

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет комп'ютерної інженерії та управління
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

Рівень вищої освіти другий (магістерський)

Метод виявлення вторгнень
у веб-сеанси користувача

(тема)

Виконав:

студент II курсу, групи СПМ-21-1
Запорожець Н.О.
(прізвище, ініціали)

Спеціальність 123 «Комп'ютерна інженерія»
(код і повна назва спеціальності)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Системне програмування
(повна назва освітньої програми)

Керівник: доц. Мартовицький В.О.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ЕОМ

(підпис)

Коваленко А.А.

(прізвище, ініціали)

2022 р.

Харківський національний університет радіоелектроніки

Факультет _____ комп'ютерної інженерії та управління _____

Кафедра _____ електронних обчислювальних машин _____

Рівень вищої освіти _____ другий (магістерський) _____

Спеціальність _____ 123 «Комп'ютерна інженерія» _____
(код і повна назва)

Тип програми _____ освітньо-професійна _____
(освітньо-професійна або освітньо-наукова)

Освітня програма _____ Системне програмування _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

“ _____ ” _____ 20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студенту _____ Запорожець Наталії Олегівні _____
(прізвище, ім'я, по батькові)

1. Тема роботи Метод виявлення вторгнень у веб-сеанси користувача

затверджена наказом по університету від “ 07 ” листопада 2022 р. № 1454 Ст

2. Термін подання студентом роботи до екзаменаційної комісії _____ 13 грудня 2022 р.

3. Вхідні дані до роботи 1) протокол веб-сеансу користувача; 2) методи машинного навчання моделі: логістична регресія, алгоритм к найближчих сусідів, випадковий ліс, метод опорних векторів

4. Перелік питань, що потрібно опрацювати у роботі _____

1) аналіз літературних даних та постановка проблеми;

2) виявлення та попередження вторгнень в роботу інформаційної комп'ютерної системи;

3) методи машинного навчання та їх використання для ідентифікації користувачів у системі за їх поведінкою;

4) розробка моделі ідентифікації користувача системи;

5) дослідження запропонованого методу виявлення вторгнень у веб-сеанси користувача;

6) висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) _____

Слайд-презентація – 12 слайдів. _____

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Огляд методів виявлення вторгнень у веб-сеанси користувача	08.11.22–11.11.22	
2	Вибір та обґрунтування методики дослідження	12.11.22–17.11.22	
3	Розробка моделі ідентифікації користувача	18.11.22–21.11.22	
4	Вибір методів машинного навчання моделі	22.11.22–28.11.22	
5	Проведення експериментальних досліджень	29.11.22–02.12.22	
6	Оформлення матеріалів кваліфікаційної роботи	03.12.22–06.12.22	
7	Подання кваліфікаційної роботи керівникові та її попередній захист	07.12.22–08.12.22	
8	Подання кваліфікаційної роботи на рецензування	09.12.22–12.12.22	

Дата видачі завдання 07 листопада 2022 р.

Студент _____
(підпис)

Керівник роботи _____
(підпис)

доц. Мартовицький В.О.
(посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 77 с., 13 рис., 2 табл., 1 дод., 27 джерел.

ВИЯВЛЕННЯ ВТОРГНЕНЬ, МАШИННЕ НАВЧАННЯ, МОДЕЛЬ, ІДЕНТИФІКАЦІЯ, КОРИСТУВАЧ, КЛАСИФІКАЦІЯ, РЕГРЕСІЯ, IDS, IPS.

Метою кваліфікаційної роботи є розробка методів та засобів виявлення вторгнень, які б давали змогу ідентифікувати користувачів системи за їх поведінкою.

У ході виконання кваліфікаційної роботи здійснено огляд літературних джерел, присвячених темі дослідження, та проаналізовано існуючі підходи до вирішення проблеми. Розглянуто особливості та класифікацію систем виявлення та попередження вторгнень у роботу комп'ютерних систем. Виконано огляд методів машинного навчання та можливості їх застосування для ідентифікації користувачів системи за їх поведінкою.

У роботі запропоновано модель процесу забезпечення ідентифікації користувачів за їх поведінкою в системі, що дозволяє створити додаткові засоби захисту користувачів системи у випадку крадіжки їх даних автентифікації. Модель ідентифікації враховує статистичні параметри поведінки користувача, які були отримані впродовж сеансу.

Проведено експериментальне дослідження запропонованого підходу ідентифікації користувача за його поведінкою в системі. Побудовані моделі поведінки користувача з використанням методів машинного навчання показали оцінку якості ідентифікації більше 0,95.

ABSTRACT

Master's thesis: 77 pages, 13 figures, 2 tables, 1 appendices, 27 sources.

INTRUSION DETECTION, MACHINE LEARNING, MODEL, IDENTIFICATION, USER, CLASSIFICATION, REGRESSION, IDS, IPS.

The major goal of this thesis is to develop methods and means of detecting intrusions that would make it possible to identify system users based on their behavior.

In the course of the qualification work, a review of literary sources dedicated to the research topic was carried out, and existing approaches to solving the problem were analyzed. Features and classification of systems for detecting and preventing intrusions into the operation of computer systems are considered. An overview of machine learning methods and the possibility of their application to identify system users based on their behavior was performed.

The paper proposes a model of the process of ensuring the identification of users based on their behavior in the system, which allows creating additional means of protecting system users in case of theft of their authentication data. The identification model takes into account the statistical parameters of the user's behavior, which were obtained during the session.

An experimental study of the proposed approach of user identification based on his behavior in the system was conducted. Constructed models of user behavior using machine learning methods showed an identification quality rating of more than 0.95.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	8
ВСТУП	9
1 АНАЛІЗ ЛІТЕРАТУРНИХ ДАНИХ ТА ПОСТАНОВКА ПРОБЛЕМИ	12
2 ВИЯВЛЕННЯ ТА ПОПЕРЕДЖЕННЯ ВТОРГНЕНЬ В РОБОТУ ІНФОРМАЦІЙНОЇ КОМП'ЮТЕРНОЇ СИСТЕМИ	18
2.1 Системи виявлення і попередження вторгнень та атак	18
2.2 Системи виявлення вторгнень IDS.....	19
2.2.1 Класифікація IDS	20
2.2.2 Способи виявлення вторгнень	21
2.3 Системи попередження вторгнень IPS	22
2.3.1 Класифікація IDS	22
3 МЕТОДИ МАШИННОГО НАВЧАННЯ ТА ЇХ ВИКОРИСТАННЯ ДЛЯ ІДЕНТИФІКАЦІЇ КОРИСТУВАЧІВ У СИСТЕМІ ЗА ЇХ ПОВЕДІНКОЮ	24
3.1 Загальні відомості про машинне навчання.....	24
3.2 Способи машинного навчання.....	25
3.2.1 Навчання з учителем.....	26
3.2.2 Навчання без учителя	28
3.3 Життєвий цикл проекту машинного навчання	30
4 РОЗРОБКА МОДЕЛІ ІДЕНТИФІКАЦІЇ КОРИСТУВАЧА СИСТЕМИ	32
5 ДОСЛІДЖЕННЯ ЗАПРОПОНОВАНОГО МЕТОДУ ВИЯВЛЕННЯ ВТОРГНЕНЬ У ВЕБ-СЕАНСИ КОРИСТУВАЧА	38
5.1 Функціональна модель процесу побудови профілю поведінки користувачів та забезпечення на його основі ідентифікації.....	38
5.2 Експериментальне дослідження підходу до ідентифікації користувачів за їх поведінкою в системі	42

5.3 Аналіз результатів дослідження підходу до ідентифікації користувачів за їх поведінкою в системі	48
ВИСНОВКИ.....	53
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	55
ДОДАТОК А Графічний матеріал кваліфікаційної роботи.....	58
ДОДАТОК Б Код програми	65

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ
І ТЕРМІНІВ

EAI – інтеграція корпоративних додатків (англ., Enterprise Application Integration)

IDS – система виявлення вторгнень (англ., Intrusion Detection System)

IPS – система попередження вторгнень (англ., Intrusion Prevention System)

IT – інформаційні технології (англ., Information Technology)

ITRC – Міждержавна технологічна та регуляторна рада (англ., Interstate Technology and Regulatory Council)

MFA – багатофакторна автентифікація (англ., Multi-Factor Authentication)

REST – передача репрезентативного стану (англ., Representational State Transfer)

SLA – угода про рівень послуг (англ., Service-Level Agreement)

SOA – сервіс-орієнтована архітектура (англ., Service-Oriented Architecture)

SSO – технологія єдиного входу (англ., Single Sign-On)

ВСТУП

З розвитком обчислювальних можливостей, все більшої популярності соціальних мереж та використанням різних Web-сервісів сучасний бізнес здійснює цифровізацію своїх активів. Також на перехід компаній в Інтернет простір спонукала і глобальна пандемія коронавірусу. Таким чином, щоб залишатися конкурентоспроможними та розвиватись, компаніям потрібно змінювати підхід до організації своєї роботи та комунікації між усіма учасниками. Одним з таких підходів є використання хмарних технологій.

Хмарні технології – один з основних інструментів діджиталізації, без якої складно уявити не лише розвиток бізнесу будь-яких масштабів, а й життя сучасної людини. Зараз хмарні технології частково або повністю використовуються у різних галузях бізнесу. Причому ці підходи мають для компаній набагато більше значення, ніж може здатися на перший погляд.

Технологічно розвиненим компаніям, в яких працюють розподілені команди, хмарні рішення дозволяють організувати сучасні механізми віддаленої роботи. А середнім і невеликим компаніям, у яких недостатньо ресурсів для побудови власної якісної інфраструктури та серверних потужностей, дають можливість делегувати «хмарі» частину своєї роботи.

Для забезпечення безпеки використання хмарних технологій слід дотримуватися наступних правил:

- захист конфіденційної інформації – це зона відповідальності самої компанії на всіх рівнях, від керівника до рядових співробітників;

- шифрування трафіку, зазвичай достатньо використовувати SSL/TSL, але обов'язково потрібно враховувати актуальність сертифікатів. Можна використовувати VPN, тим самим практично повністю гарантувати безпеку трафіку при переміщенні незахищеним каналом;

- чітке прописування параметрів інфраструктури SLA та їх перевірка. У SLA описуються умови надання послуг (сервісів), встановлюється перелік

таких послуг, а також правила, за якими замовник користуватиметься цими сервісами. У той же час SLA є одним з основних механізмів, що дозволяють керувати якістю IT-послуг.

Однак, як би компанії не дотримувались цих рекомендацій і як би не були захищені вендори, які надають їм послуги, все одно залишається ризик злому системи. І основним слабким місцем в системі завжди були і залишаються самі користувачі [1].

У 2021 році кіберзловмисники все ще користуються ситуацією, що склалася навколо пандемії COVID-19. Механізми віддаленої роботи, ізоляція працівників та поточна ситуація з вакцинацією підвищують інтерес кіберзлочинців до методів соціальної інженерії. І таким чином зловмисники отримують легітимні данні від службових акаунтів користувачів, за допомогою яких потім здійснюють крадіжку даних або обхід систем безпеки підприємства.

Звіт ITRC про наслідки для бізнесу за 2021 рік показує, що більше половини малих підприємств постраждали від злому даних чи порушення безпеки, третина компаній були зламани не менше трьох разів [2].

ITRC запросила інформацію про вплив кібератак на компанію безпосередньо у власників та керівників малого бізнесу, які постраждали від порушень безпеки та даних. У звіті Business Aftermath Report за 2021 рік 417 керівників малого бізнесу та 1050 звичайних споживачів відповіли на питання у двох окремих опитуваннях щодо впливу кіберзлочинів на малий бізнес. Висновки включають наступні факти:

- 58 % малих підприємств зіткнулися з витоком даних, порушенням безпеки або тим і іншим. Три чверті цих підприємств зіткнулися як мінімум із двома зломами, а одна третина – як мінімум із трьома зломами;

- 44 % малих підприємств витратили від 250 000 до 500 000 доларів США на покриття витрат, пов'язаних із витоком даних. 16 % малих підприємств витратили від 500 000 до 1 мільйона доларів;

- 36 % малих підприємств взяли кредити, щоб покрити витрати на

порушення безпеки, а 34 % використали грошові резерви;

- 15 % скоротили чисельність персоналу, щоб скоротити витрати;

- суб'єкти зовнішніх загроз відповідальні за 40 % атак. Зловмисники та підрядники несуть відповідальність за 35 % атак.

Ще один важливий висновок, представлений у звіті [2] про наслідки для бізнесу за 2021 рік, полягає в тому, що 42 % малих підприємств потрібно від одного до двох років, щоб повернутися до нормального функціонування. 28 % респондентів кажуть, що їхньому бізнесу потрібно від трьох до п'яти років, щоб повністю відновитися.

Однією з найбільших причин, які призвели до порушень безпеки сервісів компаній – це отримання доступу зловмисником до легітимних облікових записів користувачів системи. На жаль, боротися з цим майже не можливо, оскільки зловмисник авторизований, як легітимний користувач, що робить системи виявлення вторгнень неефективними.

Таким чином актуальною стає задача розробки методів та засобів захисту (виявлення вторгнення), які б давали змогу ідентифікувати користувачів системи за їх поведінкою. Це ні в якому разі не захистить від крадіжки даних облікових записів користувачів системи, але дасть змогу протидіяти зловмисникам у випадках, коли вони використовують цей обліковий запис для подальшого злону системи.

1 АНАЛІЗ ЛІТЕРАТУРНИХ ДАНИХ ТА ПОСТАНОВКА ПРОБЛЕМИ

Кібербезпека була і залишається одним з пріоритетних напрямків розвитку багатьох країн. Зростання досліджень, пов'язаних з кібербезпекою, наочно простежується сьогодні, у зв'язку зі зростанням кількості кіберзлочинів та випадків кібертероризму. Хакерські атаки реєструються у всіх куточках світу. Серед найбільш резонансних варто згадати поширення вірусів WannaCry [3], Petya/NotPetya [4], які завдали значної шкоди банківським системам і великим компаніям різних країн.

Автори статті [5] провели дослідження процесів забезпечення захисту Web-застосунка від атак, спрямованих на отримання несанкціонованого доступу до функцій адміністратора системи управління контентом. В результаті було представлено метод вибору заходів захисту Web-застосунка, який заснований на методі оцінювання показника успішності атаки. Оскільки всі заходи захисту відрізняються вартістю, ефективністю і впливом на різні вектори атак, в результаті вибору визначається набір контрзаходів, який надає максимальне зниження показника успішності атаки. Тому до зміни набору контрзаходів призводить не тільки зміна їх параметрів, а й зміна параметрів дерева атак. Задача вибору заходів захисту є нелінійною задачею цілочисельного програмування з булевими змінними [5].

