Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____Комп'ютерних наук_____
(повна назва)
Кафедра _____Інформаційних управляючих систем_____
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

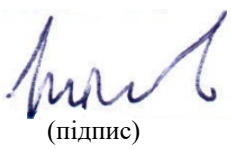
рівень вищої освіти _____другий (магістерський)_____
__Дослідження методів формування знань в ІС підтримки прийняття рішень_
_____
(тема)

Виконав:

студент _2_ курсу, групи _ІУСТм-21-1_____

_____Олена БУНЕЦЬКА _____
(власне ім'я, прізвище)
Спеціальність 122 Комп'ютерні науки_____
(код і повна назва спеціальності)
Тип програми ____освітньо-професійна___
(освітньо-професійна або освітньо-наукова)
Освітня програма Інформаційні управяючі системи та технології_____
(повна назва освітньої програми)
Керівник _професор кафедри ІУС Сергій ЧАЛИЙ_____
(посада, власне ім'я, прізвище)

Допускається до захисту

Зав. кафедри  ___Костянтин ПЕТРОВ__
(підпис) (власне ім'я, прізвище)

2022 р.

Харківський національний університет радіоелектроніки

Факультет_____Комп'ютерних наук_____

Кафедра_____Інформаційних управляючих систем_____

Рівень вищої освіти_____другий (магістерський)_____

Спеціальність_____122 Комп'ютерні науки_____
(код і повна назва)

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

Освітня програма___Інформаційні управляючі системи та технології_____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)
«21»листопада 2022 р.__

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові_____Бунецькій Олені Олександрівні_____
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів формування знань в ІС підтримки прийняття рішень_____
затверджена наказом університету від 14__листопада__2022_ р. № _1490Ст_
2. Термін подання студентом роботи до екзаменаційної комісії 14.12.2022 р.
3. Вихідні дані до роботи Класифікація DSS за способом підтримки, структура СППР на основі знань, узагальнена схема процесу прийняття рішень, задачі процесу підтримки прийняття рішень, методи формування знань, схема взаємозв'язку правил та подій, узагальнена послідовність етапів побудови знань, інформаційна технологія автоматизованого формування знань з використанням подій, екранна форма прототипу_____
4. Перелік питань, що потрібно опрацювати в роботі Дослідження систем підтримки прийняття рішень, аналіз процесу підтримки прийняття рішень, методи формування знань, постановка задачі, опис методу автоматизованого формування знань, теоретичне вдосконалення методу формування знань на основі файлів логів, опис технології та особливості реалізації, опис програмного забезпечення, яке використовується для розробки, експериментальна перевірка_____

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Дослідження систем підтримки прийняття рішень	10.10.2022 – 20.10.2022	Виконано
2	Аналіз процесу підтримки прийняття рішень	21.10.2022 – 25.10.2022	Виконано
3	Аналіз методів формування знань	26.10.2022 – 30.10.2022	Виконано
4	Постановка задачі	01.11.2022 – 10.11.2022	Виконано
5	Опис методу автоматизованого формування знань	11.11.2022 – 15.11.2022	Виконано
6	Теоретичне вдосконалення методу формування знань	16.11.2022 – 19.11.2022	Виконано
7	Опис технології	20.11.2022 – 25.11.2022	Виконано
8	Опис програмного забезпечення	26.11.2022 – 01.12.2022	Виконано
9	Експериментальна перевірка	01.12.2022 – 07.12.2022	Виконано

Дата видачі завдання _21_ «листопада» 2022 р.

Студент _____
(підпис)

Керівник роботи _____ професор кафедри ІУС Сергій ЧАЛИЙ
(посада, власне ім’я, прізвище)

РЕФЕРАТ

Пояснювальна записка до магістерської кваліфікаційної роботи містить:70с., 4 розділи, 15 рис., 4 табл., 30 джерел.

АВТОМАТИЗОВАНЕ ФОРМУВАННЯ ЗНАНЬ, БАЗА ЗНАНЬ, ЗНАННЯ, СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ

У роботі виконано огляд методів автоматизованого формування знань в процесі підтримки прийняття рішень для задачі виявлення атак на комп’ютерну систему. Проаналізовано існуючі методи формування знань. На підставі проведеного аналізу запропоновано покращений метод автоматизованого формування знань завдяки виявленню подій у файлах журналів[1][2].

В ході дослідження отримані такі результати: визначена класифікація існуючих систем підтримки прийняття рішень, проаналізовано задачі систем прийняття рішень на базі знань, визначено методи формування знань, проаналізовано існуючий метод автоматизованого формування знань, запропоновано удосконалення у процес вилучення знань із файлів журналів; проведено експериментальну перевірку по покращеному методу.

ABSTRACT

Explanatory Note to master certification work contains 70 pages, 4 sections, 15 pictures, 4 tables, 30 sources.

AUTOMATED KNOWLEDGE FORMATION, DECISION-MAKING SUPPORT SYSTEMS, KNOWLEDGE, KNOWLEDGE BASE

The paper reviewed the methods of automated knowledge formation in the process of supporting decision-making. Existing methods of knowledge formation are analyzed. On the basis of the conducted analysis, an improved method of automated knowledge formation thanks to the detection of events in log files is proposed.

During the study, the following results were obtained: the classification of existing decision support systems was determined, the tasks of knowledge-based decision-making systems were analyzed, the methods of knowledge formation were determined, the existing method of automated knowledge formation was analyzed, improvements to the process of extracting knowledge from log files were proposed; experimental verification was carried out using the improved method

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

ІС – інформаційна система;

ОПР – особа, що приймає рішення;

СППР – система підтримки прийняття рішень;

СУБД – система управління базами даних;

ШІ – штучний інтелект;

DSS – decision support system;

GSS – group support system;

IT – Information technology;

JSON – JavaScript Object Notation;

Log file – файл журналу;

KBDSS – Knowledge Based Decision Support System;

KM - Knowlage management;

MDSS – Model Decision Support System;

MIS – Management information system;

XLM – файли електронних таблиць.

ВСТУП

Інформаційні технології у сучасному світі стрімко та постійно розвиваються, залучаються до вирішення все більш широкого спектру складних задач. Одною із найскладніших людських задач можна вважати задачу прийняття рішення [3]. У процесі прийняття рішення залучається чимало ресурсів, перш за все із визначення необхідних умов та вхідних даних для формування множини альтернатив. Прийняття рішень – це процес прийняття рішень шляхом визначення рішення, збору інформації та оцінки альтернативних рішень.

Використання поетапного процесу прийняття рішень може допомогти приймати більш обдумані та продумані рішення шляхом організації відповідної інформації та визначення альтернатив. Такий підхід збільшує шанси на те, що буде обрана найбільш задовільна альтернатива.

Окрім управлінських рішень виникає потреба у забезпеченні безпеки системи. Безпека — це практика захисту систем, мереж і програм від цифрових атак. Ці кібератаки зазвичай спрямовані на доступ, зміну або знищення конфіденційної інформації; вимагання грошей у користувачів за допомогою програм-вимагачів; або переривання звичайних бізнес-процесів.

Впровадження ефективних заходів кібербезпеки сьогодні є особливо складним, оскільки пристроїв більше, ніж людей, а зловмисники стають все більш інноваційними.

Існуючі методи підтримки рішень з виявлення атак базуються на використанні патернів, що не дає можливість виявити нові нетипові атаки.

Вирішення цієї проблеми пов’язано з використанням знань, які дають можливість сформувати нові патерни і, тим самим, виявити нові нетипові атаки. Такі знання можуть бути сформовані на основі аналізу логів що

відображують поведінку існуючих програм у нормальному режимі і у випадку атаки.

Предмет дослідження – методи формування знань. Мета роботи – дослідження та удосконалення методів формування знань з підтримки прийняття рішень щодо виявлення атак на комп’ютерну систему.

Наукова новизна – вдосконалено метод формування знань завдяки виділенню правил шляхом поєднання подій «кожної з кожною». Як практичний результат було розроблено прототип програмного модуля для вирішення задачі формування знань.

1. ДОСЛІДЖЕННЯ ПІДХОДІВ ДО ФОРМУВАННЯ ЗНАНЬ ДЛЯ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ

1.1 Дослідження процесу підтримки прийняття рішень

Системи підтримки прийняття рішень (далі СППР або DSS) – це системи призначені для допомоги людям у прийнятті складних рішень завдяки обробці обсягів інформації, отриманої з інших інформаційних систем [4].

Система підтримки прийняття рішень використовується для підтримки рішень, суджень і планів дій в організації чи бізнесі. DSS просіває й аналізує величезні масиви даних, збираючи вичерпну інформацію, яку можна використовувати для вирішення проблем і прийняття рішень.

Система підтримки прийняття рішень збирає та аналізує дані, синтезуючи їх для створення вичерпних інформаційних звітів. Таким чином, як інформаційна програма, DSS відрізняється від звичайної операційної програми, функцією якої є лише збір даних [5].

DSS може бути або повністю комп’ютеризованою, або працювати з людьми. У деяких випадках він може поєднувати обидва. Ідеальні системи аналізують інформацію та фактично приймають рішення за користувача. Принаймні, вони дозволяють користувачам приймати більш обґрунтовані рішення в більш швидкому темпі.

СППР використовує інформацію, винятки, шаблони та тенденції за допомогою аналітичних моделей. Система підтримки прийняття рішень допомагає у прийнятті рішень, але не обов’язково сама приймає рішення. Особи, які приймають рішення, збирають корисну інформацію з необроблених даних, документів, особистих знань або бізнес-моделей для виявлення та вирішення проблем і прийняття рішень.

Система підтримки прийняття рішень може представляти інформацію графічно та може включати експертну систему або штучний інтелект (ШІ) [6].

Він може бути спрямований на керівників підприємств або іншу групу працівників розумової праці.

Типовою інформацією, яку може збирати та представляти програма підтримки прийняття рішень, є:

– доступ до всіх інформаційних активів, включаючи успадковані та реляційні джерела даних;

– цифри порівняльних даних;

– прогнозовані цифри на основі нових даних або припущень;

– наслідки різних альтернативних рішень, враховуючи минулий досвід у конкретному контексті.

Існує декілька видів систем підтримки прийняття рішень, їх класифікація базується на основі чого буде працювати система.

Види DSS в залежності від того способу підтримки поділяюся на різні категорії. Які наведено на рисунку 1.1.

Рисунок 1.1 – Класифікація DSS за способом підтримки

СППР на основі комунікативих підходів до прийняття – це тип СППР, який наголошує на комунікаціях, співпраці та спільній підтримці прийняття рішень. Проста дошка оголошень або ланцюгова електронна пошта – найпростіший рівень функціональності. Поширені запитання особ, що приймають рішення (ОПР) визначають групове програмне забезпечення як «програмне та апаратне забезпечення для спільних інтерактивних середовищ», призначене для підтримки та розширення групової діяльності. Групове програмне забезпечення є підмножиною ширшої концепції під назвою «Спільні обчислення» [7]. DSS дозволяє двом або більше людям спілкуватися один з одним, обмінюватися інформацією та координувати свою діяльність. Прикладами засобів підтримки групи є: аудіоконференції, дошки оголошень і веб-конференції, обмін документами, електронна пошта, програмне забезпечення для зустрічей обличчям до лиця, що підтримується комп’ютером, та інтерактивне відео.

