нейронних мереж.

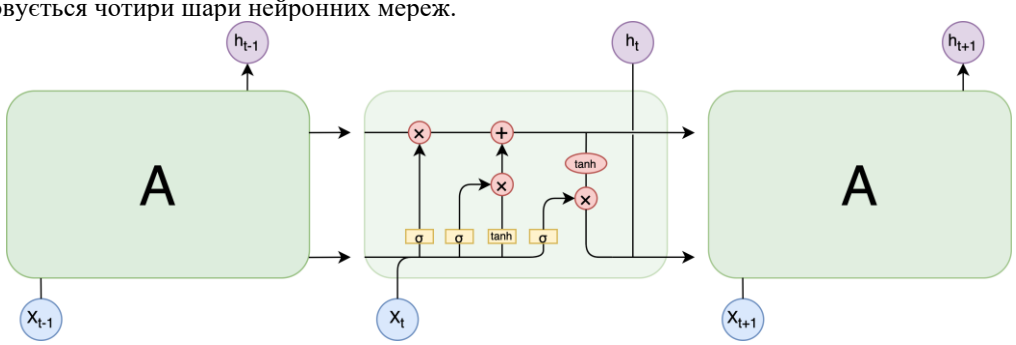

Рис. 3. Архітектура LSTM рекурентної мережі з чотирма шарами нейронної мережі

На рисунку 3 кожна лінія містить вектор, від виходу одного вузла до входу інших вузлів. Рожеві кружечки означають операції, а саме поелементне додавання та множення векторів. Жовті прямокутники в свою чергу відображають шари нейронної мережі та функцію активації. Окрім гіперболічного тангенсу, мережа також використовує сигмоїд. Функція активації сигмоїд приймає вхідні значення і перетворює їх за формулою:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

де x - вхідні дані, e - число Ейлера. Сигмоїд повертає значення в діапазоні від 0 до 1, тому він використовується для вирішення задач бінарної класифікації та регресії.

Вектори, що об'єднуються означають конкатенацію, або об'єднання, тоді як розгалуження означає, що вміст вектору копіюється, а копії спрямовуються в різні місця.

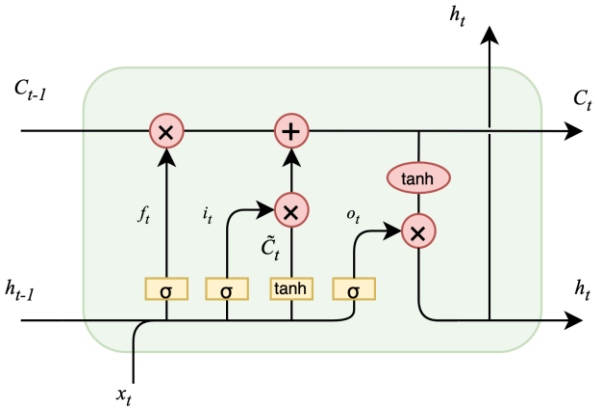

Рис. 4. Архітектура вузла в LSTM

На рисунку 4 зображено архітектуру одного вузла. Ключовим елементом вузла є горизонтальна лінія, що проходить через верхню частину діаграми, так званий стан вузла C_t, що схожий на конвеєр. Він проходить по всьому ланцюжку, лише з деякими незначними лінійними взаємодіями. Інформація може проходити по лінії без змін. Мережа може видаляти та додавати інформацію до стану вузла, що регулюється структурами, які називаються вентилі. В LSTM використовується три типи вентилів: вхідний, вихідний та забуття [13].

Перший крок це вирішення, яку інформацію слід видалити з стану вузла C_t. За це відповідає вентиль забуття f_t, що складається з нейронного шару з функцією активації сигмоїд. Значення вираховується за формулою:

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right)$$

де W - це вагові коефіцієнти, b - зміщення вектору. Цей вентиль видасть значення від 0 до 1 для кожного компонента стану вузла C_t. Після чого відбудеться операція поелементного множення і компоненти, що отримають значення 0 - будуть видалені з стану вузла C.

Наступний крок складатиметься з двох частин. Вхідний вентиль i_t відповідає за оновлення компонентів з попереднього вузла. Він використовує нейронний шар з функцією активації сигмоїд. Також шар гіперболічного тангенсу утворює новий вектор $\tilde{C}_t$ що містить нові значення-кандидати, які можуть бути додані до стану вузла C_t. Значення вхідного вентилю та нового вектору розраховуються за формулами:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$\tilde{C}_t = tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

де W - це вагові коефіцієнти, b - зміщення вектору.

Тепер слід замінити попередній стан вузла C_{t-1} на новий C_t. Обчислення нового стану визначається за формулою:

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t$$

Ми множимо попередній стан на f_t, тим самим забуваючи те, що ми вирішили забути, потім додаємо $i_t \cdot \tilde{C}_t$. Це нові значення-кандидати помножені на t - наскільки ми хочемо оновити кожне із значень стану.