У відповідь на зростання кількості вразливостей організацій до витоків даних в статті [6] представлено інтегровану модель ризику управління витокими даними, засновану на систематичному огляді літератури. Теоретичне дослідження розширює сукупність знань про управління витокими даними за рахунок виявлення та оновлення концептуальних уявлень про ризики витоку даних та дозволів (дій), а також за рахунок забезпечення основи для реагування організацій на інциденти витоку даних (евристика). На практиці дослідження дає ключову інформацію, яку фахівці із захисту інформації можуть використовувати для організації ефективного управління витокими

даних на основі всебічних профілів елементів ризику та методів усунення.

Технологія хмарних обчислень забезпечує доступ до пулу конфігурованих ресурсів, включаючи простір для зберігання даних, програми, послуги та мережу на вимогу. Використання хмарних технологій в організації зводить до мінімуму зусилля цієї організації для задоволення потреб своїх клієнтів. Однією з основних переваг хмарних обчислень є метод єдиного входу (SSO), який дозволяє користувачеві отримувати доступ до кількох служб застосувань, використовуючи єдині облікові дані користувача. У хмарних обчисленнях є багато питань та проблем, які необхідно обговорити. Однак запобігання атакам на систему безпеки є набагато складнішим при збереженні конфіденційності користувачів агентів. У статті [7] пропонується архітектура біометричної автентифікації на основі SSO для служб хмарних обчислень для подолання атак на безпеку та конфіденційність. Біометрична автентифікація ефективна для ресурсів, контрольованих кінцевими пристроями під час доступу до хмарних сервісів, оскільки ці пристрої неефективні в обчислювальному відношенні для обробки інформації користувача під час автентифікації. Відповідно, за допомогою запропонованої архітектури біометричної автентифікації зводиться до мінімуму атака на безпеку у хмарних обчисленнях. Запропонована архітектура також включає новий підхід, в якому існують стосунки один до одного між агентом користувача і постачальником послуг. У ньому користувачі агенти можуть використовувати свій відбиток пальця при запиті реєстрації та доступі до різних служб хмарних додатків у хмарі. На основі порівняльного дослідження з кількома існуючими архітектурами було представлено основні моменти запропонованої архітектури. Але такий підхід потребує використання додаткового обладнання, що не завжди є доцільним при розробці тих чи інших продуктів компанії, які опираються на технології хмарних обчислень.

У сучасну епоху корпоративних обчислень інтеграція корпоративних додатків (EAI) є добре відомим та визнаним у галузі архітектурним

принципом, заснованим на слабозв'язній архітектурі застосувань, де сервіс-орієнтована архітектура (SOA) є архітектурним шаблоном для реалізації. Хоча SOA може бути реалізована в широкому спектрі технологій, реалізація SOA через веб-служби стає популярним вибором через її простоту, що оснований на основних інтернет-протоколах. Технологія веб-сервісів визначає кілька підтримуючих протоколів та специфікацій, таких як SOAP та WSDL, для зв'язку з клієнтом та сервером для обміну даними. В 2000-х роках в SOA з'явилася нова архітектурна парадигма під назвою REpresentational State Transfer (REST), яка також використовується консорціумами системної інтеграції для інтеграції слабо пов'язаних сервісних компонентів, які називають веб-сервісами RESTful. Ця реалізація SOA не містить адекватних рішень безпеки, і її безпека повністю залежить від безпеки мережевого/транспортного рівня, яка застаріла через новітні веб-технології, такі як Web 2.0 та його оновлена версія Web 3.0. Продукти безпеки постачальників мають серйозні обмеження реалізації, такі як необхідність захищеного організаційного середовища та порушення специфікацій SOA, що призводить до появи нових вразливостей.

Тому в статті [8] пропонується адаптивне рішення безпеки для REST, яке використовує методи інфраструктури відкритого ключа для покращення архітектури безпеки. Представлено новий компонент безпеки під назвою «інтелектуальний механізм безпеки», який вивчає можливі випадки виникнення загроз безпеки у SOA з використанням алгоритмів навчання штучних нейронних мереж. Даний компонент прогнозує потенційні атаки на SOA на основі отриманих результатів за допомогою розробленої теоретичної моделі безпеки, а написані алгоритми як частина рішення безпеки запобігають SOA-атакам. Саме такі рішення є досить перспективними в плані забезпечення потрібного рівня безпеки в сучасних програмних продуктах, які орієнтовані на хмарні технології.

У нинішню епоху хмарної парадигми потік послуг, програм та доступу до даних через Інтернет суттєво зростає. Зазвичай користувачам необхідно

пройти автентифікацію кілька разів, щоб отримати повноваження та отримати доступ до потрібних служб або програм. У зв'язку з цим у роботі [9] пропонується повністю безпечна схема для пом'якшення багаторазової автентифікації, яка вимагається від конкретного користувача. У запропонованій моделі федеративна довіра створюється між двома різними доменами: споживачем та постачальником. Весь трафік, що надходить до постачальника послуг, далі ділиться на три етапи залежно від ризиків, пов'язаних із даними відповідного користувача. Єдиний вхід (SSO) та багатофакторна автентифікація (MFA) розгортаються для забезпечення автентифікації, авторизації, обліку та доступності (AAAA) для забезпечення безпеки та конфіденційності облікових даних кінцевого користувача. У запропонованому рішенні використовується висновок про те, що MFA досягає кращого шаблону AAAA проти SSO та доступність (AAAA) для забезпечення безпеки та конфіденційності облікових даних кінцевого користувача. Такий підхід ускладнює процес використання даних облікових записів легітимних користувачів системи, але якщо зловмиснику вдалося проникнути в систему за допомогою цих облікових даних, то далі злому системи не уникнути.

Ще одне досить цікаве дослідження представлено в роботі [10], де представлено дослідження методів ідентифікації шкідливого програмного забезпечення в комп'ютерних системах. Одну з найзначніших загроз безпеці комп'ютерних систем та інформації в цілому складає шкідливе програмне забезпечення, або комп'ютерні віруси. Слід зауважити, що зазначена проблема посилюється динамічним зростанням кількості мобільних пристроїв, загальним переходом на хмарні технології і поширенням Інтернет-технологій, що призводить до зростання кількості шкідливого програмного забезпечення. В роботі [10] розглянуто програмне забезпечення, яке генерує функції переходів магазинного автомату відповідно до заданих правил граматики. Далі проводиться аналіз вхідного файлу на наявність заданих ознак, характерних для шкідливого програмного забезпечення, та

моделюється робота детермінованого низхідного магазинного автомату. За результатом роботи магазинного автомату формується висновок щодо можливості зараження комп'ютерної системи.

Такі системи доцільно використовувати поряд з іншими методами та засобами забезпечення безпеки. Оскільки якщо зловмиснику вдасться проникнути в систему як легітимний користувач, то така підсистема не дасть змоги йому використати шкідливе програмне забезпечення і залишитися непомітним. Але такі підсистеми ніяким чином не вбережуть дані системи, до яких легітимний користувач має доступ.

Автори статті [11] проаналізували персоналізовані звички користувачів до їх іменування з точки зору відображених імен, а потім використовували різні методи розрахунку подібності, щоб визначити подібність функцій, які містяться в відображених іменах. Крім того, автори також вимірювали та аналізували графік інтересів користувачів, щоб ще більше підвищити ефективність ідентифікації користувачів. Автори об'єднали обмеження «один до одного» з алгоритмом Гейла-Шеплі, щоб усунути проблеми співвідношення облікових записів «один до багатьох» та «багато до багатьох», які часто виникають у процесі складання результатів. Експериментальні результати показали, що запропонований метод дозволяє ідентифікувати користувача, використовуючи лише невеликий обсяг онлайн-даних. На жаль, за рахунок використання алгоритму Гейла-Шеплі даний метод має і всі його недоліки, а отже при великій кількості користувачів співвідношення користувача та його профіля буде досить тривалою операцією. Але результати цих досліджень доводять працездатність методів ідентифікації користувачів за їх поведінкою всередині певної системи.

Також в статті [12] дослідники пропонують розраховану на велику кількість користувачів нейромережу під назвою MISS для ідентифікації користувача в системах з загальним обліковим записом. Тобто в роботі представлено рішення проблеми ідентифікації користувача з урахуванням сесій. MISS складається з двох основних компонентів: один з них – нейронна

мережа Dwell Graph (DGNN), яка включає час перебування елемента в нейронній мережі із закритим графом, щоб фіксувати зміну інтересу користувача між сесіями. Інший – модуль, розрахований на ідентифікацію користувачів (MI), який використовується для розрізнення поведінки різних користувачів під одним і тим же обліковим записом. Даний метод використовується для того, щоб видавати користувачам рекомендації щодо перегляду контенту на основі їх вподобань, тобто на основі їх поведінки під час сесії. Дане дослідження доводить можливість та ефективність використання нейронних мереж для побудови функції, яка описує поведінку користувача під час сесії в системі. Це дозволить побудувати додаткові засоби захисту сесії користувача.

У статті [13] пропонується простий, але потужний підхід до побудови профілю поведінки користувачів при веб-перегляді з метою ідентифікації користувача. Автори створюють профілі користувачів, які фіксують силу поведінкових моделей користувачів, які можна використовувати для ідентифікації користувачів. Їх експерименти показують, що ці профілі можуть бути більш точними при ідентифікації користувачів, ніж дерева рішень, коли спостерігається достатня кількість веб-активностей, і можуть досягти більшої ефективності, ніж машини опорних векторів.

На основі актуальних публікацій можна зробити висновок, що розвиток сучасних алгоритмів ідентифікації практично унеможливило злом самих методів ідентифікації користувачів системи. Але дані методи ніяким чином не захищені від випадків, коли користувач добровільно передає свої облікові дані зловмиснику. А аналіз сучасних підходів до ідентифікації користувачів за їх поведінкою говорить про те, що ці підходи є досить ефективними і використовуються при прогнозуванні вподобань користувача. Отже, опираючись на проведений аналіз на сьогоднішній день актуальною є задача розробки методів та засобів, які дозволяють ідентифікувати користувачів за їх поведінкою під час сесії, що дозволить підвищити захищеність системи від злому в цілому.

2 ВИЯВЛЕННЯ ТА ПОПЕРЕДЖЕННЯ ВТОРГНЕНЬ В РОБОТУ ІНФОРМАЦІЙНОЇ КОМП'ЮТЕРНОЇ СИСТЕМИ

2.1 Системи виявлення і попередження вторгнень та атак

У корпоративній мережі зазвичай є кілька точок доступу до інших мереж. Вони можуть бути як приватними, так і публічними. Головне завдання полягає в тому, щоб підтримувати безпеку цих мереж, зберігаючи їх відкритими для своїх користувачів. Сьогодні атаки бувають настільки комплексними, що можуть перешкодити найкращим системам безпеки. Особливо якщо вони працюють, виходячи і припущення, що ресурси організації можна захистити за допомогою брандмауерів або шифрування. Наприклад, шкідливе програмне забезпечення може відправляти пакети, що виглядають повністю «нормальними» для фаєрвола. Тому цих технологій не вистачає для протидії сучасним загрозам. На перший план виходять так звані системи виявлення вторгнень (IDS) та системи запобігання вторгненням (IPS) [14, 15].

IDS/IPS системи – програмні та апаратні інструменти для захисту мереж від несанкціонованого доступу. Вони здатні автоматично виявляти факти вторгнень та запобігати їм, сповіщаючи відповідальних фахівців. Хоча з технологічної точки зору IDS та IPS дуже схожі, завдання та вимоги до них зовсім різні, тому слід розрізняти ці поняття. Аббревіатура IDS означає Intrusion Detection System, а IPS – Intrusion Prevention System. Відповідно, перша технологія здійснює моніторинг загроз, а друга займається їх запобіганням.

Початківцю з інформаційної безпеки може бути важко зрозуміти, навіщо потрібні IPS і IDS, коли є фаєрвол. Особливо, коли вони виконують схожу функцію фільтрації трафіку. Проте між цими інструментами є фундаментальна різниця.

Головна функція міжмережевого екрана – контроль доступу на рівні мережі. Фаєрвол вказує, які комп'ютери можуть звертатися до ділянок мережі, ґрунтуючись на певному наборі роздільної здатності. Тобто пропустити певний трафік, інше заборонити. IPS/IDS у свою чергу працюють за зворотним принципом – заблокувати проблему безпеки (наприклад, будь-який пакет), все інше пропустити (у разі відсутності приводів для підозр на вторгнення).

Крім концептуальної існує й технічна відмінність. Фаєрволи добре працюють на 2-4 рівнях моделі OSI. Для прийнятної роботи на вищих рівнях у них мало вбудованого функціоналу. Тому міжмережеві екрани переважно контролюють лише параметри сесії: стан зв'язків, номери портів, IP. Системи IPS і IDS дозволяють працювати на більш високих рівнях, аналізуючи як заголовки та їх невеликі шматочки, а й вміст пакетів. А якщо є можливість розпакувати пакет, то можна перевірити дані, що передаються, на предмет негативної поведінки.

2.2 Системи виявлення вторгнень IDS

Насправді технології IDS не є чимось принципово новим. Кошти виявлення вторгнень з'явилися близько 30 років тому. Перші IDS були розроблені для операційної системи SINEX (UNIX для систем виробництва Siemens). Вони контролювали доступ користувачів з терміналів до основних ресурсів мейнфреймів.

Розглянемо більш детально, що ж таке IDS. IDS – це система виявлення атак, призначена для сканування мережного трафіку, реєстрації підозрілої активності в мережі та оповіщення при спрацюванні певних правил. Зазвичай IDS переглядає трафік та журнали, шукає в даних ознаки шкідливої активності та у разі атаки повідомляє спеціаліста з безпеки через керуючу консоль, SMS-повідомлення або електронну пошту.

IDS вміють реєструвати різноманітні види атак (DDoS; атаки через Bot

C&C і P2P, використання SQL ін'єкцій, атаки на сервіси IMAP, POP3, VoIP, SMTP і т.п.), виявляти спроби підвищення привілеїв та несанкціонованого доступу, виявляти активність шкідливого програмного забезпечення (черв'яки, трояни, малварі та експлойти), відстежувати сканування портів та їх відкриття.