DSS, керована темпоральними даними, наголошувала на доступі та маніпулюванні даними з урахуванням конкретних завдань за допомогою загальних інструментів [8]. Хоча DSS також надає підприємствам елементарну функціональність, вона значною мірою покладається на дані часових рядів. DSS може підтримувати прийняття рішень у низці ситуацій. DSS на основі даних надає користувачам доступ до великої кількості внутрішніх і зовнішніх даних. Ця DSS надсилає запит до бази даних за допомогою Інтернету, зовнішнього сервера або мейнфрейму компанії. Він покладається на інтелектуальний аналіз даних, щоб отримати шаблони та інформацію про дані, що оцінюються. Користувачі покладаються на системи підтримки прийняття рішень на основі даних, щоб приймати рішення щодо бізнесу, запасів і продуктів. Керівники можуть вважати системи підтримки прийняття рішень на основі даних найбільш корисними під час аналізу поточних і історичних даних для звітування про стан відділу чи бізнесу [9]. Виконавчі директори, менеджери та персонал можуть використовувати DSS на основі даних.

DSS, з використанням моделі об’єкту управління, наголошує на доступі та маніпулюванні моделлю, наприклад статистичними, фінансовими моделями, моделями оптимізації та/або моделюванням. Прості статистичні та аналітичні інструменти забезпечують найпростіший рівень функціональності. Деякі системи OLAP, які дозволяють комплексний аналіз даних, можуть бути класифіковані як гібридні системи DSS, що забезпечують як моделювання, так і пошук даних і функції узагальнення даних. Загалом СППР, орієнтовані на моделі, використовують складні фінансові моделі, моделі моделювання, оптимізації або багатокритеріальні моделі для забезпечення підтримки прийняття рішень. Керовані моделлю СППР використовують дані та параметри, надані особами, які приймають рішення, щоб допомогти тим, хто приймає рішення, аналізувати ситуацію, але вони зазвичай не потребують інтенсивних даних, тобто дуже великі бази даних зазвичай не потрібні для СППР, керованих моделями. Такі системи також називають модельно-орієнтованими або модельно-орієнтованими системами підтримки прийняття рішень. Поведінкові та технічні дослідження СППР, керовані моделями, потребують вирішення багатьох невирішених проблем, пов’язаних із побудовою конкретних кількісних моделей, зберіганням і пошуком даних, необхідних для різних типів моделей, передачею параметрів між моделями та іншими компонентами СППР, взаємодією багатьох учасників у використання моделі та виявлення цінностей, а також вплив альтернатив дизайну інтерфейсу користувача на керовану моделлю ефективність DSS та простоту використання. Крім того, дослідники повинні провести дослідження. Математичні та аналітичні моделі є домінуючим компонентом у керованій моделями системі підтримки прийняття рішень. Якщо для розуміння ситуації потрібна модель, тоді СППР, керована моделлю, потенційно може надати необхідне представлення менеджерам [10]. Аналітики СППР можуть створювати широкий спектр альтернативних СППР на основі моделі. Тож фактично побудова DSS на основі моделі передбачає вирішення низки важливих питань проектування та розробки. Моделі можуть допомогти

менеджерам зрозуміти фінансові, маркетингові та багато інших бізнес-рішень. Однією з головних проблем, яку необхідно вирішити, є мета запропонованої СППР, керованої моделлю. Чи має на меті допомогти у прийнятті рішень щодо кредитування та позичання, складанні бюджету чи прогнозуванні попиту на продукт? Чи буде система використовуватися регулярно в процесі прийняття рішень чи як частина спеціального дослідження? Кожна СППР, що керується моделлю, повинна мати чітко визначену та конкретну мету. Щоб досягти конкретної мети системи, іноді використовується більше ніж один тип моделі при побудові керованої моделлю DSS [11]. Отже, друге питання полягає в тому, які моделі повинні бути включені в конкретну систему.

СППР, керовані знаннями, або «базами знань» охоплюють широкий спектр систем. Вони призначені для користувачів в організації, які їх налаштовують, але можуть також включати інших, хто взаємодіє з організацією, наприклад, споживачів. В основному такі системи використовується для надання управлінських порад або вибору продуктів чи послуг. Тож для великих організацій СППР стають все більш популярними для щоденної роботи. Система є еквівалентним синонімом інформаційних систем управління (MIS). Більшість імпортованої інформації у систему використовуються в таких рішеннях, як аналіз даних. Успішна підтримка прийняття управлінських рішень критично залежить від наявності інтегрованої високоякісної інформації, організованої та представленої вчасно та легко зрозумілою формою.

Знання у формі процедур, найкращих практик, бізнес-правил, експертних знань, фактів у контексті та оброблених даних можуть зберігатися в логічних структурах, доступних комп’ютерам. Логічні структури в сховищі знань для зберігання знань аналогічні системі таблиць, які реалізують зберігання даних у сховищі даних. Знання застосовуються через багаторівневе представлення, яке можуть прочитати як люди, так і машини. Це представлення також є системним виконуваним файлом, який є переносним і

може запускатися на комп’ютері, щоб допомогти приймати рішення та виконувати відповідні дії.

СППР, що базуються на основі пошуку документів, здійснюють пошук і відбір необхідних неструктурованих документів і даних у базах даних і в Інтернеті. DSS на основі документів є відносно новою сферою підтримки прийняття рішень. Document-Driven DSS зосереджено на пошуку та управлінні неструктурованими документами. Документи можуть приймати різні форми, але їх можна розділити на три категорії: усні, письмові та відео. Прикладами усних документів є розмови, які транскрибуються; відео може бути роликом новин або телевізійною рекламою; письмовими документами можуть бути письмові звіти, каталоги, листи від клієнтів, записки та навіть електронна пошта [12].

Для вирішення задач виявлення атак можна використовувати системи підтримки прийняття рішень на основі знань. У центрі таких систем знаходяться знання, схематично структуру KBDSS зображено на рисунку 1.2.

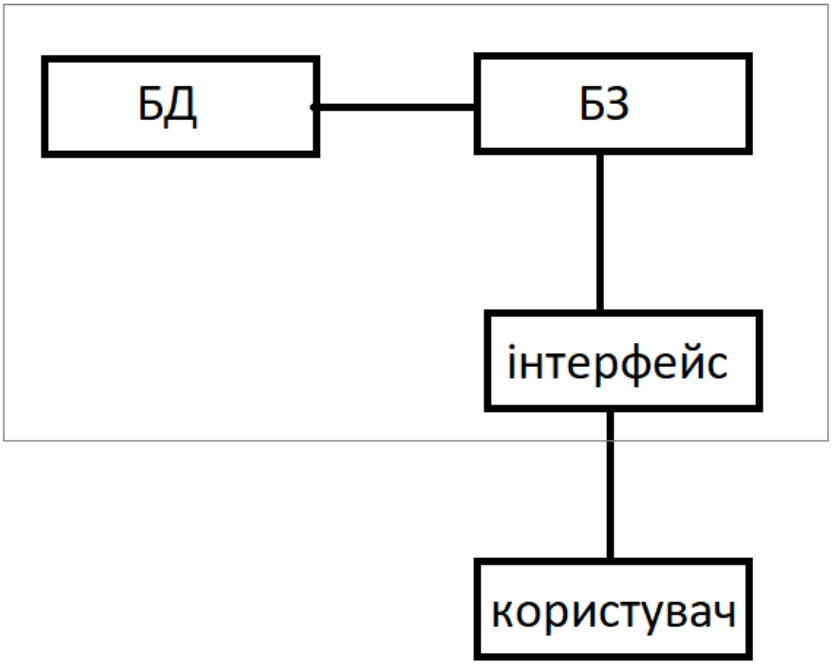


Рисунок 1.2 – Структура СППР на основі знань

DSS на основі моделі можна використовувати для сприяння прийняттю рішень у різноманітних ситуаціях. Це може допомогти менеджерам у:

- кредитні та кредитні рішення;
- прогнозування попиту на товар;
- бюджетні рішення;
- маркетингові рішення;
- рішення щодо прогнозування виробництва;
- рішення про розподіл ресурсів;
- планування проекту;
- інвестиційні рішення.

Моделювання – це процес визначення відповідної моделі для перспективної системи підтримки прийняття рішень, керованої моделлю. Він проходить наступні фази в хронологічному порядку, починаючи з визначення проблеми [13]. Після завершення моделювання життєво важливо перевірити вибрану модель, щоб переконатися, що вона добре працює та дає відповідні результати. Перевірка моделі виконується шляхом порівняння результатів моделі та фактичної поведінки події. Як пояснює сама назва, керована моделлю система підтримки прийняття рішень використовує модель для вирішення проблем або допомоги в прийнятті рішень. Модель може бути статистичною, фінансовою, математичною, аналітичною, імітаційною або оптимізаційною. DSS, керована моделлю, може використовувати одну модель або комбінацію двох або більше моделей, залежно від конкретних потреб користувачів. Прості моделі забезпечують базову функціональність, а комбінація двох або більше моделей дозволяє користувачам аналізувати складні дані.

DSS, керовані моделлю, зазвичай не потребують інтенсивних даних. Скоріше вони використовують параметри, введені особами, які приймають рішення, і допомагають їм проаналізувати ситуацію. Вони генерують оптимальні рішення, які відповідають обмеженням часу та ресурсів. Обсяг

DSS на основі моделі величезний і може бути додатково розширений шляхом інтеграції веб-додатків.

При розробці MDSS важливо розуміти інструменти моделювання та аналізу, їхню роботу та сферу застосування. Побудова DSS на основі моделі вимагає значного рівня досвіду. Менеджери та аналітики DSS повинні тісно співпрацювати, щоб розробити ефективну систему, яка є масштабованою, універсальною та легкою для інтеграції та використання.

Системи, що доставляють інформацію, для всього підприємства, які надаються в сховищі даних, можна використовувати та розширювати для створення сховища знань [14]. Основна мета сховища – надати особі, що приймає рішення, платформу інтелектуального аналізу, яка вдосконалює всі етапи процесу управління знаннями. Архітектура сховища знань не тільки сприятиме збору та кодуванню знань, але також покращить пошук і обмін знаннями в організації. Щоб зрозуміти, проаналізувати та, зрештою, використати величезну кількість даних, підприємства використовують технології інтелектуального аналізу для пошуку величезної кількості даних для отримання життєво важливої інформації. Інструменти інтелектуального аналізу, такі як інтелектуальний аналіз даних, інтелектуальний аналіз тексту та веб-майнінг, використовуються для пошуку прихованих знань у великих базах даних або в Інтернеті. Оптимізація процесів проводиться з тих пір, як з'явилися процеси для вдосконалення. Кілька різних сценаріїв можуть призвести до бажання компанії проаналізувати й оптимізувати існуючі процеси, наприклад перехід від звичайного бізнесу до онлайн-комерції, ініціатив зі скорочення витрат або трансформації IT. Інтелектуальний аналіз процесів – це застосування методів інтелектуального аналізу даних до бізнес-процесів, щоб отримати нове розуміння бізнес-процесів і контролювати їх на основі їх IT-сліду.

Зв'язок між системами прийняття рішень (DSS) і управління знаннями (KM) є областю інтенсивних наукових досліджень. Історія еволюції СППР інформує нас про те, як «обробка даних» перетворилася на інформаційні

системи управління (MIS) і як вона пізніше трансформувалася в сучасну СППР завдяки синергії з управлінням знаннями. Найбільш важливе і часте застосування концепцій КМ очевидно в системах підтримки груп (GSS). Важливо відзначити, що DSS, а точніше GSS, сприяють більш ефективному управлінню знаннями (КМ) на кожному з чотирьох кроків у циклі КМ. Зв'язок між УЗ і стратегією також є ще однією темою для ретельного вивчення та ретельного вивчення. Знання є ключовим чинником організації для побудови сталої та успішної стратегії. Володіння правильною інформацією разом зі знанням про те, як правильно її використовувати, є одним із ключів до успіху [15]. Це означає, що концепція управління знаннями настільки ж важлива для стратегічного управління, як і будь-яка інша діяльність. DSS завдяки покращенню КМ також знайшов застосування для того, щоб дозволити особам, які приймають рішення, приймати більш обґрунтовані стратегічні рішення.