Останнім етапом буде обрахунок вихідних значень за допомогою вихідного вентилю O_t. Значення буде вирахувано за формулою:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t \cdot tanh(C_t)$$

де W - це вагові коефіцієнти, b - зміщення вектору. Цей результат базуватиметься на стані вузла, до якого будуть застосовані фільтри. Спочатку ми запускаємо сигмоїдальний шар, який вирішує, яку інформацію зі стану вузла ми збираємося вивести. Потім значення стану вузла проходять через шар

гіперболічного тангенсу (щоб отримати значення в діапазоні між −1 і 1 на виході) і множаться на вихідні значення сигмоїдального шару, що дозволяє вивести лише необхідну інформацію.

Таким чином унікальна архітектура LSTM дозволяє зберігати нещодавню та минулу інформацію, дозволяючи моделі визначати які дані є релевантними. Це робить дану архітектуру придатною для обробки даних у вигляді часових рядів та дозволяє виконувати завдання прогнозування (регресії) та класифікації.

Модель безпеки на базі LSTM, дозволить відстежувати мережеву активність на наявність ознак аномальної поведінки, що вказують на потенційні вторгнення. Якщо такі ознаки перевищуватимуть визначені порогові значення аномалії, система автоматично запустить сповіщення та ініціює заходи захисту, такі як блокування підозрілих IP-адрес або відключення пов'язаних облікових записів користувачів. Ці можливості забезпечують швидке програмне реагування на виявлені загрози.

Висновки

Було розглянуто багаторівневий фреймворк для створення моделі безпеки, заснований на методах машинного навчання. Було описано структуру фреймворку, що складається з декількох основних етапів - збір даних про безпеку, підготовка даних, моделювання безпеки на основі машинного навчання, а також поступове навчання і динамізм. На основі фреймворку розроблено модель безпеки на базі LSTM архітектури.

Література

1. Malware Statistics & Trends Report | AV-TEST. https://www.av-test.org/en/statistics/malware/

2. Security & Identity | Juniper Research. https://www.juniperresearch.com/research/security-identity

3. Abomhara M., Geir M. Køien Cyber Security and the Internet of Things: Vulnerabilities, Threats, Intruders and Attacks. Journal of Cyber Security and Mobility. 2015. Vol. 4, no. 1. P. 65–88. DOI: 10.13052/jcsm2245-1439.414

4. Aftergood S. Cybersecurity: The cold war online. Nature. 2017. Vol. 547, no. 7661. P. 30–31. DOI: 10.1038/547030a

5. Framework for Improving Critical Infrastructure Cybersecurity, Version 1.1. Gaithersburg, MD: National Institute of Standards and Technology, 2018. DOI: 10.6028/nist.cswp.04162018

6. Stefan Fenz. Information Security Fortification by Ontological Mapping of the ISO/IEC 27001 Standard. 13th Pacific Rim International Symposium on Dependable Computing (PRDC 2007), Melbourne, Australia, 17–19 December 2007. [S. l.], 2007. DOI: 10.1109/prdc.2007.29

7. Steven De Haes COBIT as a Framework for Enterprise Governance of IT. Management for Professionals. Cham, 2019. P. 125–162. DOI: 10.1007/978-3-030-25918-1_5

8. Ullah F., Muhammad Ali Babar Architectural Tactics for Big Data Cybersecurity Analytics Systems: A Review. Journal of Systems and Software. 2019. Vol. 151. P. 81–118. DOI: 10.1016/j.jss.2019.01.051

9. Shuai Zhao I-CaN-MaMa: Integrated campus network monitoring and management. NOMS 2014 - 2014 IEEE/IFIP Network Operations and Management Symposium, Krakow, Poland, 5–9 May 2014. [S. l.], 2014 DOI: 10.1109/noms.2014.6838304

10. Iqbal H. Sarker IntruDTree: A Machine Learning Based Cyber Security Intrusion Detection Model. Symmetry. 2020. Vol. 12, no. 5. P. 754. DOI: 10.3390/sym12050754

11. Romero C., Sebastian Ventura Data mining in education. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. 2012. Vol. 3, no. 1. P. 12–27. DOI: 10.1002/widm.1075

12. Sarker I. H., Alan Colman, Jun Han RecencyMiner: mining recency-based personalized behavior from contextual smartphone data. Journal of Big Data. 2019. Vol. 6, no. 1. DOI: 10.1186/s40537-019-0211-6

13. Yong Yu A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. Neural Computation. 2019. Vol. 31, no. 7. P. 1235–1270. DOI: 10.1162/neco_a_01199