Важливо пам'ятати, що IDS – це не засіб безпосереднього контролю, а інструмент поліпшення видимості мережі. IDS допомагає фахівцям з безпеки зрозуміти, наскільки все добре із захищеністю. У цьому плані ця технологія схожа на аналізатор протоколів (наприклад, Wireshark), тільки у даному контексті йдеться про аналіз та оцінку безпеки.

2.2.1 Класифікація IDS

Системи виявлення атак бувають найрізноманітнішими: апаратними та програмними, опенсорсними та пропріетарними. Розглянемо дві класифікації, важливі під час виборів тієї чи іншої рішення.

На вигляд аналізованого трафіку IDS поділяють на:

- засновані на протоколі (PIDS);
- засновані на прикладних протоколах (APIDS).

Перший різновид моніторить комунікаційні протоколи зі зв'язаними користувачами чи системами. Другий вид аналізує вузький список прикладних протоколів, специфічних додатків. Приклади досить популярних APIDS: PHPIDS, GreenSQL-FW та Mod_Security.

Від розташування в мережі IDS поділяють на:

- хостові (Host-based Intrusion Detection Systems, HIDS);
- мережеві (Network Intrusion Detection Systems, NIDS).

Тут все більш-менш очевидно. HIDS проводить моніторинг у межах єдиного хоста, а NIDS – у межах мережного сегмента, де вона встановлена. Мережеві IDS більш універсальні, ніж прикладні або хостові. Багато в чому це досягається завдяки технології глибокого інспектування пакетів (Deep

Packet Inspection, DPI), що дозволяє аналізувати весь трафік від каналного рівня і вище. Проте за універсальність доводиться платити додатковим навантаженням на обчислювальні ресурси.

Існують інші різновиди IDS, наприклад VMIDS. Цей тип базується на використанні технологій віртуалізації, що дозволяє обійтися без розгортання системи на окремому пристрої.

2.2.2 Способи виявлення вторгнень

У технічній літературі та статтях, присвячених методам виявлення вторгнень, можна знайти два основних методи: виявлення зловживань (сигнатурні IDS) та виявлення аномалій (IDS, засновані на аномаліях).

В основі виявлення зловживань лежить сигнатурний аналіз трафіку та/або вузлових подій (журнали додатків, сисколи) плюс аналізатори протоколів. Якщо говорити по-простому, цей спосіб базується на існуванні опису відомих атак. Система шукає в реальному трафіку та поведінці додатків патерни, які є в базі. Переважна кількість сучасних систем використовують цей підхід. Головний плюс такого методу – фахівцю зрозуміло, чому IDS зреагувала. Якщо спрацювала сигнатура на трафік, ми можемо подивитись її текст, звернутися до логів системи, бази сигнатур і розібратися, що саме сталося. Проте бази потрібно завжди підтримувати у актуальному стані.

Метод виявлення аномалій працює навпаки. Нам відомо, що таке нормальний трафік та нормальна поведінка додатків. Система намагається розпізнати відхилення від цієї поведінки. Дивовижно, але IDS на основі аномалій були першими та з'явилися близько 30 років тому. Нині такі системи навчають методами машинного навчання на прикладах нормального функціонування.

Головний недолік даних IDS – вони мають навчатися на нормальних даних. Це означає, що система працює деякий час у конкретній мережі на

певному вузлі, на якому виявлятимуться атаки. Коли трапляється щось погане (система реагує), аналітику дуже складно зрозуміти, чому пакет чи сесія вважалися ненормальними. Добре, якщо ми маємо лише кілька параметрів (наприклад обсяг переданих даних та кількість відкритих з'єднань за секунду), за якими розпізнається нормальна поведінка. Завдання ускладнюється, якщо таких параметрів кілька десятків плюс використовується якийсь хитромудрий алгоритм машинного навчання. Однак є клас завдань, у якому IDS на основі аномалій немає рівних – системи виявлення DDoS атак. Такі системи здатні швидко визначати джерело атаки DDoS, ефективно блокувати атакуючий трафік і навіть скидати його на стороні провайдера.

2.3 Системи попередження вторгнень IPS

IDS здатні лише повідомити відповідальну особу про небажану активність. Фахівцю доведеться самостійно переналаштувати фаєрвол під час перегляду звітів. Але часто потрібно зреагувати в реальному часі, запобігши вторгненню на ранній стадії. Для цього використовують вже згадувані IPS (системи запобігання атакам). Вони здатні автоматично припиняти шкідливі дії, наприклад, перервати сеанс або переналаштувати пакетний фільтр.

2.3.1 Класифікація IDS

IPS – це один з різновидів IDS, оскільки використовує ті ж методи виявлення атак. Виходить своєрідний гібрид IDS та фаєрвола. Найчастіше IDS та IPS є одним і тим же пристроєм, який можна по-різному налаштувати та підключити до мережі.

IDS і IPS класифікують так. Існують хостові IPS (HIPS) та мережеві IPS (NIPS). NIPS запобігає вторгненню шляхом вбудовування «в розрив» мережі та пропускання через себе трафіку. Як правило у цього виду є зовнішній

інтерфейс, що приймає трафік, і внутрішній, що пропускає легітимний трафік.

Також IPS поділяють на ті, що моніторять трафік і порівнюють його з відомими сигнатурами, і ті, які вишукують підозрілий трафік на основі аналізаторів протоколів та бази знайдених уразливостей. Другий спосіб допомагає захищатися від ще невідомих класів атак. Якщо говорити про способи реакції на вторгнення, то основними є: переналаштування комунікаційного обладнання, блокування конкретних користувачів та хостів, обрив сеансів за допомогою TCP RST або засобами фаєрволу.

Поведінка IDS та IPS під час атаки показана на рисунку 2.1.

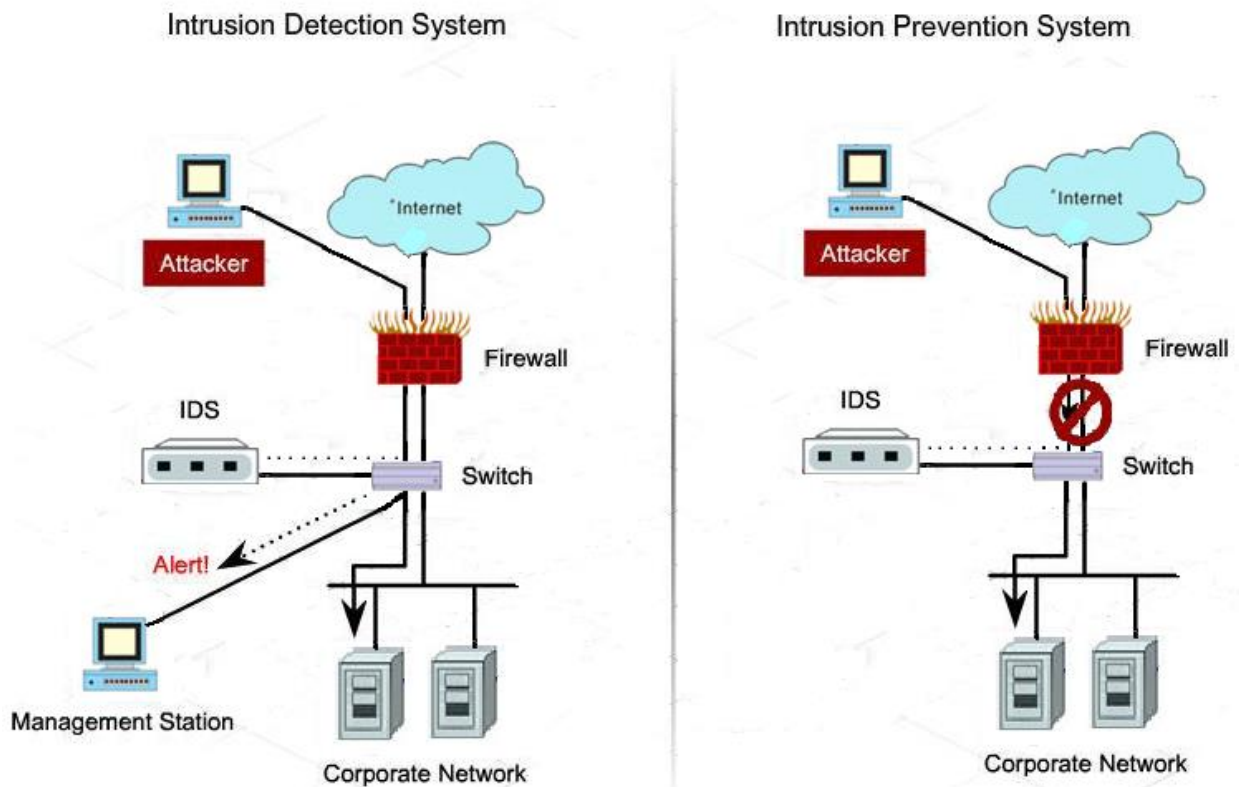


Рисунок 2.1 – Різниця між IDS та IPS

3 МЕТОДИ МАШИННОГО НАВЧАННЯ ТА ЇХ ВИКОРИСТАННЯ ДЛЯ ІДЕНТИФІКАЦІЇ КОРИСТУВАЧІВ У СИСТЕМІ ЗА ЇХ ПОВЕДІНКОЮ

3.1 Загальні відомості про машинне навчання

За останні кілька років машинне навчання для широкого загалу стало синонімом штучного інтелекту. І хоча машинне навчання як наукова дисципліна існує вже кілька десятиріч, у світі знайдеться лише жменька організацій, які повною мірою усвідомили його потенціал.

Незважаючи на доступність сучасних бібліотек, пакетів та каркасів машинного навчання з відкритим вихідним кодом, які підтримуються провідними організаціями та широким співтовариством вчених та програмістів, більшість компаній все ще зазнають труднощів із застосуванням машинного навчання до вирішення практичних ділових задач.

Машинне навчання (machine learning) – клас методів штучного інтелекту, характерною рисою яких є не пряме розв'язання задачі, а навчання за рахунок застосування рішень безлічі подібних задач. Для побудови таких методів використовуються засоби математичної статистики, чисельних методів, математичного аналізу, методів оптимізації, теорії ймовірностей, теорії графів, різних технік роботи з даними в цифровій формі.

Якщо казати простіше, машинне навчання – це розділ інформатики, присвячений побудові алгоритмів, які працюють із набором прикладів, що описують явисьце. Приклади можуть надходити з природи, створюватися людьми чи генеруватися іншим алгоритмом [16].

Машинне навчання також можна визначити як процес розв'язання практичної задачі шляхом збору набору даних та алгоритмічного навчання статистичної моделі на цьому наборі. Передбачається, що ця статистична модель якимось чином використовується для розв'язання практичної задачі.

Назва «машинне навчання» була введена в ужиток Артуром Семюелем

у 1959 році. Пройшовши певний шлях еволюції від задач розпізнавання образів та теорії обчислювального навчання в царині штучного інтелекту, машинне навчання займається вивченням та побудовою алгоритмів, які можуть навчатися й робити передбачення з даних, – такі алгоритми відходять від строгого слідування статичним програмним інструкціям, здійснюючи керовані даними прогнози або прийняття рішень шляхом побудови моделі з вибіркового входу. Машинне навчання застосовують для розв'язання обчислювальних задач, в яких розробка та програмування явних алгоритмів з високою продуктивністю є складним або нездійсненним завданням; до прикладів застосувань належать фільтрування електронної пошти, виявлення вторгнень у роботу мережі або зловмисників, які намагаються створити витік даних, оптичне розпізнавання символів, навчання ранжуванню та комп'ютерний зір.

Як окрема область наукового пізнання машинне навчання почало бурхливо розвиватися в 1990-х роках. Ця область змінила свої цілі з досягнень штучного інтелекту на розв'язання практичних задач. Вона змістила фокус із символічних підходів, успадкованих нею від штучного інтелекту, в бік методів та моделей, запозичених зі статистики та теорії ймовірності. Вона також виграла від збільшеної доступності оцифрованої інформації та можливості розповсюдження її через Інтернет.

3.2 Способи машинного навчання

Якщо потрібно класифікувати методи машинного навчання у залежності від способу навчання, то тут можна виділити дві великі категорії, які залежать від наявності зворотного зв'язку або навчального сигналу у процесі навчання: навчання з учителем та навчання без учителя. В окремих випадках вхідний сигнал може бути доступним лише частково, або бути обмеженим особливим зворотним зв'язком, тоді використовується навчання з частковим залученням учителя або навчання з підкріпленням [16].

3.2.1 Навчання з учителем

Навчання з учителем, яке ще називають контрольованим або керованим навчанням (supervised learning) – один зі способів машинного навчання, в ході якого модель примусово навчається за допомогою наявної множини прикладів «стимул-реакція» з метою визначення «реакції» для «стимулів», які не належать до наявної множини прикладів. З точки зору кібернетики, навчання з учителем є одним із видів кібернетичного експерименту.

Між входами та еталонними виходами (стимул-реакція) може існувати деяка залежність, але вона апріорі не відома. Відома лише кінцева сукупність прецедентів – пар «стимул-реакція», яку називають навчальною вибіркою. На основі цих даних потрібно відновити залежність (побудувати модель відношення стимул-реакція, придатну для прогнозування), тобто побудувати алгоритм, здатний для будь-якого об'єкта видати досить точну відповідь. Для вимірювання точності відповідей, так само як і в навчанні на прикладах, може вводиться функціонал якості.

Скажімо, якщо прикладами є повідомлення електронної пошти, а наше завдання полягає у виявленні спаму, можна виділити два класи: спам та не спам. У випадку навчання з учителем задача передбачення класу називається класифікацією, а задача передбачення дійсного числа називається регресією. Значення, яке має бути передбачене моделлю, навченою з учителем, називається цільовим показником, чи метою. Прикладом регресії є задача передбачення заробітної плати співробітника з урахуванням його досвіду роботи та знань. Прикладом класифікації є ситуація, коли лікар вводить характеристики пацієнта у додаток, а він повертає діагноз.

Відмінність між класифікацією та регресією показано на рисунку 3.1. У разі класифікації алгоритм навчання шукає лінію (або, у загальному випадку, гіперповерхню), яка розділяє приклади різних класів. З іншого боку, у разі регресії алгоритм навчання прагне знайти лінію чи гіперповерхню, яка добре відповідає навчальним прикладам.