Основною метою використання DSS є представлення інформації клієнту в легкому для розуміння вигляді. Система DSS є вигідною, оскільки її можна запрограмувати для створення багатьох типів звітів, які базуються на специфікаціях користувача. Наприклад, DSS може генерувати інформацію та виводити її графічно, як у вигляді стовпчастої діаграми, яка представляє прогнозований дохід, або у вигляді письмового звіту.

Оскільки технологія продовжує розвиватися, аналіз даних більше не обмежується великими громіздкими мейнфреймами. Оскільки DSS – це, по суті, програма, її можна завантажити на більшість комп’ютерних систем, будь то настільні комп’ютери чи ноутбуки. Деякі програми DSS також доступні через мобільні пристрої [16].

Гнучкість DSS надзвичайно корисна для користувачів, які часто подорожують. Це дає їм можливість бути добре поінформованими в будь-який час, надаючи їм можливість приймати найкращі рішення для своєї компанії та клієнтів у дорозі чи навіть на місці.

1.2 Аналіз задач підтримки прийняття рішень при виявленні атак комп’ютерних системах

Підтримка прийняття рішень має вирішальне значення для ведення бізнесу, яке стикається з великою кількістю проблем, що вимагають прийняття рішень.

Прийняття рішень – це процес прийняття рішень шляхом визначення рішення, збору інформації та оцінки альтернативних рішень.

Системи підтримки прийняття рішень є популярними інструментами, які допомагають приймати рішення в організації. Важливість управління знаннями (KM) також визнається через його внесок у прийняття рішень в організаціях. СППР були об’єднані з системами управління знаннями та еволюціонували від попередніх концепцій «обробки даних» та інформаційних систем управління (MIS) до їх поточної форми як незамінної допомоги ІС для прийняття рішень. Найбільш поширене застосування синергії DSS і KM можна знайти в системах підтримки груп. Функції групи підтримки GSS, такі як мозковий штурм, оцінка ідей і комунікаційні засоби. Також обговорюються зв'язки між управлінням знаннями (KM) і стратегічним управлінням бізнесом. КМ має потенціал, щоб дозволити підприємствам отримати конкурентну перевагу завдяки детальному вивченню факторів навколишнього середовища. Таким чином, СППР автоматично розглядаються як ключові допоміжні функції, оскільки вони дозволяють спеціалістам із знаннями та особам, які приймають рішення, приймати добре обґрунтовані рішення шляхом ефективного вивчення напів- та погано структурованих змінних факторів зовнішнього середовища.

Система підтримки прийняття рішень (DSS) – це інформаційна система, яка допомагає бізнесу приймати рішення, які вимагають оцінки, рішучості та послідовності дій. Інформаційна система допомагає керівництву середнього та високого рівня організації, аналізуючи величезні обсяги неструктурованих

даних і накопичуючи інформацію, яка може допомогти вирішити проблеми та допомогти в прийнятті рішень. DSS є або автоматизованим, або автоматизованим, або комбінацією обох.

Система підтримки прийняття рішень створює докладні інформаційні звіти шляхом збору та аналізу даних. Таким чином, DSS відрізняється від звичайної операційної програми, метою якої є збір даних, а не їх аналіз.

В організації DSS використовується відділами планування, такими як операційний відділ, які збирають дані та створюють звіти, які можуть використовуватися менеджерами для прийняття рішень. В основному DSS використовується для прогнозування продажів, для даних про запаси та операції, а також для представлення інформації клієнтам у зрозумілій для розуміння формі.

Теоретично СППР можна використовувати в різних сферах знань від організації до управління лісами та медицини. Одним із основних застосувань СППР в організації є звітність у реальному часі. Це може бути дуже корисним для організацій, які беруть участь у управлінні запасами точно вчасно .

Крім того можна використати СППР для виявлення зловмисницьких атак.

Метою виявлення вторгнень є визначення поведінки мережі. Через швидкий розвиток моделі атак необхідно розробити систему, яка може оновлюватися відповідно до нових атак. Також рівень виявлення має бути високим, оскільки рівень атак у мережі дуже високий. У відповідь на цю проблему зазвичай застосовується алгоритм на основі патернів (або шаблонів). Більшість напівконтрольованих алгоритмів, які використовуються для виявлення вторгнень, є бінарними класифікаторами.

У для виявлення атак потрібні дані в реальному часі, щоб запобігти затримкам у виявленні атак [17].

Виявлення потенційних атак на систему є важливим кроком у розробці безпечних систем, оскільки виявлені атаки визначатимуть основні вимоги безпеки. Поширеність соціально-технічних систем робить аналіз атак

особливо складним. Ці системи складаються з людей і організацій, їх програмних систем, а також фізичної інфраструктури. Таким чином, ретельний аналіз атак повинен враховувати стратегічні (соціальні та організаційні) аспекти залучених людей і організацій, а також технічні аспекти, що впливають на програмні системи та фізичну інфраструктуру, що вимагає великого обсягу знань безпеки, які важко отримати. Зокрема використовується систематичний метод моделювання шаблонів атак щоб напівавтоматично вибирати та застосовувати такі шаблони. Використовуючи шаблони як частину систематичного процесу, що підтримується інструментами, можна ефективно реалізувати стратегії атак і визначити реалістичні альтернативні атаки.

Системи підтримки прийняття рішень на основі знань – це системи, розроблені для забезпечення більш точного прийняття рішень шляхом ефективного використання своєчасних і відповідних даних, інформації та управління знаннями для індустрії конвергенції. Ці системи стосуються прийняття рішень на основі відповідних знань, які базуються на штучному інтелекті та на застосуванні інформаційних і комунікаційних технологій. Крім того, ці системи підтримують прийняття рішень за допомогою методів прогнозування та рекомендацій. Залежно від критеріїв існують різні класифікації. На основі знань, які використовуються для дедукції, дані класифікуються на системи, засновані на знаннях, з використанням знань, визначених словником, і системи, що не базуються на знаннях, з використанням машинного навчання та методів багатовимірного статистичного розпізнавання образів.

Системи підтримки прийняття рішень здатні перетворювати бізнес-дані в інформацію, чітко представлену у звітах і на інформаційних панелях, на основі якої управлінські команди та менеджери можуть приймати рішення. Таким чином, дані, що зберігаються в бізнес-системах, стають основою прийняття рішень, яка підтримує та покращує роботу.

Хоча підтримка прийняття рішень може базуватися на різних формах штучного інтелекту на основі правил, найчастіше вона пов’язана із системами вилучення та представлення даних. Підтримку прийняття рішень часто називають «діловою аналітикою» [18].

На основі аналізу технологій і областей застосування СППР було виявлено певні виклики та тенденції щодо майбутніх напрямків дослідження з двох точок зору: розвиток СППР у цілому та, зокрема, для підтримки прийняття рішень.

Незважаючи на те, що ручне вилучення знань було добре досліджено як засіб захоплення знань і моделювання структури знань, побудова середнього розміру в СППР все ще є трудомістким завданням. Одна з проблем полягає в отриманні предметної термінології та зв’язків із концептуальної моделі. Щоб відповісти на виклик, з’являється способи отримання форм даних автоматично або напівавтоматично. Ключові елементи вивчення включають вилучення інформації, відкриття та організацію. Є надія, що розвиток відповідних технологій, таких як кластерний аналіз може пролити світло на визначення зв’язків між термінами, застосовними до домену знання. Автоматизація, безумовно, знаходиться в початковому стані, і для цього в майбутньому потрібні додаткові дослідження підтримувати створення кращої СППР [19].

Майбутні дослідження повинні витрачати більше зусиль на перевірку знань, наприклад правил у знаннях база повинна бути підтверджена експертами. Потреба в перевірці знань стає ще більш критичною в клінічній СППР, оскільки один фрагмент неправильних або неточних знань може призвести до небезпечної або неправильної рекомендації, яка, у свою чергу, може завдати шкоди або викликати проблеми з безпекою пацієнтів. Третя проблема для технологій міркування полягає в тому, як включити невизначеність знань у СППР [20]. Нещодавні дослідження показали, що шляхом інтеграції існуючих міркувань на основі правил або аргументів на основі випадків з нечіткою логікою та штучними мережами можна підвищити продуктивність міркувань з точки зору невизначеності, яка повинна

залишатися гарячою темою для майбутніх досліджень. Нарешті, через внутрішню природу неповноти знань ні предметні знання, ні контекстуальні знання не є статичними або повними, оскільки самі знання постійно розвиваються, і ми ніколи не матимемо повного знання про проблему прийняття рішення чи рішення за один раз. Паралельно, технології міркування для отримання нових знань на основі існуючих знань, зафіксованих у базі знань, повинні вирішувати це питання еволюції.

Три основні компоненти структури DSS:

- модель системи управління;
- інтерфейс користувача;
- база знань.

Система управління моделями зберігає моделі, які менеджери можуть використовувати для прийняття рішень. Моделі використовуються для прийняття рішень щодо фінансового стану організації та прогнозування попиту на товар чи послугу.

Інтерфейс користувача містить інструменти, які допомагають кінцевому користувачеві DSS орієнтуватися в системі.

База знань включає інформацію з внутрішніх джерел (інформація, зібрана в системі процесу транзакцій) і зовнішніх джерел (газети та онлайн-бази даних).

Система підтримки прийняття рішень підвищує швидкість і ефективність прийняття рішень. Це можливо, оскільки DSS може збирати та аналізувати дані в реальному часі.

Це сприяє навчанню всередині організації, оскільки необхідно розвинути спеціальні навички для впровадження та використання СППР в організації.

Він автоматизує монотонні управлінські процеси, а значить, більше часу керівника може витрачатися на прийняття рішень.

Окрім того СППР можна використовувати для виявлення та прийняття рішень щодо зловмисницької діяльності всередині програмного продукту.

Зловмисна мережева діяльність може включати різну поведінку, пов’язану з незвичайними шаблонами доступу, змінами файлів і бази даних або будь-якою іншою підозрілою діяльністю, яка може свідчити про витік даних або атаку.

Своєчасне виявлення зловмисної активності допоможе визначити джерело злому та його природу, щоб якнайшвидше його усунути.

Якщо в організації є веб-додаток час від часу вона стикатиметься з інцидентами безпеки та нещасними випадками. У міру того як технологія прогресує, зловмисна діяльність в Інтернеті також зростає з кожним днем.

Зловмисники впроваджують нові способи зламу даних організації. Підтримувати кібербезпеку – це складна проблема, з якою стикається кожна організація, особливо після пандемії. Тепер головне завдання — знайти найкращі способи виявлення цих зловмисних дій і як їх уникнути.

Захист цифрових даних та інформації має вирішальне значення для виживання організації. Нездатність передбачити, а потім запобігти чи виявити порушення кібербезпеки (або кібербезпеки) може завдати непоправної шкоди. Аналітика та підтримка прийняття рішень можуть захистити цілісність мереж, програмного забезпечення та даних від атак, пошкодження чи несанкціонованого доступу. Підтримка прийняття рішень може допомогти оцінити профілактичні заходи безпеки та провести оцінку вразливості. Аналітика може відстежувати неавторизований доступ до мережі та виявляти підозрілі моделі мережевого трафіку. Інструменти підтримки прийняття рішень, особливо ті, що базуються на штучному інтелекті та знаннях, також можуть відстежувати фішингові шахрайства, які намагаються отримати конфіденційну інформацію, таку як імена користувачів, паролі та дані кредитної картки. Системи підтримки прийняття рішень також можуть допомогти у «візуалізації» загроз.

Сучасні комп’ютерні та комунікаційні інфраструктури дуже вразливі до різного роду атак.

У попередніх дослідженнях алгоритми класифікації успішно використовувалися для виявлення невідомого шкідливого коду. Більшість цих досліджень витягували ознаки на основі паттернів, щоб представити перевірені файли.

На рисунку 1.3 зображено загальну схему процесу прийняття рішень щодо виявлення атак в комп'ютерній системі.

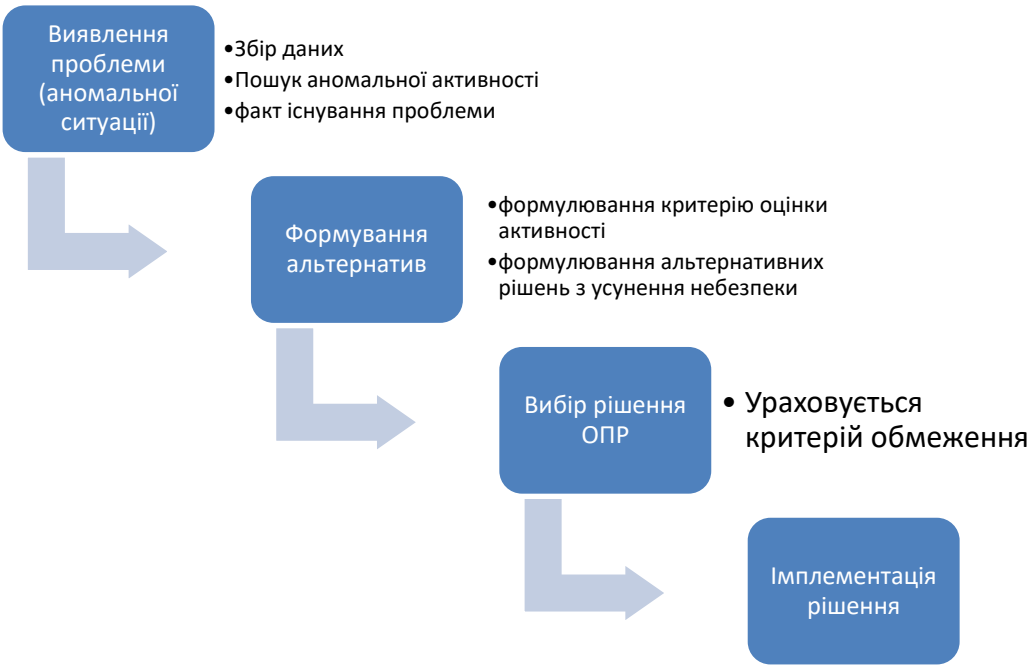


Рисунок 1.3 – Загальна схема процесу прийняття рішень

Першим кроком у процесі прийняття рішення є визначення характеру проблеми є важливим, оскільки прийняття рішень, зрештою, призначене для вирішення проблеми.

Виявлення аномалій – це етап інтелектуального аналізу даних, який визначає точки даних, події та/або спостереження, які відхиляються від нормальної поведінки набору даних. Аномальні дані можуть вказувати на критичні інциденти, такі як технічний збій, або потенційні можливості, наприклад, зміну поведінки споживачів.

Виявлення атак і аномалій в викликає все більше занепокоєння. Із збільшенням використання інфраструктури Інтернету речей у кожному домені

відповідно зростає кількість загроз і атак у цих інфраструктурах. Відмова в обслуговуванні, перевірка типу даних, зловмисне керування, зловмисна операція, сканування, шпигунство та неправильне налаштування – це такі атаки та аномалії, які можуть спричинити збій системи.

Щоб правильно ідентифікувати атаки, систему потрібно навчити розпізнавати звичайну системну активність. Дві фази більшості систем виявлення аномалій складаються з фази навчання (де будується профіль нормальної поведінки) і фази тестування (де поточний трафік порівнюється з профілем, створеним на фазі навчання). Аномалії виявляються кількома способами. Метод полягає в тому, щоб визначити, що включає нормальне використання системи, використовуючи математичну модель, і позначити будь-які відхилення від цього як атаку. Це відомо як виявлення аномалій.

Наприклад, бавовняна текстильна фірма може виявити, що її прибутки зменшуються. Потрібно дослідити причини проблеми зменшення прибутку. Чи то неправильна цінова політика, погані стосунки між працівниками та адміністрацією чи зловмисницька активність спричиняє проблему зниження прибутку. Коли джерело або причина падіння прибутку знайдено, проблема виявлена та визначена.

Після визначення проблеми наступним кроком є пошук альтернативних рішень проблеми. Це вимагатиме розгляду змінних, які впливають на проблему. Таким чином необхідно встановити зв’язок між змінними та проблемами.

У зв'язку з цим можуть бути розроблені різні гіпотези, які стануть альтернативними шляхами вирішення проблеми. Наприклад, у випадку проблеми, згаданої вище, якщо буде виявлено, що проблема зниження прибутку пов’язана з використанням у виробництві технологічно неефективного та застарілого обладнання.

Після виявлення атаки користувачі комп’ютерів повинні запроваджувати комбінацію заходів безпеки, використовуючи як технологічні, так і навчальні методи.

Як один із найстаріших методів боротьби з проблемами кібербезпеки, антивірусне програмне забезпечення має бути простим. Однак багато користувачів комп’ютерів просто не встановлюють його або нехтують оновленням наявного програмного забезпечення.

Наступним кроком у прийнятті бізнес-рішень є оцінка альтернативних варіантів дій. Це вимагає збору та аналізу відповідних даних. Деякі дані будуть доступні в різних відділах самої фірми, інші можуть бути отримані від промисловості та уряду.

Дані та інформація, отримані таким чином, можуть бути використані для оцінки результатів, очікуваних від кожного можливого курсу дій. Для визначення оптимального курсу використовуються такі методи, як регресійний аналіз, диференціальне числення, лінійне програмування, аналіз витрат і вигод. Оптимальним буде рішення, яке допомагає досягти поставленої фірмою мети. Насправді буде обраний оптимальний спосіб дії. Можна також відзначити, що для вибору оптимального рішення проблеми менеджер працює в умовах певних обмежень.

Обмеження можуть бути юридичними, наприклад закони щодо забруднення та утилізації шкідливих відходів; вони можуть бути фінансовими (тобто обмежені фінансові ресурси); вони можуть стосуватися наявності фізичної інфраструктури та сировини, і вони можуть бути технологічними за своєю природою, які встановлюють обмеження на можливу продукцію, вироблену за одиницю часу. Вирішальна роль бізнес-менеджера полягає у визначенні оптимального курсу дій, і він повинен прийняти рішення в умовах цих обмежень.

Після того, як альтернативні варіанти дій були оцінені та обрано оптимальний, останнім кроком є виконання рішення. Реалізація рішення потребує постійного моніторингу, щоб отримати очікувані результати від оптимального курсу дій. Таким чином, якщо буде виявлено, що очікуваних результатів не буде досягнуто через неправильне виконання рішення, то слід вжити коригувальні заходи.

Однак слід зазначити, що після впровадження курсу дій для досягнення встановленої мети час від часу можуть виникати потреби в його змінах у відповідь на зміни в умовах або операційному середовищі фірми, на основі яких ухвалювалися рішення.

Прийняття рішень можна визначити як вибір між альтернативними варіантами дій. Прийняття управлінських рішень стосується вибору, з яким стикаються керівники в рамках своїх обов’язків в організації. Прийняття рішень є важливим аспектом планування.

Основна задача прийняття рішень полягає у знаходженні оптимального методу досягнення цільового стану із поточного стану організаційної системи з урахуванням наявних ресурсів та впливу зовнішнього середовища.

Для знання-орієнтованих СППР Серед зображених задач ключову роль грає задача формування знань. доцільно розглянути саме метод формування знань.

Система підтримки прийняття рішень збирає дані та робить їх доступними. Вони вказують на те, що сталося, чому це сталося, а іноді й на те, що можна очікувати в майбутньому.

Існуючі підходи виявлення атак на систему складаються із двох основних частин:

- формування знань;
- задачі підтримки прийняття рішень.

Схематично зв’язок цих двох частин зображено на рисунку 1.5.

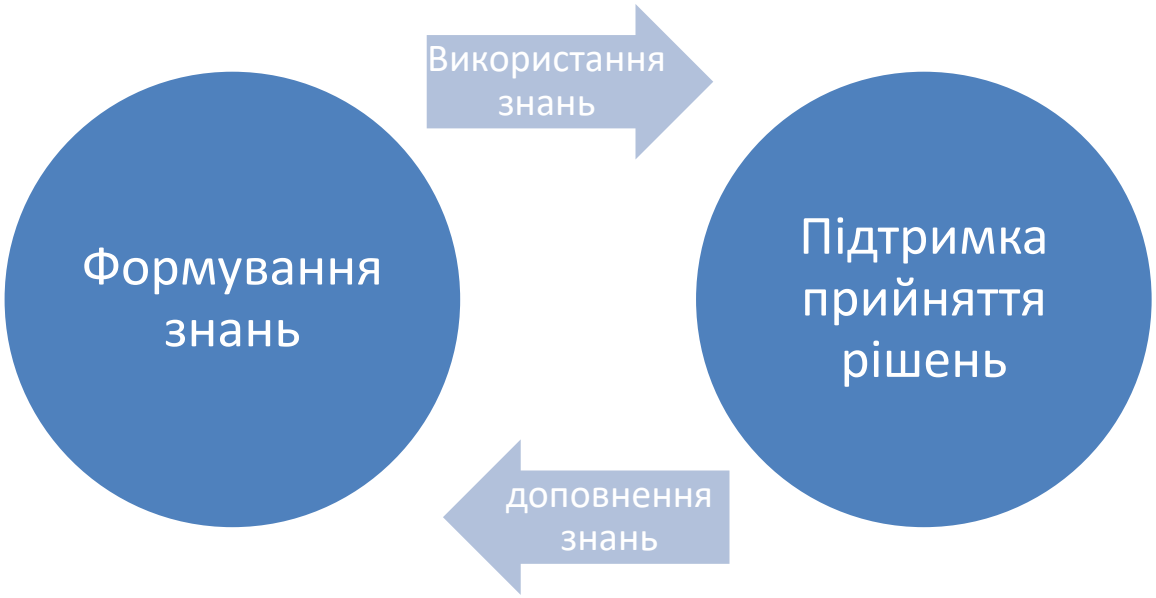


Рисунок 1.5 – Схематичне зображення існуючого підходу до виявлення атак

Якщо бізнес пов’язаний із великими обсягами даних, може бути складно отримати чіткий огляд і хорошу основу для прийняття рішень, особливо якщо бізнес-системи не пов’язані між собою.

Сховище даних потрібне для отримання даних і перетворення їх у корисну інформацію ефективним і якісним способом. Це форма бази даних, яка полегшує відбір, обробку та аналіз даних із кількох бізнес-систем. Хороша система підтримки прийняття рішень також включає сховище даних.

1.3 Дослідження методів формування знань

На практиці підтримка прийняття рішень поєднує стратегічне планування, бюджетування та прогнозування.