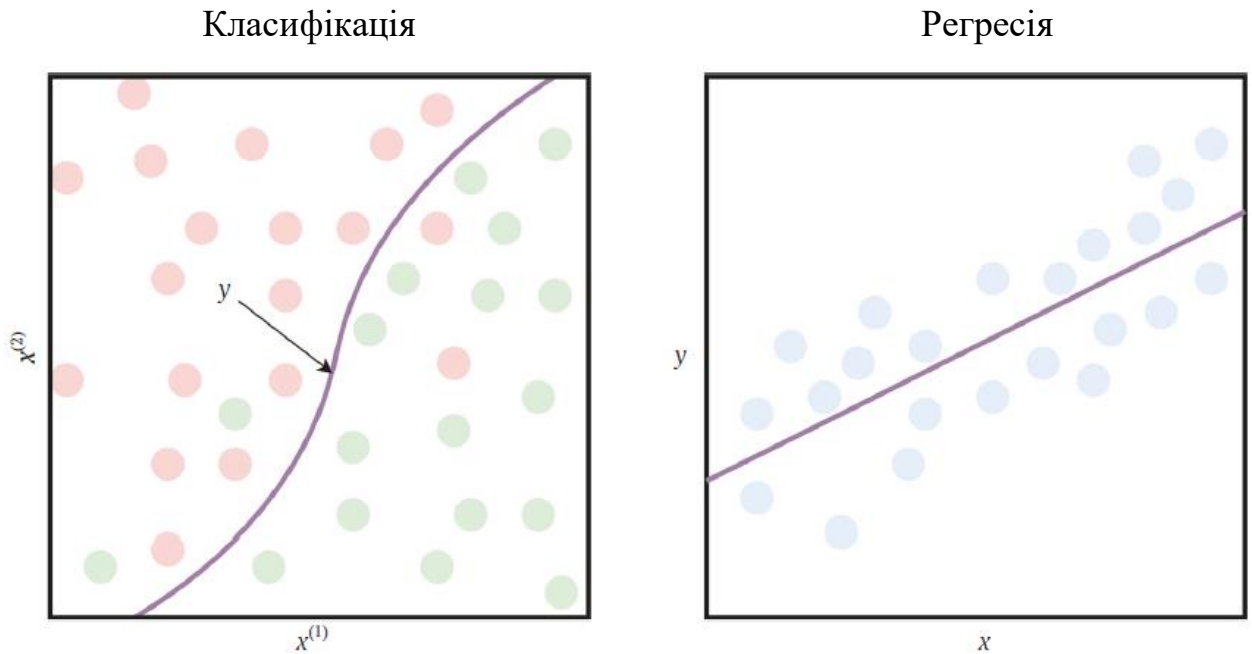


Рисунок 3.1 – Різниця між класифікацією та регресією

Навчання з частковим залученням учителя, яке відоме ще як напівавтоматичне навчання або часткове навчання (semi-supervised learning) – спосіб машинного навчання, різновидність навчання з учителем, яке також використовує немарковані дані для тренування – зазвичай невелику кількість маркованих даних та велику кількість немаркованих даних. Напівавтоматичне навчання займає проміжну позицію між навчанням без учителя (без залучення будь-яких маркованих даних для тренування) та навчанням з учителем (із залученням лише маркованих даних). Багато дослідників машинного навчання виявили, що немарковані дані, при використанні в поєднанні з невеликою кількістю маркованих даних, можуть значно поліпшити точність навчання. Задання маркованих даних для задачі навчання часто вимагає кваліфікованої людини (наприклад, для транскрибування аудіофайлу) або фізичного експерименту (наприклад, для визначення 3D структури білка або виявлення наявності нафти в певному регіоні). Тому затрати на маркування даних можуть зробити процес навчання з використанням лише маркованих даних нездійсненним, в той час як процес задання немаркованих даних не є дуже затратним. У таких ситуаціях,

напіваавтоматичне навчання може мати велике практичне значення. Таке навчання також представляє інтерес у сфері машинного навчання та як модель для людського навчання.

Навчання з підкріпленням (reinforcement learning) – один із способів машинного навчання, в ході якого випробувана система (агент) навчається, взаємодіючи з деяким середовищем (environment). З погляду кібернетики, навчання з підкріпленням є одним із видів кібернетичного експерименту. Відгуком середовища (а не спеціальної системи управління підкріпленням, як це відбувається у навчанні з учителем) на прийняті рішення є сигнали підкріплення, тому таке навчання є окремим випадком навчання з учителем, але вчителем є середовище або його модель. Також потрібно мати на увазі, що деякі правила підкріплення базуються на неявних учителях, наприклад, у разі штучного нейронного середовища, одночасної активності формальних нейронів, через що їх можна віднести до навчання без вчителя.

Агент впливає на середовище, а середовище впливає на агента. Про таку систему говорять, що вона має зворотний зв'язок. Таку систему слід розглядати як єдине ціле, і тому лінія поділу між середовищем та агентом є досить умовною. Звичайно, з анатомічної або фізичної точок зору між середовищем та агентом (організмом) існує цілком певна межа, але якщо цю систему розглядати з функціональної точки зору, то поділ стає нечітким.

3.2.2 Навчання без учителя

Навчання без учителя, відоме ще як самонавчання або спонтанне навчання (unsupervised learning) – один із способів машинного навчання, при якому модель спонтанно навчається виконувати поставлену задачу, без втручання з боку експериментатора. З точки зору кібернетики, навчання без учителя є одним з видів кібернетичного експерименту. Як правило, цей спосіб підходить тільки для задач, у яких відомий опис множини об'єктів (навчальна вибірка), і необхідно виявити внутрішні взаємозв'язки,

залежності, закономірності, що існують між об'єктами.

Навчання без учителя часто протиставляється навчанню з учителем, коли для кожного об'єкта, що навчається, примусово задається «правильна відповідь», і потрібно знайти залежність між стимулами та реакціями системи.

Мета алгоритму навчання без учителя полягає у породженні моделі, яка на вході приймає вектор ознак і перетворює його або в інший вектор, або у значення, що використовується для вирішення практичного завдання. Наприклад, у разі кластеризації для кожного вектора ознак набору даних модель повертає ідентифікатор кластера. Кластеризація використовується для пошуку груп схожих об'єктів у великому наборі об'єктів, наприклад зображень або текстових документів. Використовуючи кластеризацію, аналітик може, наприклад, вибрати досить репрезентативну, але малу підмножину непомічених прикладів з великого набору, щоб потім помітити їх вручну: з кожного кластера вибирається лише кілька прикладів, замість відбирати безпосередньо з великого набору з ризиком вибрати дуже схожі приклади.

У задачі зниження розмірності виходом моделі є вектор ознак з меншим числом вимірів, ніж на вході. Наприклад, дослідник має вектор ознак, надто складний для візуалізації (оскільки число вимірів більше трьох). Модель зниження розмірності може перетворити цей вектор на інший (зберігаючи частину інформації) – двовимірний чи тривимірний. Цей новий вектор ознак можна зобразити на графіку.

У задачі виявлення викидів виходом є дійсне число, що показує, наскільки вхідний вектор ознак відрізняється від типового прикладу в наборі даних. Виявлення викидів застосовується для вирішення задачі проникнення в мережу (шляхом виявлення аномальних мережевих пакетів, що відрізняються від типового пакета в «нормальному» трафіку) або виявлення новизни (наприклад, документа, що відрізняється від інших документів у наборі).

3.3 Життєвий цикл проекту машинного навчання

Робота над проектом машинного навчання відрізняється від роботи над типовим проектом у галузі програмної інженерії. На відміну від традиційної програмної інженерії, де поведінка програми зазвичай детермінована, додатки машинного навчання включають моделі, поведінка яких з часом може з природних причин погіршуватися або ставати аномальною. Така аномальна поведінка моделі може мати різні причини, у тому числі фундаментальну зміну вхідних даних або впровадження нового екстрактора ознак, що повертає інший розподіл чи значення іншого типу. Нерідко можна почути, що системи машинного навчання відмовляють мовчки. Інженер з машинного навчання повинен бути здатний запобігати таким відмовам або, якщо повне запобігання неможливе, знати, як їх виявляти та усувати.

Проект машинного навчання починається з осмислення цільового критерію з погляду бізнесу. Зазвичай бізнес-аналітик працює із замовником та аналітиком даних, щоб трансформувати бізнес-задачу у технічний проект. Технічний проект може містити або не містити частину, пов'язану з машинним навчанням.

Після визначення технічного проекту якраз і починається область інженерії машинного навчання. Машинне навчання в рамках ширшого технічного проекту насамперед має чітко визначену мету. Метою машинного навчання є опис того, що статистична модель отримує на вході, що вона генерує на виході, та критеріїв прийнятної (або неприйнятної) поведінки моделі.

Загалом життєвий цикл проекту машинного навчання, показаний на рисунку 3.2, складається з наступних етапів:

- визначення мети;
- збір та підготовка даних;
- конструювання ознак;
- навчання моделі;

- оцінювання моделі;
- розгортання моделі;
- виконання моделі;
- моніторинг моделі;
- супровід моделі.

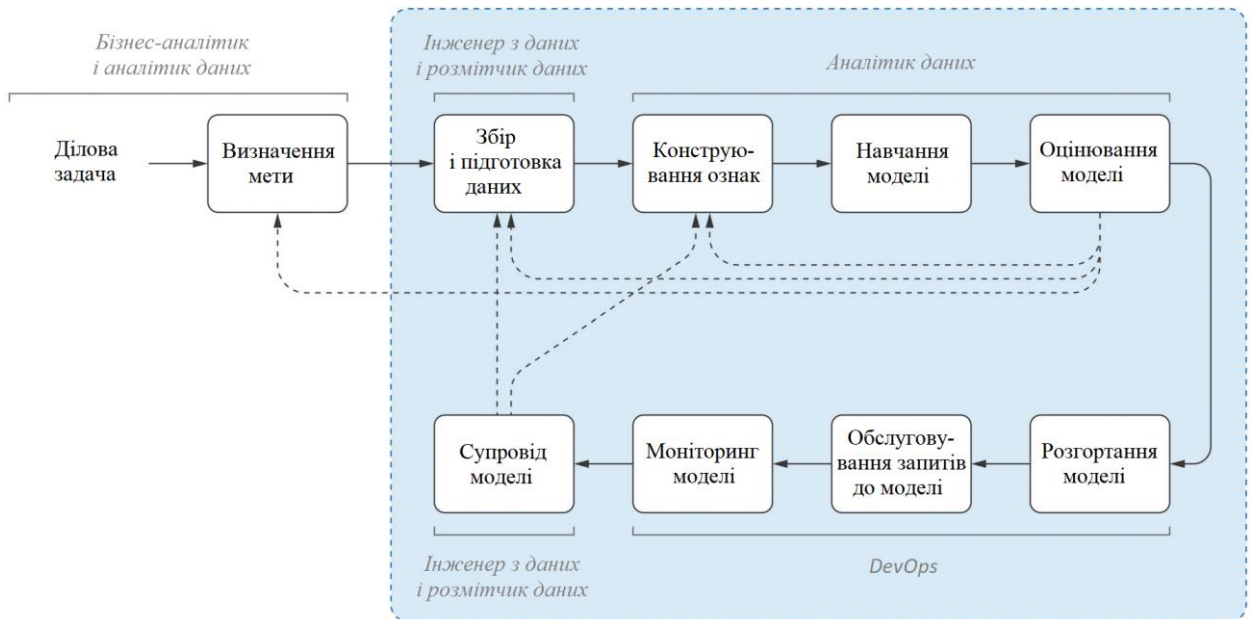


Рисунок 3.2 – Життєвий цикл проекту машинного навчання

На рисунку 3.2 сфера застосування машинного навчання обмежена синьою зоною. Суцільні стрілки показують типовий технологічний потік. Пунктирні стрілки означають, що на деяких етапах може бути прийнято рішення повернутися назад або зібрати більше даних, або зібрати інші дані і переглянути ознаки (виключивши одні і сконструювавши інші).

4 РОЗРОБКА МОДЕЛІ ІДЕНТИФІКАЦІЇ КОРИСТУВАЧА СИСТЕМИ

Метою даної роботи є розробка підходу ідентифікації користувача системи за його поведінкою під час сесії для виявлення вторгнень в систему зловмисників за допомогою легітимних даних користувачів. Це дасть можливість побудови додаткових підсистем захисту, які будуть непомітні для зловмисника і які важко буде обійти. Це в свою чергу підвищить загальний захист системи від необачних дій користувачів, які випадково або навмисно передадуть свої облікові записи зловмиснику.

Для досягнення мети були поставлені наступні завдання:

- розробити функціональну модель процесу забезпечення ідентифікації користувачів за їх поведінкою в системі;
- провести експериментальне дослідження, щодо запропонованого підходу.

Задача ідентифікації полягає у встановленні математичних співвідношень між вимірюваними входами і виходами при заданих вимірах у часі [17].

Ідентифікація здійснюється за допомогою моделі, яку можливо налаштовувати, тієї чи іншої структури, параметри якої можуть бути змінені. Функціональну схему ідентифікації можна представити у наступному вигляді, зображеному на рисунку 4.1.

В кожен момент часу $t=1,2,\dots,n$ до входів об'єкта і моделі, що налаштовується, прикладено зовнішній сигнал $u(t)$. Об'єкт збурюється також деякою випадковою величиною $\xi(t)$. Вихідна величина об'єкта $y(t)$ залежить як від зовнішнього впливу та завад (шуму), так і від невідомого вектору параметрів w' . Вихідна величина $y'(t)$ моделі, що налаштовується, залежить від вектору параметрів, які налаштовуються. Вони перераховується відповідно до алгоритму, що обробляє вектор всіх спостережень $z(t)$. Набір цих спостережень залежить від певних задач ідентифікації.