Управління знаннями є фундаментальним для галузі інформаційних технологій (IT). Бази знань, по суті, є інформаційними системами, і обмін знаннями є ключовим для забезпечення безперебійного перебігу бізнесу під

час впровадження нових процесів та інструментів, а також для забезпечення безпеки та дотримання нормативних вимог.

Управління знаннями та інформаційні технології пов’язані на базовому рівні – фактично, протягом більшої частини останніх 50 років IT були традиційним бізнес-власником управління знаннями через його складні технології та важкі вимоги до впровадження.

Однак із запровадженням моделі SaaS багато баз знань стали набагато простішими та легшими для встановлення та використання, що скорочує кількість часу, необхідного IT-відділам для їх адміністрування.

Чітка, надійна база знань (доповнена масштабованою стратегією управління знаннями) дозволяє всій компанії виконувати ті самі кроки під час впровадження нових процесів або технологій або під час усунення несправностей існуючих.

Замість того, щоб витрачати час на спілкування з кожною людиною, яка має запитання або стикається з проблемою (або, ймовірно, шукає невідповідне рішення), IT-спеціалісти можуть попередньо заповнити базу знань відповідями, які, як вони знають, матимуть працівники.

DSS може бути використаний управлінням операцій та іншими відділами планування в організації для збирання інформації та даних і синтезу їх у дієві дані. Насправді ці системи в основному використовуються керівництвом середнього та вищого рівня.

Системи підтримки прийняття рішень на основі знань (KBDSS) значно розвинулися за останні кілька десятиліть. Ключові технології, що лежать в основі розвитку KBDSS, можна класифікувати за трьома різними категоріями: технології для моделювання та представлення знань, технології для міркувань і логічних висновків і веб-технології.

Тим часом з’явилися системи послуг, які стають все більш важливими для доданої вартості в сучасній економіці знань.

Далі буде розглянуто саме методи формування знань. Їх можна поділити на два види, класифікація представлена у таблиці 1.1.

Таблиця 1.1 – Методи формування знань

Група методів	Метод	Характеристика методу
Автоматизвані	Методи вилучення знань із джерел структурованої або неструктурованої інформації (електронні таблиці, реляційні бази даних, XML документи, файли журналів, текст, зображення, тощо)	Формування джерел даних для автоматизованої побудови бази знань, а також формування залежностей між ними.
Мануальні	Комунікативні методи вилучення знань(анкетування, інтерв'ювання, вільний діалог)	Застосовуються для виявлення персональних знань співробітників, які можуть бути використані для аналізу проблемної ситуації на підприємстві
	Пасивні комунікативні методи	Використовуються для протоколювання поточного стану організаційної системи. Дана група методів об'єднує спостереження за реальним процесом функціонування підприємства або за його імітацією та аналіз вербальних звітів про рішення та дії виконавців

Кінець таблиці 1.1

Мануальні	Групові комунікативні методи	Використовуються для того, щоб усунути протиріччя в знаннях про частково структуровані або неструктуровані задачі шляхом взаємодії із групою експертів та передбачає обговорення проблемної ситуації за круглим столом та проведення «мозкового штурму» для виявлення принципово нових знань

Автоматизоване отримання знань полягає у вилученні знань із різних джерел неструктурованої чи структурованої інформації. В якості джерел структурованої інформації можна використати реляційні бази даних, електронні таблиці, документи в форматі JSON, XLM. У якості Неструктурованих джерел виступають текстові документи, зображення, тощо.

Обробка знань потребує розробки та підтримки їх фізичного представлення, індексації та обробки запитів і можу бути реалізована засобами СУБД [22]. Для автоматизації створення бази знань треба: сформувати множину джерел знань (цей процес у даній роботі пропонується покращити), формування залежності між знаннями, підтримка роботи бази знань.

Одним із прикладів ручного отримання знань є мозковий штурм ОПР. Сеанси мозкового штурму ОПР дозволяють учасникам офіційно висловити проблему та надати ідеї як рішення. Потім ідеї анонімно передаються (без оціночних коментарів) іншим учасникам. Потім вони надають власні вдосконалення та модифікації та сприяють потоку пов’язаних і значущих ідей, спрямованих на вирішення зазначеної проблеми. Після генерації ідей зазвичай відбувається оцінка конкретних ідей. Оцінки зазвичай включають стислий

перелік того, що учасникам подобається і не подобається в конкретній ідеї, разом із причиною, чому учасники так думають [23]. Потім група розглядає ці проблеми та працює над дійсним і загальновизнаним рішенням зазначеної проблеми, яке може бути реалізовано. Інформація, зібрана з ідей, формально зберігається у формі тексту або інших даних для подальшого використання. Зберігання інформації є каталізатором перетворення явних знань у нові.

При автоматизованому вилученні знань вилучення зв’язків відіграє важливу роль у вилученні структурованої інформації з неструктурованих джерел, таких як необроблений текст. Хтось може захотіти знайти взаємодію між ліками, щоб побудувати медичну базу даних, зрозуміти сцени на зображеннях або витягти стосунки між людьми, щоб створити базу знань, яку легко шукати.

Перш ніж почати виокремлювати зв’язки, доцільно визначити, які слова стосуються того самого «об’єкта» в реальному світі. Ці об’єкти називаються сутностями. Наприклад, «Барак», «Обама» або «президент» можуть стосуватися сутності «Барак Обама». Скажімо, ми виділяємо співвідношення щодо одного зі слів вище. Було б корисно об’єднати їх як інформацію про одну особу. Визначення того, які слова або згадки стосуються однієї сутності, є процесом, який називається зв’язуванням сутностей. Існують різні методи зв’язування сутностей, починаючи від простого зіставлення рядків і закінчуючи більш складними підходами машинного навчання. У деяких доменах ми маємо базу даних усіх відомих об’єктів, з якими можна зв’язатися, наприклад, словник усіх країн. В інших сферах ми повинні бути відкритими для відкриття нових сутностей.

1.4 Постановка задачі дослідження

Актуальність даної роботи полягає у тому, що існуючі методи підтримки прийняття рішень при виявленні атак на комп’ютерну систему базуються на використанні множини патернів таких атак, що ускладнює виявлення нових нетипових втручань у діяльність комп'ютерної системи. Для виявлення таких атак можуть бути використані знання, що відображають процеси у комп’ютерній системі у нормальному режимі функціонування й у випадку атак та можуть бути побудовані автоматизованим способом на основі аналізу логів цієї системи.

Об'єктом дослідження є процес формування знань на основі аналізу логів в СППР для виявлення атак в комп’ютерній системі.

Предметом дослідження – методи формування знань в системах підтримки прийняття рішень.

Метою роботи є дослідження методів формування знань для підтримки прийняття рішень при виявлення атак в комп’ютерній системі.

Науковою новизною є вдосконалено метод формування знань для підтримки прийняття рішень, шляхом врахування ваги правил для виявлення нетипової ситуації об'єкту управління безпосередньо у процесі побудови знань.

У ході виконання роботи було розроблено модуль формування знань на основі аналізу логів комп'ютерної системи.

У ході виконання роботи було розглянуто задачі дослідження процесу підтримки прийняття рішень, аналіз задач підтримки прийняття рішень при виявленні атак в комп’ютерних системах, дослідження методів формування знань, удосконалення методу формування знань для підтримки прийняття рішень з виявлення атак в комп’ютерній системі, експериментальна перевірка удосконаленого методу.

2 УДОСКОНАЛЕННЯ МЕТОДУ ФОРМУВАННЯ ЗНАНЬ ДЛЯ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ З ВИЯВЛЕННЯ АТАК В КОМП'ЮТЕРНІЙ СИСТЕМІ

2.1 Загальний підхід до формування знань на основі аналізу логів в комп'ютерній системі

Формування знань – це процес використання різноманітних джерел інформації для створення цілісного банку знань. У рамках цього підходу вилучення часто спиратиметься на низку як структурованих, так і неструктурованих джерел. У разі успіху вилучення знань призводить до надійних даних, які можна легко прочитати та інтерпретувати певною програмою, дозволяючи кінцевому користувачеві використовувати ці формальні знання для будь-яких цілей, які він або вона бажає [24]. У філософії сучасна концепція формування знання об'єднує емпіричну та раціоналістичну концепції. Знання створюються емпірично, досвідченим шляхом, у тому сенсі, що формування знань передбачає перебудову системи організм-середовище та включення в систему нових елементів. Знання, однак, базується не на функціонуванні органів чуття, а на структурі системи організм-середовище та на її модифікаціях у процесі диференціації та розширення системи з новими результатами поведінки.

Знання є основою всього існуючого. Без знання ніщо не існувало б таким, яким ми його сприймаємо. Це обов'язково і незамінно. Знання є будівельними блоками будь-якої основи. База знань покращує здатність компанії створювати, організовувати, керувати своїми знаннями, ділитися ними та експоненціально ефективно використовувати ці знання [25].

Кілька різних джерел можуть бути використані в процесі автоматизованого вилучення знань.

У рамках структурованих джерел дані можуть бути витягнуті з різних типів реляційних баз даних або певного типу розширюваної мови розмітки чи джерела XML.

Неструктуровані джерела, такі як зображення, різні форми текстових документів, електронні таблиці та навіть текст, записаний у програмах у стилі блокнота, можуть бути використані як частина процесу вилучення. Поки джерела доступні для читання програмою, яка використовується для керування процесом вилучення знань, їх можна використовувати як джерела, які розширюють потенціал для проекту, який просувається за допомогою вилучення, і дозволяють використовувати отримані кінцеві знання.

Розроблені у даному розділі методи базуються на роботах [25] – [30].

Така схема показує варіативність об’єктів управління в межах часових знань, а також додавання цих знань у процесі функціонування об’єкта управління.

У той самий час під час використання знань для підтримки прийняття рішень необхідно постійно актуалізувати знання про стан об'єкта управління та зв'язки. Організаційна креативність – відносно нове поняття в теорії менеджменту, яке частково виникло на ґрунті управління знаннями. Під системою з часовою базою знань (або темпоральною) наводиться система, яка явно виконує часові міркування. Така система містить не лише фактологічну базу, базу правил і механізм логічного висновку, але й безпосередньо стосується питання часу. Щоб інтелектуальна система була тимчасовою, вона повинна містити явні представлення часу у своїй базі знань – формалізовані за допомогою часових логік – і принаймні на рівнях представлення та міркування.

Сучасні засоби Інтернету накладають певні умови на архітектуру та використання експертних систем. Багато компонентів ES, такі як база знань (БЗ), механізм логічного висновку, підсистема пояснення, змінюють свої властивості та функції під впливом Інтернету. Все більшу роль тепер відіграють не статичні знання, а динамічні, не поверхневі, а глибокі знання.

Підсистема пояснення забезпечена методами, заснованими на аргументації отриманих результатів з використанням нерелевантної інформації. Механізм логічного висновку все більше базується на принципах, заснованих на міркуванні за допомогою асоціацій і аналогії. І такі системи мають працювати в режимі реального часу та на мобільних пристроях. Інформаційні домени мають динамічну структуру, наприклад Інтернет, прогнозування аварій та надзвичайних ситуацій, розподілене навчання тощо. Їх особливостями є: наявність величезної кількості автономних утворень зі своїми специфічними підцілями (автономією). Суб'єкти піддаються впливам зовнішнього середовища (відкритість), взаємодіють між собою (розподіл). Бази знань сутностей унікальні (локальність) і утворюють ієрархічні коаліції (ієрархія рівня сутності). Для побудови моделей баз знань таких предметних областей використовуються, наприклад, як моделі штучних нейронних мереж, так і самоорганізовані відкриті багатоагентні системи. Проблема виявлення темпоральних знань є суттєвою у вирішенні багатьох проблем у сфері штучного інтелекту.