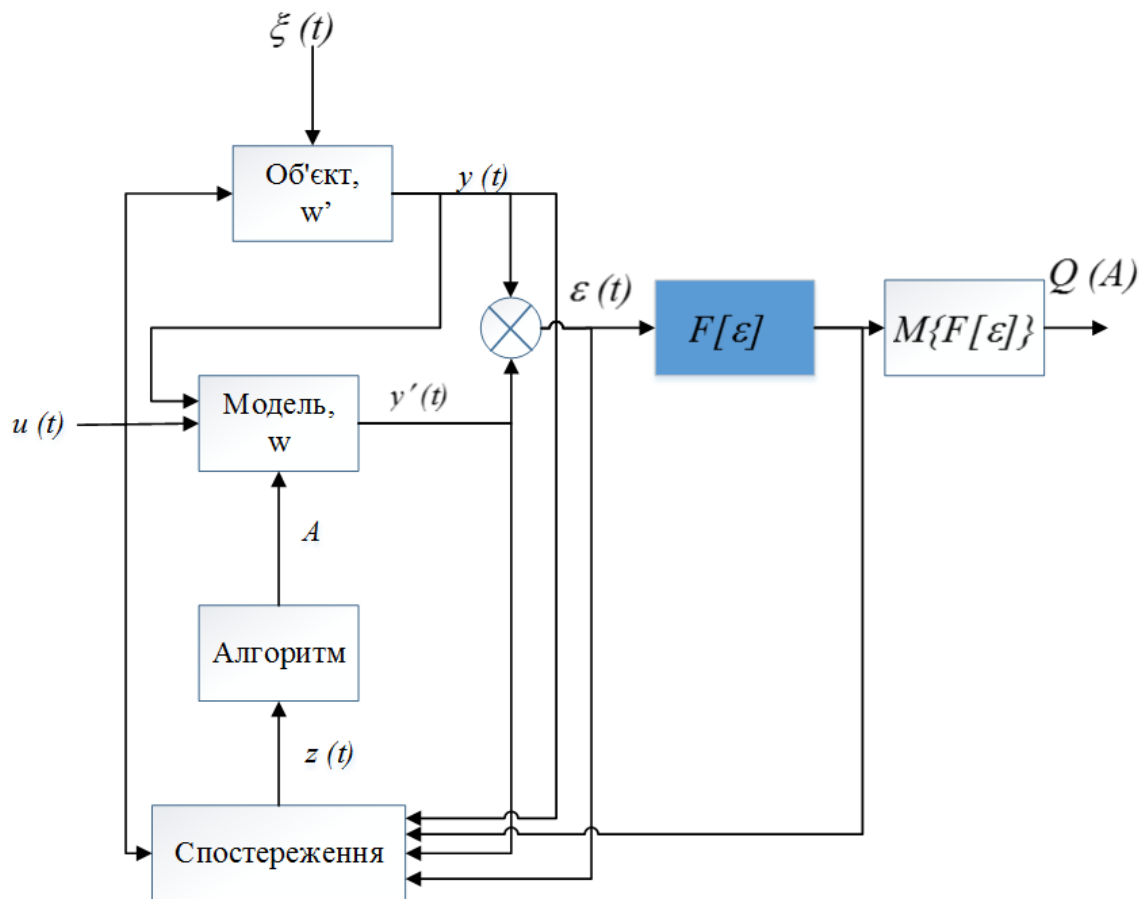


Рисунок 4.1 – Функціональна схема ідентифікації

Різниця вихідних величин об'єкта і моделі, що налаштовується, утворює деяку похибку

$$\varepsilon(z(t), A) = y(t) - y'(t), \quad (4.1)$$

яка подається на вхід функціонального перетворювача, зображеного на рисунку 4.1 синім прямокутником. Далі завжди мається на увазі, що об'єкт працює в стаціонарному режимі, тобто ймовірні характеристики послідовностей $y(t)$, $y'(t)$, а отже і $z(t)$ не залежать від моменту часу t . Такий режим називається режимом нормальної роботи.

Відповідність моделі, що налаштовується, та об'єкта, тобто якість ідентифікації, оцінюється критерієм якості ідентифікації

$$Q(A) = M\{F[\varepsilon(z(t), w)]\}, \quad (4.2)$$

де F – функція втрат, а M – символ математичного сподівання.

Критерієм якості ідентифікації (4.2) є середні втрати (похибка). Чим менше середні втрати, тим краща якість ідентифікації. Поліпшення якості ідентифікації здійснюється належним вибором структури моделі, що налаштовується, і набором її параметрів. Налаштування цих атрибутів здійснюється алгоритмом ідентифікації в процесі навчання моделі.

Алгоритм ідентифікації визначається функцією втрат і структурою моделі, що налаштовується. За спостереженнями вхідного впливу вихідних величин об'єкта і моделі алгоритм ідентифікації змінює параметри останньої так, щоб середні втрати досягали мінімуму. Ці умови відповідають ідентифікації в режимі нормальної роботи об'єкта.

Для розв'язання задачі ідентифікації, як це впливає із функціональної схеми (рисунок 4.1), необхідно:

- визначити класи об'єктів;
- вибрати для цього класу об'єктів модель для налаштування, тобто модель, параметри якої можна змінювати;
- вибрати критерій якості ідентифікації, який характеризував би різницю між вихідними величинами об'єкта та результатами моделі;
- сформулювати алгоритм ідентифікації, який, використовуючи доступні для спостереження значення вхідних та вихідних величин, змінював би параметри моделі, що налаштовується так, щоб середні втрати зі зростанням t досягали мінімуму.

В загальному вигляді задачу ідентифікації можна звести до простої класифікації об'єктів, тому для побудови моделі ідентифікації можуть використовуватися методи машинного навчання, такі як:

- лінійна регресія (Linear Regression) [18];
- логістична регресія (Logistic Regression) [19];

- алгоритм k-найближчих сусідів (K-Nearest Neighbors) [20];
- дерево рішень і випадковий ліс (Decision Trees and Random Forests) [21];
- метод опорних векторів (Support Vector Machines) [22];
- штучна нейронна мережа (Artificial Neural Networks) [23].

Оскільки вибір методу машинного навчання цілком залежить від складності самої моделі класифікації, то в роботі пропонується для демонстрації представленого підходу ідентифікації використовувати логістичну регресію. В експериментальній частині роботи для демонстрації працездатності запропоновано відносно нескладну модель поведінки користувача, яка з легкістю може бути описана логістичною регресією. А як показали експерименти, інші методи навчання мають дуже схожі результати.

Логістична регресія – спосіб побудови лінійного класифікатора, що дозволяє оцінювати апостеріорні можливості приналежності об'єктів класам і є окремим випадком узагальненої лінійної регресії. Передбачається, що залежна змінна набуває двох значень і має біномний розподіл [19].

Оскільки головна сфера використання запропонованого підходу – це системи виявлення вторгнень, то потрібно реалізовувати один із найпростіших бінарних класифікаторів, виявлення легітимного користувача або порушника, використавши логістичну регресію та її навчання за допомогою звичайного (повного) та стохастичного градієнтних спусків.

В логістичній регресії будується лінійний алгоритм класифікації $a: X \rightarrow Y$ виду:

$$a(x, w) = \text{sign} \left(\sum_{j=1}^n w_j f_j(x) - w_0 \right) = \text{sign} \langle x, w \rangle, \quad (4.3)$$

де w_j – вага j -ї ознаки;

w_0 – поріг прийняття рішення;

$w = (w_0, w_1, \dots, w_n)$ – вектор ваг;

$\langle w, x \rangle$ – скалярний добуток ознак об'єкта на вектор ваг.

Передбачається, що штучно запроваджено «константну» нульову ознаку: $f_0(x) = -1$.

Завдання навчання логістичної регресії з L_2 -регуляризацією можна представити наступним чином:

$$Q(w, X) = \frac{1}{m} \sum_{i=1}^m \log(1 + \exp(-y_i \langle x_i, w \rangle)) + \frac{\lambda_2}{2} \|w\|^2 \rightarrow \min_w \quad (4.4)$$

Вважаємо, що $y_i \in \{-1, +1\}$, а нульовою ознакою зроблений одиничний (тобто w_0 відповідає вільному члену). Шукати будемо за допомогою градієнтного спуску:

$$w^{(k+1)} = w^{(k)} - \alpha \nabla_w Q(w, X). \quad (4.5)$$

У разі повного градієнтного спуску $\nabla_w Q(w, X)$ рахується як ϵ , тобто, використовуючи всі об'єкти вибірки. У разі стохастичного градієнтного спуску $\nabla_w Q(w, X) \approx \nabla_w q_{i_k}(w)$, де i_k – випадково вибраний номер доданку з функціоналу (регуляризатор можна внести у суму, попередньо помноживши та розділивши на m). Довжину кроку $\alpha > 0$ у межах даної задачі пропонується брати рівною деякій відносно малій константі.

Градієнт за об'єктом x_i рахується за наступною формулою:

$$\nabla_w Q(w, x_i) = -\frac{y_i x_i}{1 + \exp(-y_i \langle w, x_i \rangle)} + \lambda_2 w. \quad (4.6)$$

В якості критеріїв зупинки необхідно використовувати (одночасно):

- перевірку на евклідову норму різниці вагів на двох сусідніх ітераціях (наприклад, менше деякого малого числа порядку 10^{-6});
- досягнення максимальної кількості ітерацій (наприклад, 1000).

Ініціалізувати ваги можна випадково або нульовим вектором.

Ймовірність приналежності об'єкта x класу $+1$ обчислюється так:

$$P(y = +1 | x) = \frac{1}{1 + \exp(-\langle w, x \rangle)}. \quad (4.7)$$

Матрицю об'єкти-ознаки X необхідно попередньо нормувати.

У логістичній регресії також можна використовувати регуляризацію L_1 . Тоді в функцію втрат додається доданок $\lambda_1 \|w\|_1$. У формулі для обчислення градієнта втрат по вектору коефіцієнтів цей доданок буде відповідати $\lambda_1 \operatorname{sgn}(w)$, де sgn – обчислення знаку числа, що застосовується до вектору коефіцієнтів поелементно.

5 ДОСЛІДЖЕННЯ ЗАПРОПОНОВАНОГО МЕТОДУ ВИЯВЛЕННЯ ВТОРГНЕНЬ У ВЕБ-СЕАНСИ КОРИСТУВАЧА

5.1 Функціональна модель процесу побудови профілю поведінки користувачів та забезпечення на його основі ідентифікації

Ідентифікація користувача за допомогою патернів поведінки є відносно новою та цікавою задачею з точки зору забезпечення додаткових методів захисту інформаційних та комп'ютерних систем побудованих на базі хмарних технологій областю досліджень. У цій роботі продовжено дослідження, представлене в роботах [24-26], та пропонується простий, але дієвий (як показали експерименти) метод ідентифікації користувача за профілем його поведінки в системі.

Для цього опишемо інформаційну комп'ютерну систему наступним кортежем:

$$Sys = \{U, O\}, \quad (5.1)$$

де U – множина користувачів системи;

O – множина спостерігаємих операцій над певними програмами інформаційної комп'ютерної системи для відповідних користувачів множини U і описується наступним кортежем:

$$O_i = \{A_i, Op_i, F_i\}, \quad (5.2)$$

де A_i – це програма, яку використав користувач системи;

Op_i – це операція, яка була використана в A_i програмі;

$F_i = [f_{i,1}, \dots, f_{i,j}]$ $j \geq 1$ – це набір ознак спостерігаємої операції.

Далі для конкретного користувача U_k будемо визначати поведінку користувача за період часу $T = [t_0, t_1]$, як набір сеансів в системі $\{S_{k,1}(T), \dots, S_{k,q}(T)\}$, де $S_{k,q}(T)$ – це підмножина спостерігаємих операцій над певними програмами O_i , для k -го користувача системи за період часу T .

Таким чином відповідно формулі (4.2) для користувача U_k з поведінкою $S_{k,q}(T')$ потрібно знайти оптимальний і надійний спосіб ідентифікації. Цей спосіб визначає чи достатньо схожий профіль U_k в період часу T' на побудований профіль U_k в режимі нормальної роботи, використовуючи методи, представлені в розділі 4 на основі сеансів $S_{k,q}(T)$.

Також в процесі ідентифікації користувача в системі за його поведінкою можна зіштовхнутися з такою проблемою, як велика кількість помилок другого роду. Це пов'язане з тим, що в процесі довгої роботи в системі користувач поступово навчається та переймає досвід інших користувачів. Тим самим його профіль поведінки поступово починає відрізнятися від його ж моделі поведінки побудованій на етапі нормальної роботи системи. Для вирішення цієї проблеми пропонуємо оцінювати ймовірність P , що сеанс $S_{k,q}(T)$ належить користувачу U_k . І якщо $1 - P \geq \alpha$, де α – це допустимий рівень похибки, то на основі валідних даних сеансів $S_{k,q}(T)$ для k -го користувача системи перебудова моделі поведінки користувача.

На рисунку 5.1 в загальному вигляді представлена схема процесу побудови моделі поведінки користувача.

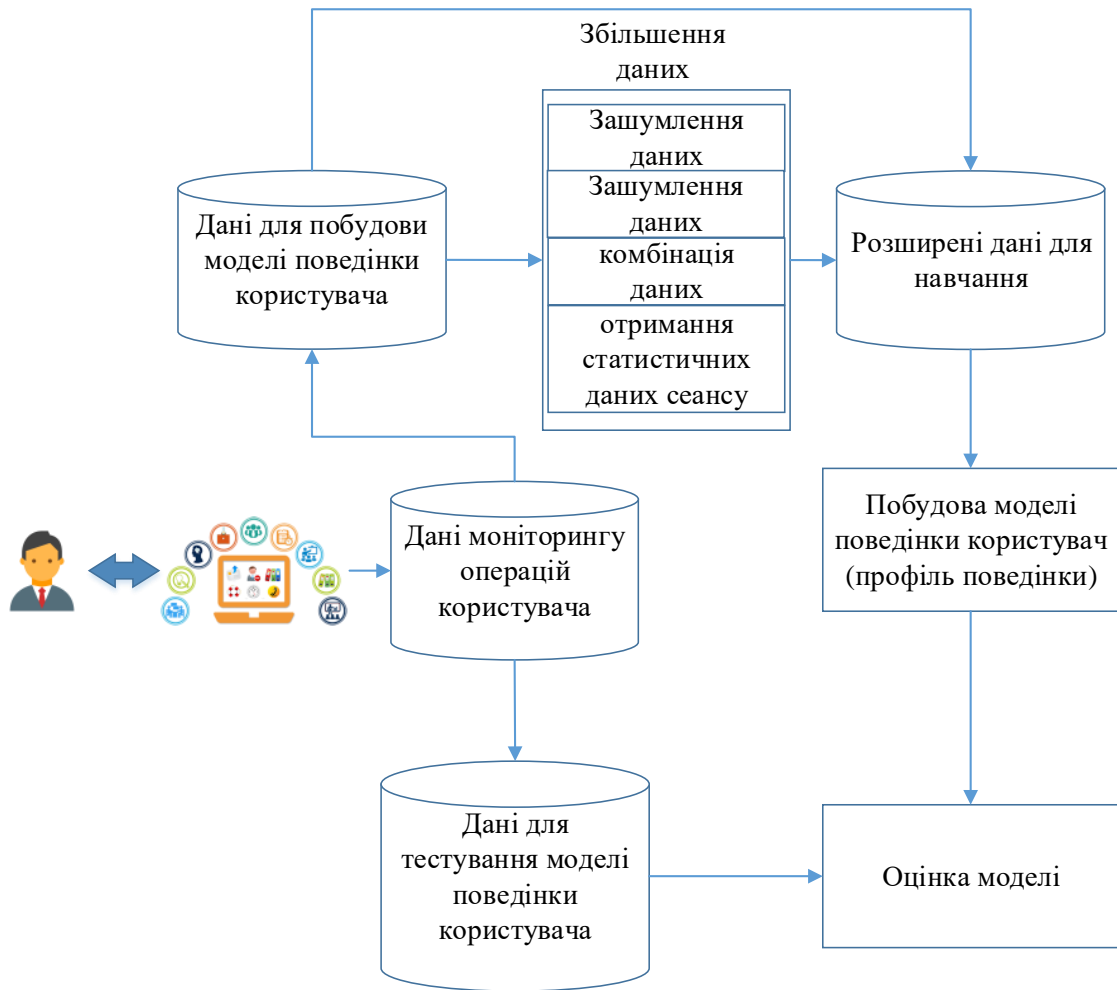


Рисунок 5.1 – Схема процесу побудови моделі поведінки користувача

На рисунку 5.2 представлено схему процесу додаткового захисту системи від вторгнення за допомогою ідентифікації поведінки користувача.

В процесі роботи в інформаційній системі підсистема моніторингу збирає данні про активність користувача в системі O_i , тим самим формуючи набір сеансів $S_{k,q}(T)$, які будуть використовуватися для ідентифікації і побудови моделі поведінки. Далі на основі цих даних за схемою, представленою на рисунку 5.1, здійснюється побудова моделі поведінки користувача. Цей процес відбувається при додаванні нового користувача в систему та в режимі нормальної роботи.

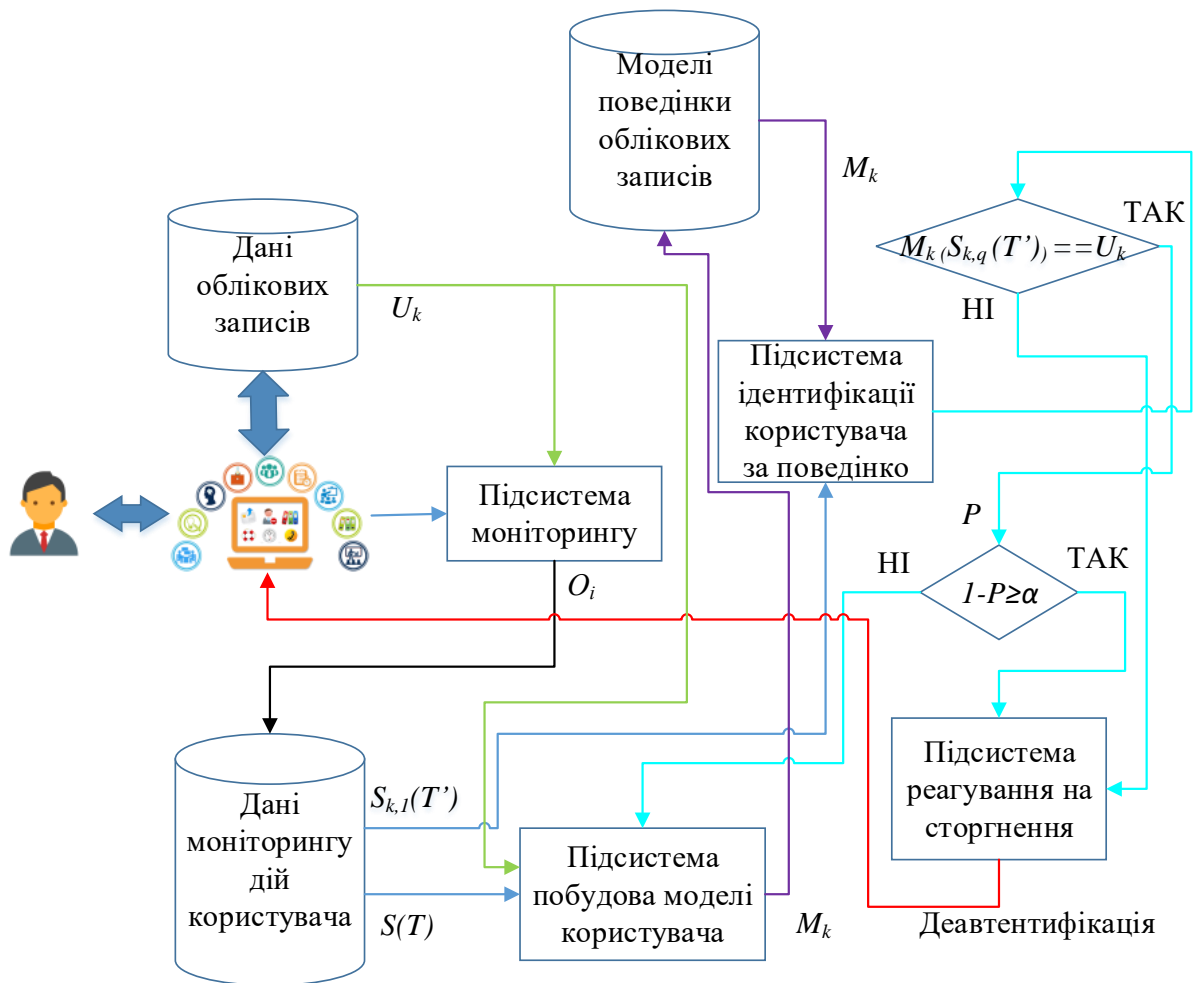


Рисунок 5.2 – Схема процесу додаткового захисту системи від вторгнення за допомогою ідентифікації поведінки користувача

Далі модель зберігається в базі даних для подальшого використання. Після цього підсистема ідентифікації використовуючи дані про поточний сеанс $S_{k,q}(T')$ та модель поведінки користувача M_k оцінює відповідність поточної поведінки користувача його моделі. Далі, якщо поведінка користувача не відповідає його моделі, то відбувається деавтентифікація користувача. Якщо ж поведінка співпадає, то перевіряється значення ймовірності відповідності правильної поведінки $1 - P \geq \alpha$ і в разі чого відбувається перебудова моделі поведінки цього користувача.

5.2 Експериментальне дослідження підходу до ідентифікації користувачів за їх поведінкою в системі

Для проведення експериментального дослідження будемо використовувати дані зі статі [27], які представлені на Kaggle – платформі для змагань з аналітики та передбачувального моделювання.

Ці дані зібрані з проксі-серверів університету Блеза Паскаля. Він складається з 17×10^6 рядків журналів, підключених від більш ніж 3000 користувачів, і містить ідентифікатор користувача, відмітку часу та доменні імена для кожного рядка. При формуванні вибірки застосувалися два типи фільтрів до доменних імен: фільтри чорного списку та фільтри на основі HTTP-запитів. Автори використовували кілька списків доменних імен, щоб видалити всі домени, які розглядаються як рекламні. Також відфільтрували дані за кодом стану, отриманим після простого HTTP-запиту на доменне ім'я. Після цих кроків авторами було отримано 4×10^6 рядків. Автори розділили файл між 3000 користувачами, щоб отримати файли класів. Цей набір даних доступний за адресою <http://fc.isima.fr/~kahngi/cez13.zip>. Дослідження проводилися для 150 користувачів з найбільшою кількістю запитів.

Таким чином будемо будувати модель поведінки користувача, аналізуючи послідовності з кількох веб-сайтів, що були відвідані підряд одним і тим же користувачем, і визначити, чи це Еліс (легітимний користувач), чи зловмисник (інша людина).

Відповідно до схеми процесу побудови моделі поведінки користувача (рисунок 5.1), маючи дані для побудови моделі, потрібно збільшити кількість ознак. Оскільки даних про сайт та мітки часу недостатньо для того, щоб побудувати якісну модель поведінки, потрібно створити більш інформативний набір ознак, використовуючи всі методи, представлені на рисунку 5.1.

Перше, що потрібно – це проаналізувати наші дані на наявність пропущених значень та розподіл сесій Еліс та зловмисника.

Проаналізувавши розподіл сесій, отримали наступні значення: сесій Еліс 2297, а сесій зловмисників 251264. Далі подивимося на наявність пропущених значень. Результат представлено в таблиці 5.1.

Таблиця 5.1 – Аналіз пропущених значень в даних

Ознака	Кількість ненульових значень	Ознака	Кількість ненульових значень	Ознака	Кількість ненульових значень
session_id	253561	site4	244321	time7	237297
site1	253561	time4	244321	site8	235224
time1	253561	site5	241829	time8	235224
site2	250098	time5	241829	site9	233084
time2	250098	site6	239495	time9	233084
site3	246919	time6	239495	site10	231052
time3	246919	site7	237297	time10	231052

Таким чином можна побачити, що маємо нерівномірний розподіл даних та безліч різнотипних даних з пропущеними значеннями. Оскільки з такими даними неможливо побудувати якісну модель поведінки користувача, то створимо на їх основі необхідний список ознак. В якості ознак оберемо: кількість сайтів в сесії, тривалість сесії, день тижня, година початку сесії, година кінця сесії, початок і кінець сесії, хвилина, день місяця, індекс часу. Разом з цим проводимо відразу і нормалізацію даних.

Наступним етапом буде проведено аналіз деяких ознак для демонстрації їх інформативності. Отже побудуємо графіки розподілу окремо для Еліс та зловмисників для деяких ознак. На рисунках 5.3, 5.4 представлено розподіл сесій по дням тижня.

З рисунків 5.3, 5.4 можна побачити, що розподіл сесій по днях у Еліс відрізняється від усіх інших меншою кількістю інформації про середу, суботу і неділю. І дуже багато про понеділок. Тому вважаємо, що будній день краще залишити як фіктивну ознаку і не групувати їх за такою ознакою.

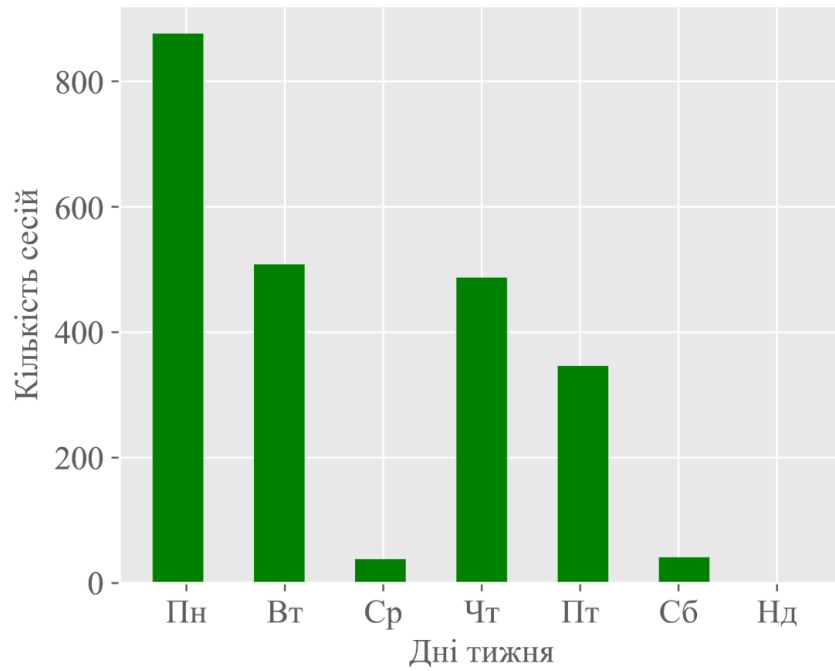


Рисунок 5.3 – Розподіл сесій по днях тижня для користувача Еліс

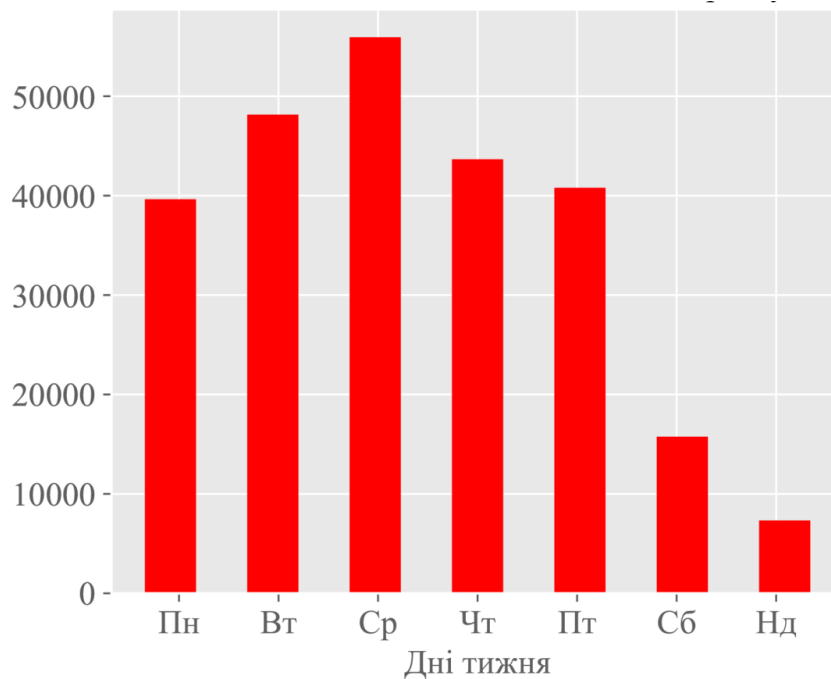


Рисунок 5.4 – Розподіл сесій по дням тижня для інших користувачів

Далі проаналізуємо ще одну важну ознаку – години, коли відбувались сесії. Розподіл даних за годинами представлено окремо для Еліс та інших користувачів (зловмисників) на рисунках 5.5, 5.6 відповідно.