Існує кілька шляхів її вирішення, наприклад, традиційний напрямок полягає в явному використанні часу в темпоральних моделях знань. Інший підхід полягає в тому, щоб неявно використовувати час на ідеях шарування бази знань.

У теорії складних динамічних систем однією з проблем є прийняття рішень з багатьма цілями. Динамічна система характеризується тим, що її компоненти і параметри явно або неявно залежать від часу. Еволюція динамічної системи задається або диференціальними рівняннями, або графіком її станів, або іншими законами. У випадку, коли динамічна система задається графом станів, вона має такі важливі властивості, як зв’язність, складність, стійкість, цілісність, ієрархічність та цілі поведінки, які погано формалізовані.

При вирішенні задач підтримки рішень база темпоральних знань має виявляти незвичні стани та прояви об’єкта управління в умовах невизначеності, і формувати альтернативи дій.

Представлення знань поєднує логічний опис правил та функцію розрахунку ймовірності можливих альтернативних рішень Vnewi.

Метод містить такі етапи:

Етап 1. Побудова переліку подій, тобто виділення із файлу журналу в послідовності виконання, включаючи поточний стан $p_{i,j}$.

$$S_i = \langle p_{i,1}, \ldots, p_{i,j-1}, \grave{p_{i,j}} \rangle, \tag{2.1}$$

Результатом цього кроку є набір усіх можливих послідовностей, які призводять до стану, представленого фактом p'. Кожен S'i відповідає одній послідовності подій πi.

Етап 2. Відбір вхідних даних. Особливість даного етапу полягає у пошуку атрибутів, обираються події із ключовим атрибутом. Події без нього відкидаються.

Етап 3. Побудова пар фактів Ej у класи еквівалентності. Факти будуються на основі пар подій шляхом поєднання один з одним

Етап 4. Побудова правил за принципом кожна подія із ключовим атрибутом пов’язана із наступною.

Етап 5. Знаходження ваг темпоральних залежностей з урахуванням ймовірностей Vi.

Представлення тимчасової інформації та міркування щодо такої тимчасової інформації важливі для багатьох (інтелектуальних) комп’ютерних систем. Це призвело до розробки широкого розмаїття часових формалізмів із лаконічним синтаксисом, правильною семантикою та правилами

обчислювального висновку. Подано огляд таких часових міркувань і формалізмів моделювання даних. Будь-який часовий формалізм зазвичай розробляється з урахуванням його передбачуваного застосування – будь то планування, прогнозування, навчання чи планування, оскільки конкретна часова інформація будь-якої програми суттєво відрізняється від інформації в іншій програмі.

Представлення тимчасової інформації та логічне обґрунтування такої інформації має велике значення для багатьох інтелектуальних комп'ютерних систем. Для таких систем було розроблено цілий ряд формалізмів опису, заснованих на логіці, для представлення часу (коротко: формалізми часу). Ця робота пропонує вичерпний огляд формалізмів часу. Зрозуміло, що не може бути універсально застосовних формалізмів часу. Отже, напружені формалізми особливо залежать від їх відповідного передбачуваного застосування.

2.2 Удосконалений метод формування знань з використанням логів комп'ютерної системи

При пошуку знань найчастіше використовується існуюча документація фірми чи підприємства.

Однак, у даній роботі як вхідні дані пропонується використати файли журналів (або log-файли). Пропонується використовувати саме файли журналів, бо зазвичай їм не приділяється увага саме у контексті пошуку нових знань.

Файл журналу, також відомий як файл журналу або файл журналу, - це файл, у якому комп'ютерні процеси реєструють різні події. Файли журналів є важливими джерелами інформації, які дозволяють відстежувати процеси в

системі. Їх можна використовувати, наприклад, для аналізу проблем або відновлення втрачених даних.

Файли журналів є основним джерелом даних для спостереження за мережею. Файл журналу – це створений комп'ютером файл даних, який містить інформацію про моделі використання, дії та операції в операційній системі, програмі, сервері чи іншому пристрої. Файли журналів показують, чи належним чином і оптимально працюють ресурси.

Файли журналу автоматично створюються комп'ютером щоразу, коли в мережі відбувається подія з певною класифікацією. Причина існування файлів журналів полягає в тому, що розробникам програмного та апаратного забезпечення легше виявляти неполадки та налагоджувати свої творіння, коли вони отримують доступ до текстового запису подій, які створює система. Кожна з провідних операційних систем унікально налаштована для створення та класифікації журналів подій у відповідь на певні типи подій. Системи керування журналами централізують усі файли журналів, щоб сортувати й аналізувати дані журналів, а також полегшують розуміння, відстеження та вирішення ключових проблем, пов'язаних із продуктивністю додатків.

Великі IT-організації залежать від розгалуженої мережі IT-інфраструктури та додатків для роботи ключових бізнес-сервісів. Моніторинг та аналіз файлів журналу підвищують спостережуваність цієї мережі, створюючи прозорість і забезпечуючи видимість у середовищі хмарних обчислень. Хоча спостережливість не слід розглядати як кінцеву мету, її завжди слід розглядати як механізм досягнення реальних бізнес-цілей.

Підвищення надійності систем для кінцевого користувача. Файли журналу містять інформацію про продуктивність системи, яку можна використовувати для визначення того, коли потрібна додаткова ємність для оптимізації роботи користувача. Файли журналів можуть допомогти аналітикам визначити повільні запити, помилки, через які транзакції тривають занадто довго, або помилки, які впливають на продуктивність веб-сайту чи програми.

Підтримка рівня безпеки середовищ хмарних обчислень і запобігання витоку даних.

Файли журналу фіксують такі речі, як невдалі спроби входу, невдала автентифікація користувача або несподіване перевантаження сервера, усе це може сигналізувати аналітику про можливу кібератаку. Найкращі інструменти моніторингу безпеки можуть надсилати сповіщення та автоматизувати відповіді, щойно ці події виявляються в мережі.

Удосконалення процесу прийняття бізнес-рішень. Файли журналу фіксують поведінку користувачів у програмі, створюючи область запитів, відому як аналітика поведінки користувачів. Аналізуючи дії користувачів у програмі, розробники можуть оптимізувати програму, щоб користувачі швидше досягали своїх цілей, підвищуючи задоволеність клієнтів і збільшуючи дохід у процесі.

Лог-файли містять у собі події (або факти), що виконувались один за одним.

Ідеє удосконаленого методу полягає у поєднанні формування знань і підтримки прийняття рішень. Тобто задачі формування знань будуть вирішуватись лише у тому випадку, коли у процесі формування знань виявлено нетипову ситуацію.

Тобто було виділено послідовність кроків.

Крок 1. Формування знань на основі логів

Крок 1.1. Виявлення нетипової (аномальної ситуації), якщо на цьому кроці такої ситуації не було виявлено, алгоритм закінчується.

Крок 2. Формування альтернатив рішення.

Крок 3. Вибір рішення ОПР.

Крок 4. Імплементація рішення.

Для виконання кроку 1 треба сформувати знання та правила, для яких встановити ваги, завдяки цьому можна визначити аномальну ситуацію.

Між цими подіями можна сформувати правила. Правила формуються не хаотично, а послідовно – кожна подія із кожною. Наприклад, у системі

виникла послідовно перша подія (Е1), потім друга (Е2)Б а потім третя (Е4). Їх можна об’єднати правилами таким чином:

- після Е1 може виникнути подія Е2;
- після Е1 може виникнути подія Е3;
- після Е2 може виникнути подія Е3.

Схематично правила між подіями можна зобразити на рисунку 2.1.

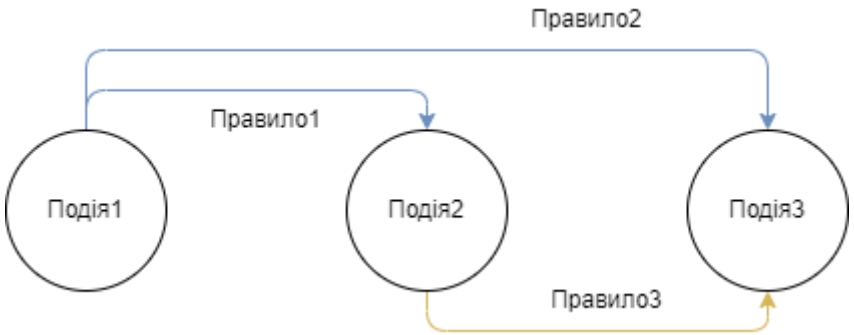


Рисунок 2.1 – Схема взаємозв’язку правил та подій

Тобто у запропонованому методі у підрозділі 2.1 змінюється вибір патернів на схему правил.

Кожна подія, що відображає знання про появу чи зміну стану об’єкту управління може бути визначена предикатом на підмножині атрибутів артефактів. Атрибути можуть являти як управляючу дію так і контекст її виконання. Кожна дія має характеристики, які визначають задачу, стан тощо.

Окрім того для побудови правил варто визначити, чи є у певної події атрибути. Якщо подія у лозі не має визначного атрибута доцільно не враховувати її при побудові правил.

Метод можна покращити саме для задачі формування знань для процесу підтримки прийняття рішень. Знання, які потребуються у процесі, не обов’язково мають міститись у всіх подіях, що знаходяться у лог-файлах. Це надлишкова інформація і від неї треба позбавитись для більш чіткого представлення про об’єкт управління і події всередині нього. Для цього доцільно звернути увагу на атрибути кожної події. Їх може бути багато, але

для швидкого та чіткого вирішення поставленої задачі доцільно виділити ключовий атрибут.

Атрибут, на який можна спиратися, це, наприклад, назва. Якщо запис у лозі не має ніякої назви (це можуть бути “Error..”, “Info” або інше), то таку подію не доцільно розглядати для утворення правил.

У наведеному вище прикладі візьмемо подію E1, як подію, що не має атрибутів. Тоді правила будуть побудовані так:

після E2 може виникнути подія E3.

Удосконалений метод формування знань дає можливість визначити упорядкованість для пар послідовних станів об'єкту управління, що відображені у базі знань парами послідовних у часі фактів. Метод в якості вхідних даних використовує множину послідовностей подій V = {Vi}, яку отримано з файлів журналів.