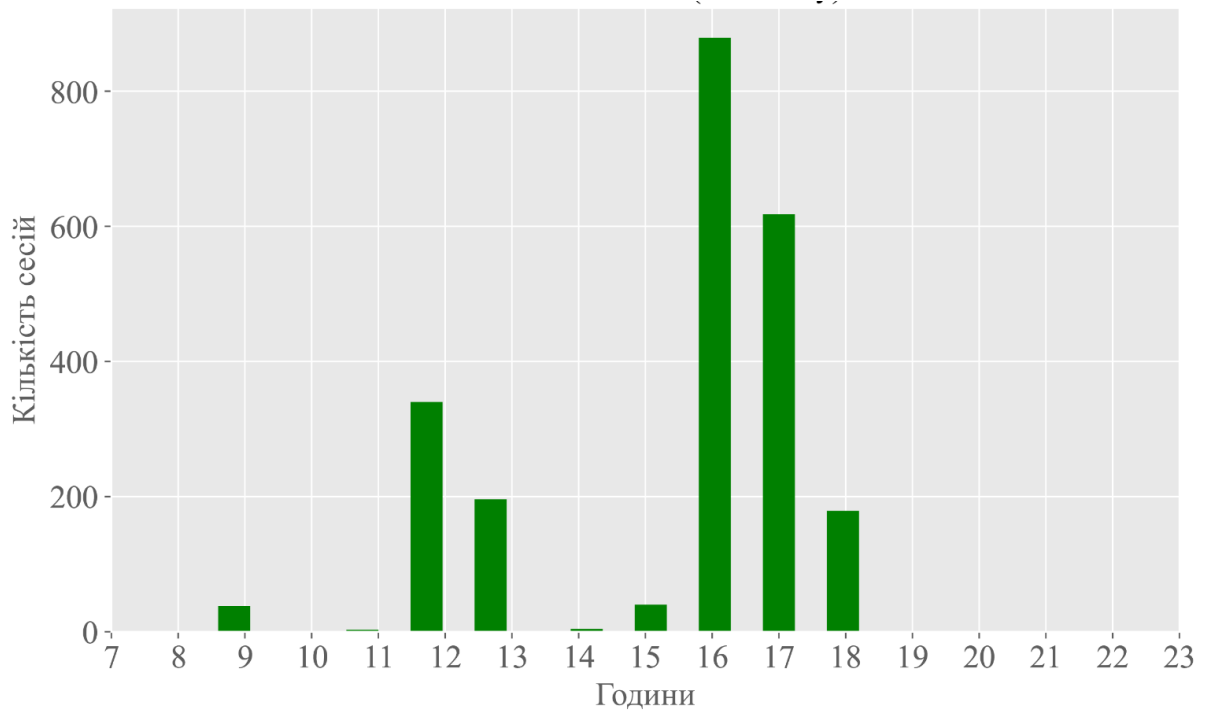


Рисунок 5.5 – Розподіл сесій по годинам для користувача Еліс

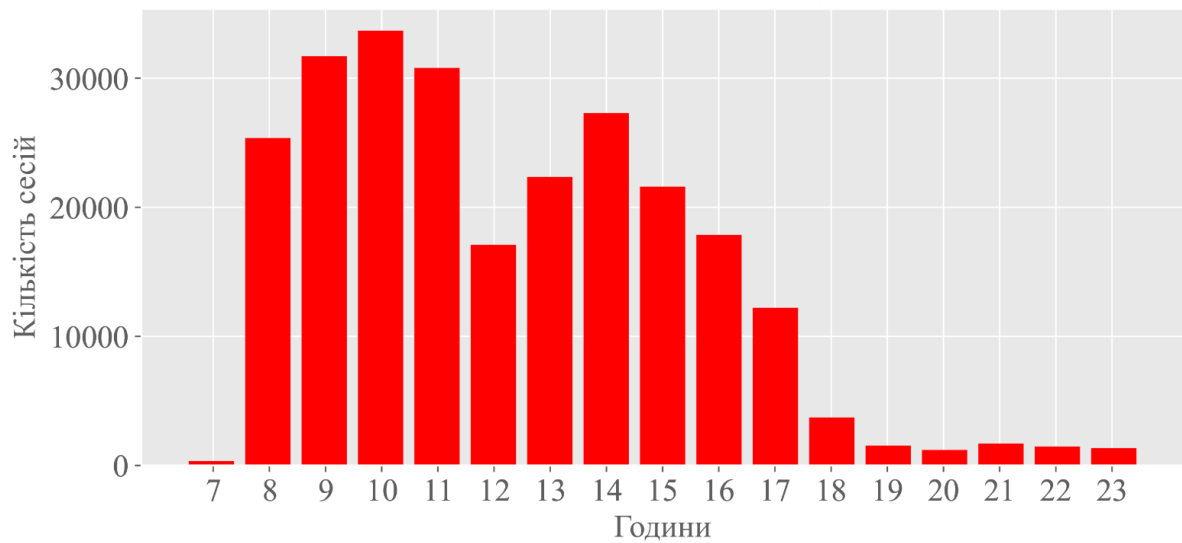


Рисунок 5.6 – Розподіл сесій по годинам для інших користувачів

Щоб краще продемонструвати розподіл тривалості сеансів, було використано логарифм від тривалості сеансу.

Як можна спостерігати з рисунків 5.7, 5.8 є невелика різниця між гістограмами, яка не є критичною. Але цю ознаку можна залишили, оскільки в купі з іншими вона дає невеликий приріст в точності моделі.

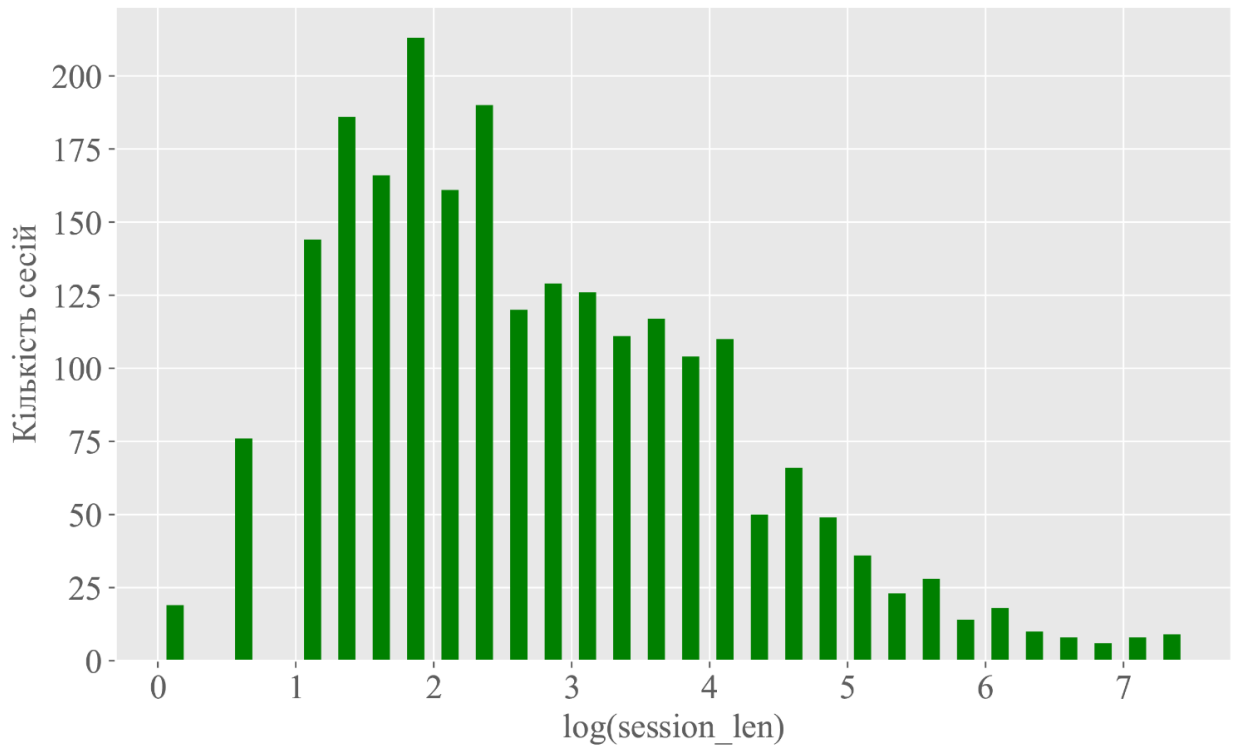


Рисунок 5.7 – Розподіл логарифма від тривалості сесій для Еліс

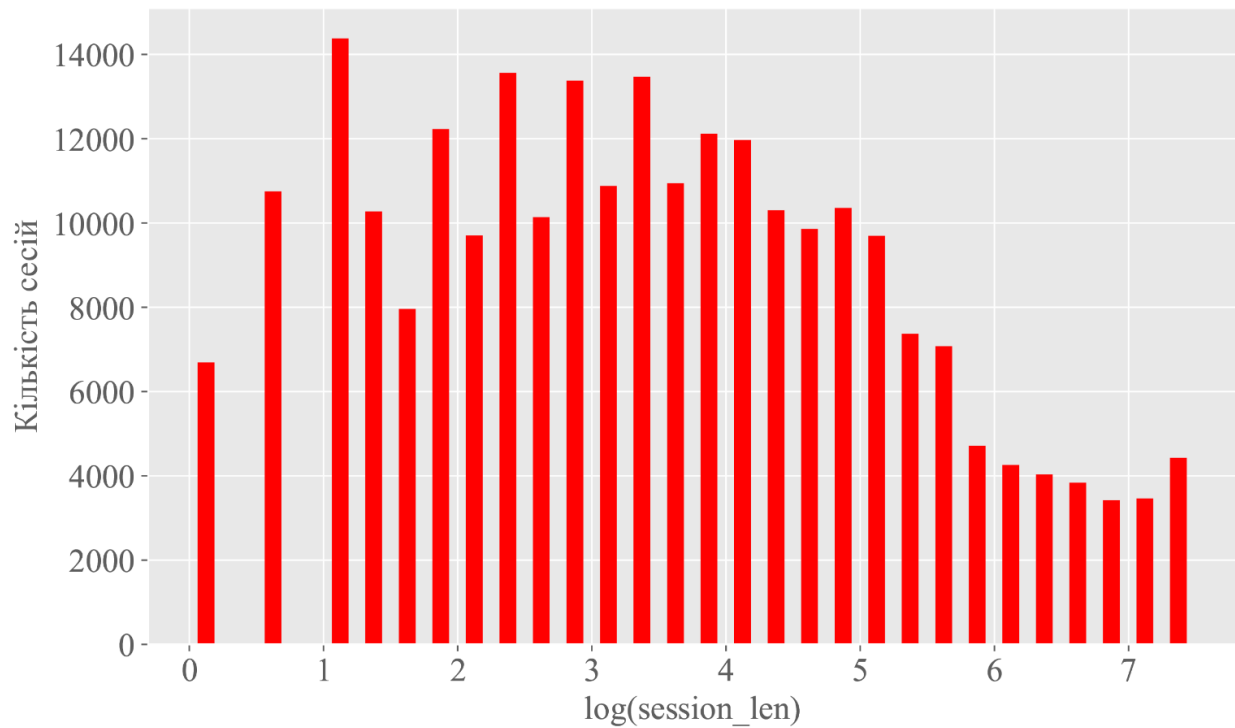


Рисунок 5.8 – Розподіл логарифма від тривалості сесій для інших користувачів

Далі на основі цих ознак за допомогою методів машинного навчання було побудовано чотири моделі поведінки користувача, а саме:

- логістична регресія;
- алгоритм k-найближчих сусідів;
- випадковий ліс;
- метод опорних векторів.

На рисунку 5.9 представлена ROC-крива для логістичної регресії.

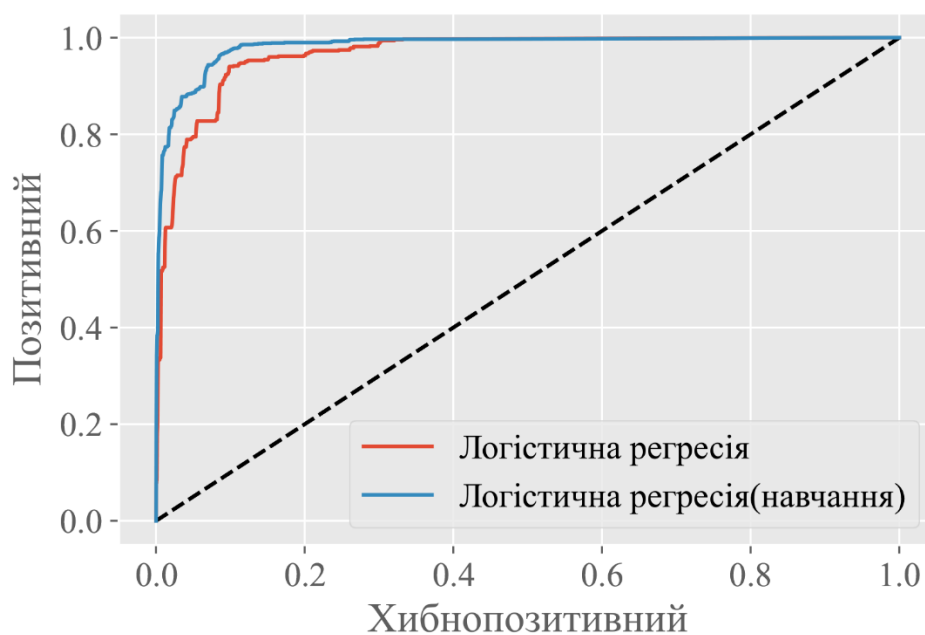


Рисунок 5.9 – ROC-крива для логістичної регресії

Для трьох інших моделей ROC-крива має схожий вигляд і тому дані графіки не приводяться в роботі.

В таблиці 5.2 представлено оцінку якості всіх чотирьох моделей.

Таблиця 5.2 – Оцінка якості моделей

Методи побудови моделей ідентифікації	Оцінка якості на валідаційній вибірці	Оцінка якості на тестовій вибірці
Логістична регресія	0.983091	0.966114
Алгоритм k-найближчих сусідів	0.983124	0.956038
Випадковий ліс	0.983183	0.967171
Метод опорних векторів	0.983519	0.965642
Метод з статті [16]	0.949347	0.876598

Як можна побачити з таблиці 5.2, всі моделі поведінки, які побудовані за допомогою методів машинного навчання, мають високу оцінку якості моделей.

5.3 Аналіз результатів дослідження підходу до ідентифікації користувачів за їх поведінкою в системі

Для оцінки працездатності запропонованого підходу було проведено експеримент з ідентифікації користувача з наступними обмеженнями:

- в якості множини A , яка описує множину програм, що використовуються, будемо розглядати множину доступних доменних імен;
- в якості операцій в програмі, які моніторяться – множина Op буде містити всього одну операцію (відвідування сайту);
- в якості ознак кожної події (відвідування певного сайту) – множина F_i буде містити індекс відвіданого сайту в сесії та мітку часу;
- сесії користувачів $S_{k,q}(T)$ виділені таким чином, що вони не можуть бути довжиною більше ніж 10 сайтів або при T , що дорівнює 30 хвилинам.

Таким чином кортеж $S_{k,q}(T)$ буде мати наступний вигляд:

$$S_{k,q}(T) = \{site_{1,q}^k, time_{1,q}, \dots, site_{10,q}^k, time_{10,q}\}. \quad (5.3)$$

де *site* – це індекси сайтів, а *time* – це мітка часу, коли було відвідано сайт.

В сесіях можуть зустрічатися пропущені значення, це означає, що сесія складається менше, ніж з 10 сайтів. Ці значення будуть замінені нулем.

Проведення експерименту складалося з трьох основних етапів.

Перший етап – аналіз даних моніторингу дій користувача та проведення операцій з фільтрації та розширення набору ознак, за рахунок методів статистики та комбінації наявних даних.

Другий етап – побудова моделей поведінки користувача за допомогою декількох методів машинного навчання.

Третій етап – проведення оцінки якості побудованих моделей поведінки користувача.

Поглянувши на гістограми рисунків 5.3-5.8 можна помітити, що у наборі даних Еліс є два основні фактори. Перший – це нерівномірний розподіл, що ускладнює процес побудови якісної моделі поведінки користувача. Другий – це повторюваність дій Еліс, що дасть можливість методам машинного навчання в конкретних випадках досить швидко знайти патерни поведінки користувача, тим самим підвищити якість та стійкість моделі в цілому.

На рисунках 5.3, 5.4 представлено розподіл сесій по днях тижня для користувача Еліс та інших користувачів. З цих розподілів можна побачити, що день Еліс відрізняється від усіх інших. Це свідчить про те, що це гарна ознака для ідентифікації, але повністю спиратися на неї не можна. Також такий розподіл свідчить, що в реальних системах можлива відсутність деякої інформації та її зашумленість і якісні моделі повинні боротися з цим.

На рисунках 5.5, 5.6 розподіл годин активної роботи користувачів показує, що Еліс працює за досить чітким графіком. Розподіл годин дійсно різний. Можемо спостерігати періоди, коли Еліс взагалі не користується Інтернетом. Це означає те, що дана ознака також досить ефективно може бути використана методами машинного навчання для побудови якісної моделі користувача.

Також однією з ознак, яку було проаналізовано, є тривалість сеансів. Щоб краще продемонструвати розподіл тривалості сеансів, було використано логарифм від тривалості сеансу, де можна спостерігати різницю між сесіями Еліс та інших користувачів. Хоч ця різниця несуттєва, але як можна побачити на рисунку 5.9, її використання також дозволило побудувати якісну модель.

Графік ROC-кривої для логістичної регресії свідчить про те, що побудована модель ідентифікації на основі логістичної регресії дуже гарно виявила патерни поведінки користувача Еліс. Графіки ROC-кривої для інших моделей, які були побудовані за допомогою інших методів машинного навчання, не було представлено в роботі, оскільки вони мають досить схожий вигляд. Це свідчить про те, що було обрано гарний набір ознак для побудови моделі. На відміну від методу ідентифікації поведінки користувача, який представлено в роботі [14], методи машинного навчання досить гарно справляються з побудовою моделі поведінки користувача.

На відміну від підходу, представленого у роботі [15], запропонована модель дозволяє забезпечити комплексний підхід до аналізу поведінки користувача, як під час його роботи (у режимі реального часу), так і після закінчення сеансу (у відкладеному режимі). Це досягається за рахунок того, що обчислювальна складність представлених моделей невелика, окрім моделі, що основана на методі випадковий ліс.

Для підтвердження цього можна проаналізувати дані з таблиці 5.2 і побачити, що всі моделі мають високу якість. З них виділилась модель, яка побудована за допомогою методу випадковий ліс. Але враховуючи, що це ансамблевий метод машинного навчання, то він використовує більше обчислювальних ресурсів, ніж інші методи для побудови моделі поведінки.

Результати з таблиці 5.2 показують, що використовуючи моделі поведінки, які побудовані за допомогою методів машинного навчання, з ймовірністю більше 0,95 зможемо правильно визначити чи відповідає профіль поведінки користувача його даним автентифікації, чи ні. Оскільки представлений підхід використовується, як додатковий фактор ідентифікації

при багатofакторній автентифікації, то відповідно до стандартів тестування NIST представлена якість моделей є прийнятною.

Також при порівнянні розробленого підходу з підходом до ідентифікації користувачів за веб-сесіями, представленому в роботі [16], спостерігаємо, що якість моделі запропонованого підходу на 0,09 краще. Такі результати можна пояснити двома факторами:

- окрім вихідного набору даних з роботи [27], який однаковий в обох випадках, був використаний ще додатковий статистичний аналіз і початковий набір даних було розширено цими даними;

- оскільки основне призначення розробленого підходу до ідентифікації використовується, як додатковий фактор при багатofакторній автентифікації для виявлення нелегітимних користувачів, то модель ідентифікації базується на бінарній класифікації. Це, в свою чергу, дозволяє моделям машинного навчання більш якісно апроксимувати дані для побудови моделі поведінки. Але на відміну від методів, запропонованих у роботі [16], представлений підхід буде не ефективним в інших сферах застосування методів ідентифікації, таких як ідентифікація для рекомендаційних систем або ідентифікація для направленої контекстної реклами, оскільки в таких випадках використовується багатокласова класифікація.

Ці результати підтверджують, що завдання з розробки методів та засобів, які дозволяють ідентифікувати користувачів за їх поведінкою під час сесії, виконано. А представлена в роботі модель побудови профілю поведінки та ідентифікації користувача за нею може бути використана як додатковий засіб забезпечення безпеки інформаційних систем.

Недоліком такого підходу ідентифікації користувачів є те, що він потребує певного часу для збору інформації про поведінку користувача. Це в свою чергу дає можливість зловмиснику здійснити деяку кількість шкідливих операцій.

Для подальшого розвитку даного підходу слід ще приділити увагу розробці інтерактивної моделі поведінки, що враховує динаміку поведінки

користувачів і модуль аналізу трендів, призначений для виявлення можливих змін в поведінці користувачів. Використання цих моделей дозволить передбачати наступні дії користувача, що дозволить скоротити загальний час ідентифікації користувача за його поведінкою.

ВИСНОВКИ

Розроблено функціональну модель процесу забезпечення ідентифікації користувачів за їх поведінкою в системі, що дозволяє створити додаткові засоби захисту користувачів системи у випадку крадіжки їх даних автентифікації. Модель ідентифікації враховує статистичні параметри поведінки користувача (сигнатура користувача), які були отримані впродовж сеансу. Такий підхід дозволив підвищити якість ідентифікації поведінки користувача на 0,09 від класичного методу, розглянутого в роботі [16]. Слід виділити такі переваги запропонованої моделі:

- незалежність від кількості користувачів в системі, оскільки даний підхід використовує модель оцінки поведінки авторизованого користувача і поточні характеристики користувача;

- можливість виявлення прихованих закономірностей в поведінці користувача, це досягається за рахунок використання методів машинного навчання;

- адаптація до зміни поведінки користувачів. Оскільки під час роботи в будь-якій системі користувач поступово навчається, то модель поведінки, яка була побудована в режимі нормальної роботи, буде поступово відрізнятися від реальної поведінки користувача. Саме тому було введено додатковий критерій оцінки допустимого відхилення поточної поведінки від побудованої. І в разі перевищення цього критерію відбувається процес перебудови моделі поведінки користувача на актуальних даних.

Проведено експериментальне дослідження щодо запропонованого підходу ідентифікації користувача за його поведінкою в системі. Побудовані моделі поведінки користувача з використанням методів машинного навчання показали оцінку якості ідентифікації більше 0,95. Результати ідентифікації за допомогою логістичної регресії та методу випадковий ліс показали практично однаковий результат ідентифікації. Отже не обов'язково

використовувати складні моделі при описі поведінки користувача, головне сформувані інформативні ознаки та правильно сформувані валідаційну та тестову вибірки.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. The cyber-threat landscape: The digital rush left many exposed
URL: <https://www.pwc.com/us/en/services/consulting/cybersecurity-risk-regulatory/library/2021-digital-trust-insights/cyber-threat-landscape.html>
2. The Identity Theft Resource Center's Inaugural 2021 Business Aftermath Report Shows the Impacts Identity Crimes Have on Small Businesses
URL: <https://www.idtheftcenter.org/post/the-identity-theft-resource-centers-inaugural-2021-business-aftermath-report-shows-the-impacts-identity-crimes-have-on-small-businesses/>
3. Ghafur S., Kristensen S., Honeyford K. et al. A retrospective impact analysis of the WannaCry cyberattack on the NHS. *npj Digit. Med.* 2, 98 (2019).
<https://doi.org/10.1038/s41746-019-0161-6>
4. Gohwong, Srirath Goi. The State of the Art of Cryptography-Based Cyber-Attacks (July 1, 2019). *International Journal of Crime, Law and Social Issues*. Vol. 6. No. 2. 2019. URL: <http://dx.doi.org/10.2139/ssrn.3546334>
5. Tetskyi A. The method of selecting measures to protect the web application against attacks // *Advanced Information Systems*. 2018. 2(4). pp. 114–118. <https://doi.org/10.20998/2522-9052.2018.4.19>
6. Khan F., Kim J. H., Mathiassen L., Moore R. Data breach management: An integrated risk model // *Information & Management*. 58. 2021. doi:10.1016/j.im.2020.103392.
7. Alemu B., Kumar R., Sinwar D., Raghuwanshi G. Fingerprint based authentication architecture for accessing multiple cloud computing services using single user credential in IOT environments // *Journal of Physics: Conference Series*. Vol. 1714. No. 1. 2021.
8. Beer M.I., Hassan M.F. Adaptive security architecture for protecting RESTful web services in enterprise computing environment. *SOCA* 12. pp. 111–121 (2018). <https://doi.org/10.1007/s11761-017-0221-1>

9. Hussain MI, He J, Zhu N, Sabah F, Zardari ZA, Hussain S, Razque F. AAAA: SSO and MFA Implementation in Multi-Cloud to Mitigate Rising Threats and Concerns Related to User Metadata // *Applied Sciences*. 2021; 11(7):3012. <https://doi.org/10.3390/app11073012>
10. Gavrylenko S., Chelak V., & Vassilev V. Malicious software identification system provision on the basis of context-free grammars. *Advanced Information Systems*. 2018. 2(2). pp. 101–105. <https://doi.org/10.20998/2522-9052.2018.2.17>
11. Xing L, Deng K, Wu H, Xie P, Gao J. Behavioral Habits-Based User Identification Across Social Networks// *Symmetry*. 2019. 11(9):1134. <https://doi.org/10.3390/sym11091134>
12. Wen X., Peng Z., Huang S., Wang S., Yu P.S. (2021). MISS: A Multi-user Identification Network for Shared-Account Session-Aware Recommendation. In: , et al. *Database Systems for Advanced Applications. DASFAA 2021 // Lecture Notes in Computer Science* . 2021. vol 12683. Springer, Cham. https://doi.org/10.1007/978-3-030-73200-4_15
13. Yang, Yinghui C. Web user behavioral profiling for user identification. // *Decision Support Systems*. 2010. 49.3. pp. 261-271.
14. Шаньгин В.Ф. Информационная безопасность компьютерных систем и сетей: учеб. пособие. М.: ИНФРА-М, 2011. 416 с.
15. Коллинз М. Защита сетей. Подход на основе анализа данных / пер. с англ. А.В. Добровольская. М.: ДМК Пресс, 2020. – 308 с.
16. Бурков А. Инженерия машинного обучения / пер. с англ. А.А. Слинкина. М.: ДМКПресс, 2022. 306 с.
17. Billings S.A. Identification of nonlinear systems-a survey. // *IEEE Proceedings D-Control Theory and Applications*. IET, 1980. pp. 272-285.
18. Su X., Xin Y. Chih-Ling T. Linear regression // *Wiley Interdisciplinary Reviews: Computational Statistics*. 2012. Vol. 4.3. pp. 275-294.
19. LaValley M. P. Logistic regression // *Circulation*. 2008. 117(18). pp. 2395-2399.

20. Kramer O. K-nearest neighbors // Dimensionality reduction with unsupervised nearest neighbors. Springer, Berlin, Heidelberg, 2013. pp. 13-23.
21. Quinlan J. Ross. Induction of decision trees // Machine learning. 1986. 1.1. pp. 81-106.
22. Thorsten J. Svmlight: Support vector machine // SVM-Light Support Vector Machine. University of Dortmund. 1999. 19.4. p. 25. <http://svmlight.joachims.org/>
23. Zell A. Simulation Neuronaler Netze [Simulation of Neural Networks] (German). Addison-Wesley. 1994. 624 p.
24. Martovytskyi V., Ruban I., Sievierinov O., Nosyk A., Lebediev V. Mathematical Model of User Behavior in Computer Systems. 2020 IEEE International Conference on Problems of Infocommunications. Science and Technology (PIC S&T). 2020. pp. 127-131.
25. Ruban I.V., Martovytskyi V.O., Kovalenko A.A., Lukova-Chuiko N.V. Identification in Informative Systems on the Basis of Users' Behaviour. 2019 IEEE 8th International Conference on Advanced Optoelectronics and Lasers (CAOL). 2019. pp. 574-577.
26. Запорожець Н.О., Запорожець О.В., Мартовицький В.О. Ідентифікація користувачів інформаційної комп'ютерної системи за їх поведінкою за допомогою методів машинного навчання // Тези доповідей десятої міжнародної науково-технічної конференції «Проблеми інформатизації», Черкаси– Баку – Бельсько-Бяла – Харків, 24-25 листопада 2022 р. – Т. 2. – С. 59.
27. Kahn G., Yannick L., Raynaud O. A tool for classification of sequential data. ECAI 2016 (Workshop FCA4AI). 2016.