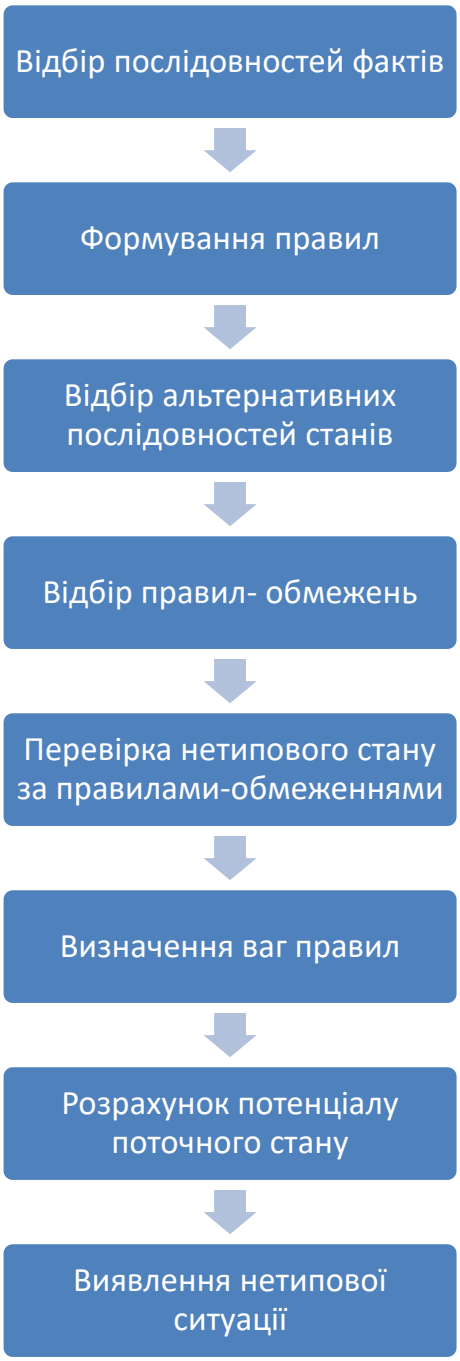


Рисунок 2.2 – Узагальнена послідовність етапів побудови знань

Отже метод містить такі етапи:

Етап 1. Відбір послідовностей фактів, що відображають послідовність станів комп'ютерної системи.

Етап 2. Формування правил, що визначають зв'язки між фактами

Етап 3. Відбір альтернативних послідовностей станів, що передують поточному стану системи.

Етап 4. Відбір правил- обмежень, що виконуються для всіх відібраних послідовностей станів

Етап 5. Перевірка нетипового стану за правилами-обмеженнями. Якщо стан – нетиповий – то завершення роботи.

Етап 6. Визначення ваг правил r_m^j, які не є обмеженнями, тобто задовольняють умові:

$$\left(\forall r_m^j\right) w_m^j \neq \infty, \tag{2.2}$$

Етап 7. Розрахунок потенціалу поточного стану для альтернативних станів системи. Потенціал розраховується як сума ваг правил в послідовності станів, що передує поточній.

$$\Phi_{i,j} = exp(\sum_m (w_m^j : (\forall j, m) w_m^j \neq \infty)), \tag{2.3}$$

де w_m^j вага правила r_m^j.

Етап 8. Виявлення нетипової ситуації (ситуації атаки) шляхом порівняння потенціалів поточних станів.

$$\left|\Phi_{i,j} - \Phi_{i,J}\right| > \varepsilon, \tag{2.4}$$

де ε – заздалегідь визначений поріг.

Відповідно до формули 2.2, якщо різниця потенціалів двох фактів перевищує заздалегідь визначений поріг ε, цей факт є ненормальним. В іншому випадку стан поточний належить до того ж класу, що і стандартний

стан $p_{1,M}$. Результатом цього методу є ненормальний або стандартний клас поточного стану, представлений як факт.

Великі програмні системи складаються з різних компонентів часу виконання, партнерських програм і процесів. Коли такі системи функціонують, вони контролюються, щоб можна було проводити аудит, коли виникає збій або коли виконуються операції з технічного обслуговування. Однак файли журналу зазвичай мають великий розмір і для ефективної обробки вимагають фільтрації та скорочення. Крім того, немає очевидної відповідності того, як зареєстровані події стосуються конкретних випадків використання, які може виконувати система. У цій роботі розглянуто структуру, яка базується на евристичних алгоритмах кластеризації для досягнення фільтрації журналу, зменшення журналу та інтерпретації журналу.

Щоб отримати хороші результати інтелектуального аналізу процесів, журнал подій потребує високої якості. Існують різні параметри якості даних у журналах подій.

Файли журналу містять інформацію майже про всі події, що відбуваються в системі, залежно від рівня журналу. Для цього розгорнута інфраструктура журналювання автоматично збирає, об’єднує та зберігає журнали, які постійно створюють більшість компонентів і пристроїв, наприклад, веб-сервери, бази даних або брандмауери. Текстові повідомлення журналу, як правило, читаються людиною та додаються до позначки часу, яка вказує момент часу створення запису журналу. Особливо для великих організацій і підприємств переваги доступу до довгострокових журнальних даних є різноманітними: історичні журнали дозволяють проводити аналіз минулих подій. Аналіз, який найчастіше застосовується після виникнення збоїв у системі, дає системним адміністраторам можливість відстежити коріння виявлених проблем. Крім того, журнали можуть допомогти відновити систему до безпомилкового стану, скинути неправильні транзакції, відновити дані, запобігти втратам інформації та відтворити сценарії, які призводять до помилкових станів під час тестування.

Великі програмні системи складаються з ряду різних компонентів часу виконання, партнерських програм і процесів. У багатьох ситуаціях потрібно перевірити та проаналізувати файли журналів, створені цими різними компонентами середовища виконання, партнерськими програмами та процесами, щоб можна було виконати аналіз першопричини, діагностику або просто отримати уявлення про можливі випадки використання запущено в будь-яку задану точку з метою обслуговування, планування або еволюції. Однак аналіз подій у файлах журналу є обчислювально дорогим і складним процесом, особливо коли задіяно багато різних компонентів і програмних моніторів. Методи, які використовуються для аналізу файлів журналу, створених різними джерелами та в різних форматах, поділяються на дві основні категорії. Перша категорія базується на статистичному аналізі, метою якого є кореляція подій за допомогою інтелектуального аналізу даних, вдосконалених методів кореляції подій і складних методів обробки подій. Мотивація, яка лежить в основі цих підходів, полягає в тому, щоб оператор міг ідентифікувати події, які виявляють високий ступінь спільності, а також можуть бути пов’язані з високим ступенем ймовірності з певною помилкою або причиною збою системи. У цій категорії підходів система моніторингу повинна мати доступ до великої кількості минулих випадків, щоб спочатку можна було встановити статистично значущі кореляції.

Існує три основні способи вилучення знань: ручне сортування та вилучення; вилучення з протоколювання операторів у вихідному коді і вилучення з необроблених кластерів. На практиці журнали програмного забезпечення мають складні приховані структури та є великими. Таким чином, ручне створення шаблонів є трудомістким і схильним до помилок. Крім того, вихідний код певних компонентів системи часто недоступний (наприклад, сторонні бібліотеки). Таким чином, доцільно використати метод вилучення шаблону з журналів програмного забезпечення за допомогою ітераційної кластеризації.

Вхід методу – це файл журналу, що складається з необроблених повідомлень журналу, а вихід – витягнуті шаблони та структуровані журнали. Зокрема, під час ітерації спочатку вибираємо частину вхідних журналів. Потім до зразків журналів застосовується метод ієрархічного поділу для створення кількох кластерів, з яких шаблони можна витягувати автоматично.

На етапі зіставлення алгоритм зіставляє всі необроблені журнали з цими шаблонами, зібрати невідповідні журнали та передати їх у наступну ітерацію як вхідні дані. Повторюючи ці кроки, можна точно й ефективно зіставити всі повідомлення журналу за допомогою належного призначення шаблону. Причина цього полягає в тому, що вибіркові журнали часто можуть покривати шаблони, приховані в більшості вхідних журналів під час кожної ітерації. Зокрема, частина операторів журналювання може виконуватися набагато частіше, ніж інші. Таким чином, шаблони, згенеровані з невеликої частини журналів, зазвичай можуть відповідати більшості необроблених журналів під час перших кількох ітерацій.

Запропоноване рішення забезпечує підхід до вилучення знань із журналу подій для інтелектуального аналізу процесів. Він реалізований як з’єднувач між інструментом інтелектуального аналізу процесів і системою підтримки прийняття рішень. Це включає визначення необхідних даних і правильну ідентифікацію подій і випадків.

3 ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ФОРМУВАННЯ ЗНАНЬ ДЛЯ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ З ВИЯВЛЕННЯ АТАК НА КОМП'ЮТЕРНУ СИСТЕМУ

Інформаційна технологія – це широкий термін, який передбачає використання технологій для зв'язку, передачі даних і обробки інформації.

Різні тенденції в інформаційних технологіях включають, але не обмежуються:

- аналітика;
- автоматизація;
- штучний інтелект;
- хмарні обчислення;
- комунікації;
- кібербезпека;
- управління даними/базами даних;
- інфраструктура;
- машинне навчання;
- мережі;
- розробка програмного забезпечення/додатків.

Інформаційні технології відіграють важливу роль у бізнесі та забезпечують основу для значної частини нашої нинішньої робочої сили. Від комунікацій до керування даними та ефективності роботи, IT підтримують багато бізнес-функцій і допомагають підвищити продуктивність.

IT також включає в себе управління даними, будь то у формі тексту, голосу, зображення, аудіо чи в іншій формі. Це також може включати речі, пов'язані з Інтернетом. Це надає IT абсолютно нового значення, оскільки Інтернет є окремою сферою діяльності. IT передбачає передачу даних, тому

має сенс, щоб Інтернет був частиною IT. IT стали частиною нашого повсякденного життя та продовжують поширюватися в нові сфери.

Основна технологія у роботі спирається на аналіз знань, що отримуються із файлів журналу та потім використовуються системами підтримки прийняття рішень. Log-файли містять інформацію про час виконання, записаний розробниками, яка широко аналізується для різноманітних завдань. Аналіз журналу проводиться для різних цілей, таких як тестування коду, виявлення проблем, аналіз поведінки користувачів, моніторинг безпеки тощо. Більшість із цих завдань використовують моделі інтелектуального аналізу даних для вилучення критичних функцій або шаблонів із великої кількості журналів програмного забезпечення. Тому вважаємо, що робота над стисненням журналів може принести користь зберіганню даних журналів і заощадити витрати на створення дампу великого обсягу журналів. Розбір журналу. зазвичай використовується як перший крок подальших завдань.

Запропонована технологія (на рисунку 4.1) інтегрує моделі представлення знань, методи формування знаньта використовується у процесі прийняття управлінських рішень.

Процес формування знань базується на темпоральних подіях, записаних у лог файлах, які за допомогою технології представлення у вигляді послідовності.

При вирішенні задачі необхідно виділити окремо кожну подію з яких згодом будуть отримані знання.

Технологія в якості вхідних даних використовує:

– файли логів щодо подій у системі, представлених фактами щодо виникнення послідовностей станів;

– знання зі стану об'єкту дослідження.

Рисунок 4.1 – Інформаційна технологія автоматизованого формування знань для виявлення атак

Технологія містить у собі такі етапи.

Етап 1. Створення моделі виконання дій, тобто виділення із файлу журналу послідовності виконання подій.

Етап 3. Відбір вхідних даних та пошук атрибутів.

Етап 4. Створення класів еквівалентності подій.

Етап 5. Побудова темпоральних правил за принципом кожна подія із ключовим атрибутом пов’язана із наступною.

Етап 6. Знаходження ваг залежностей.

Етап 7. Порівняння ваг та виявлення нетипової ситуації.

Таким чином, розроблена технологія дозволяє сформувати знання, необхідні для прийняття управлінських рішень у системах підтримки прийняття рішень з урахуванням послідовності подій, записаних у файлах журналів.

Таким чином можна провести автоматизацію процесу формування знань.

Перелік інструментальних засобів за допомогою яких можна програмно реалізувати магістерську роботу:

– python3;

– Windows, Linux или macOS.

Формат даних, що використовується:

– кілька файлів матриці послідовності журналу: кожен файл складається з векторів послідовності журналу в межах інтервалу часу.

Однією з основних ролей інформаційних технологій у програмах управління знаннями є прискорення швидкості передачі та створення знань. Інструменти управління знаннями мають на меті допомогти процесам збору та організації знань груп осіб, щоб зробити ці знання доступними в спільній базі. Через масштабність концепції знань ринок програмного забезпечення для управління знаннями виглядає досить заплутаним. Постачальники технологій розробляють різні реалізації концепцій управління знаннями у своїх програмних продуктах. Через різноманітність і кількість інструментів управління знаннями, доступних на ринку, типологія може бути цінною допомогою для організацій, які шукають відповіді на конкретні потреби. Метою цієї статті є представлення вказівок, які допоможуть розробити таку типологію. Рішення для управління знаннями, такі як інтранет-системи, електронний документообіг, групове програмне забезпечення, робочий процес, системи на основі штучного інтелекту, бізнес-аналітика, системи карт знань, підтримка інновацій, інструменти конкурентної розвідки та портали знань, обговорюються з точки зору їх потенційний внесок у процеси створення, реєстрації та обміну знаннями.

Управління знаннями має намір стати сферою досліджень і практики, яка поглиблює розуміння процесів отримання знань в організаціях і розробляє процедури та інструменти для підтримки перетворення знань в економічний і соціальний прогрес. Фактично, різні аспекти цих питань вивчалися протягом десятиліть у багатьох різних дисциплінах, через багато різних фільтрів, як-от управління дослідженнями та розробками та інноваціями, управління

інформаційними системами, інформатика, інформатика, бібліотекознавство, інноваційна економіка, наука та соціальна техніка епістемологія та багато інших. Можливо, одним із найважливіших внесків концепції управління знаннями є створення простору (в академії, у діловому світі та в кіберпросторі), де ці численні групи та точки зору можуть обговорювати та працювати разом.

Об’єкти дослідження охоплюють людей, організації, процеси та технології. Хоча технологія не є основним компонентом КМ, було б наївним ставленням до впровадження КМ без урахування будь-якої технологічної підтримки. Технології автоматизації знань (КА) поєднують машинне навчання з контентом і аналітикою, формуючи здатність виявляти прогалини в знаннях і потім виправляти їх за допомогою підібраного контенту або інформаційної кампанії.

Поєднання машинного навчання з постійним вимірюванням того, що люди роблять у контексті конкретного завдання чи роботи, дозволяє системам автоматизації знань постійно надавати потрібну інформацію в потрібний час. Зазначимо, що існує зв’язок між додатками КА та розвитком цифрових помічників.

Знання, як інформація, мають сенс лише у зв’язку з когнітивною здатністю. Знання –це об’єднання правил, принципів, ментальних моделей, спогадів, у яких закладена людська дія. Отримані повідомлення (інформація) можуть доповнювати наявні знання в когнітивній системі. А можуть і ні, якщо вони не є джерелом спогадів, правил, моделей, які впливають на дії. Зразок даних, надісланий когнітивною системою з наміром надіслати повідомлення, може негайно генерувати нові знання в приймачі, лише тимчасову інформацію в іншому та шум (відсутність будь-якого сенсу) у третьому.

Інформація та знання відрізняються щільністю та глибиною.

Явні знання – це формальні знання, які можна запакувати як інформацію та знайти в документах організації: звіти, статті, посібники, патенти, малюнки, зображення, відео, звук, програмне забезпечення тощо. Неявні знання – це

особисті знання, вбудовані в індивідуальний досвід і передається та обмінюється через прямий контакт. Очевидно, що неявні знання можна передати найбільш прямим і ефективним способом. Навпаки, отримання явних знань є опосередкованим: вони повинні бути декодовані та перекодовані у своїх ментальних моделях, де потім інтерналізуються як неявні знання.

Насправді ці два типи знань – однаково важливі для загального знання організації. Неявні знання – це практичні знання, які є ключовими для виконання завдань, але ними, на жаль, знехтували в минулому, часто стаючи жертвами останньої моди в менеджменті. Наприклад, нещодавня хвиля ініціатив із реорганізації бізнес-процесів, де зниження витрат зазвичай ототожнювалося зі звільненням людей, які є справжніми й єдиними сховищами неявних знань, завдала шкоди неявним знанням багатьох організацій. Явні знання визначають ідентичність, компетенцію та інтелектуальні активи організації незалежно від її працівників; таким чином, це організаційні знання, але вони можуть рости та підтримувати себе лише завдяки багатому фону неявних знань.

З одного боку, інформаційні технології широко використовуються в організаціях і, таким чином, кваліфікуються як природне середовище для потоку знань. Нещодавнє дослідження, проведене Американським центром продуктивності та якості, показує, що організації, які починають роботу з управління знаннями, для досягнення своїх цілей зазвичай покладаються на створення відповідної IT-інфраструктури. На іншому кінці спектру провідні теоретики управління знаннями попереджають про ставлення, яке спонукає керівництво до сильної інвестиції в IT, можливо, за рахунок інвестицій у людський капітал.

Частково проблема тут походить від лінгвістичної неоднозначності: сьогодні інформаційні технології створюють прямі зв’язки між людьми за допомогою таких додатків, як електронна пошта, чати, відеоконференції та інші типи групового програмного забезпечення, так само як і зберігання інформації в базах даних. та інші типи сховищ. Що стосується інформаційних

баз даних, їх також можна переосмислити в перспективі управління знаннями як ресурси для обміну передовим досвідом і для збереження інтелектуального капіталу організацій. Загалом, інвестиції в ІТ здаються неминучими для розширення проектів управління знаннями. Найкращим способом застосування інформаційних технологій для управління знаннями є, ймовірно, поєднання двох факторів: з одного боку, усвідомлення обмежень інформаційних технологій і того факту, що будь-яке розгортання ІТ не досягне багато чого, якщо воно не супроводжується через глобальну культурну зміну в бік цінностей знання; з іншого боку, доступність інформаційних технологій, які були спеціально розроблені з метою управління знаннями.

4 ЕКСПЕРИМЕНТАЛЬНА ПЕРЕВІРКА УДОСКОНЛАЛЕНОГО МЕТОДУ ФОРМУВАННЯ ЗНАНЬ

4.1 Реалізація програмного модулю з формування знань

Модуль вирішує задачу формування знань на основі журналів. Він використовує як системні журнали, так і системні показники для швидкого й точного виявлення серйозних системних проблем.

Модуль реалізує завантаження даних.

Для розробки прототипу використано:

– мова програмування Python;

– платформа Windows, Linux або macOS.

Формат віхдних даних:

– кілька файлів журналу: кожен файл складається з векторів послідовності журналу в межах інтервалу часу.

Приклад такого файлу наведено у таблиці 4.1.

Таблиця 4.1 – Приклад вхідного файлу журналу

Подія	Атрибут1	Атрибут2
1	2	1
2	3	2
…	…	…

Припустимо, що всього N часових інтервалів, тоді N таких матриць. Ці значення зберігаються в одному файлі.

Для відкриття log-файлів можна використати вбудований Windows додаток «Блокнот» або використати Notepade. Перевага Notepade у зручній зміні кодування та більш широкому інтерфейсі.

На мові python завантаження вихідного логу виглядає так (рисунок 4.1).

```
import pandas as pd
initiallog = pd.read_csv(C:/Users/eventlog.csv)
print(initiallog)
```

Рисунок 4.1 – Лістинг фрагменту програми завантаження файлу

На рисунку 4.2 зображено екранну форму прототипу модуля, на якому реалізується завантаження вхідного файлу журналу.

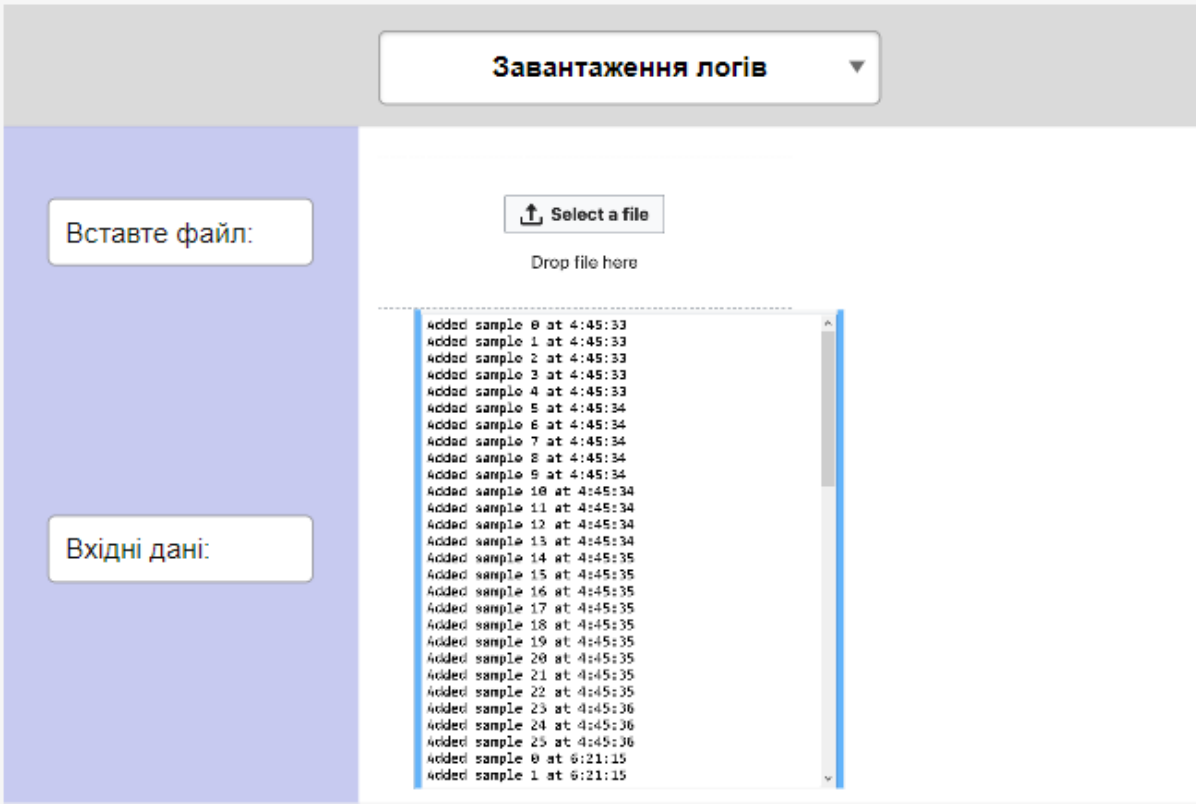


Рисунок 4.2 – Екранна форма прототипу програми, що реалізує процес завантаження файлу журналу

Після завантаження файлу програма повинна визначити події та впорядкувати їх одна за одною, у порядку їх виконання у системі.

Наступний крок – визначення ключових подій, які і будуть використані для формування знань. Вони визначаються на основі атрибута, визначеного як основний.

На мові python пошук ключового атрибуту виглядає так (рисунок 4.3).

```
with open(initiallog) as f:
    f = f.readlines()

   for line in f:
        for phrase in keep_phrases:
            if phrase in line:
                resFind = re.findall('*?FINESTEERING,(+).*?,(+*)',line)[0]
                gpsWeek = re.findall('*?FINESTEERING,(+)',line)[0]
                gpsWeekStr = str(gpsWeek)

                gpsSOW = re.findall('*?FINESTEERING,'+ gpsWeekStr + ',(*)',line)[0]
                gpsSOWStr = str(gpsSOW)

                file.write(gpsWeekStr+','+gpsSOWStr+'\n')
                break
```

Рисунок 4.3 – Лістинг фрагменту програми визначення атрибуту

Після цього послідовно утворюються правила на основі цих подій. Цей процес зображено на екранній формі 4.4.

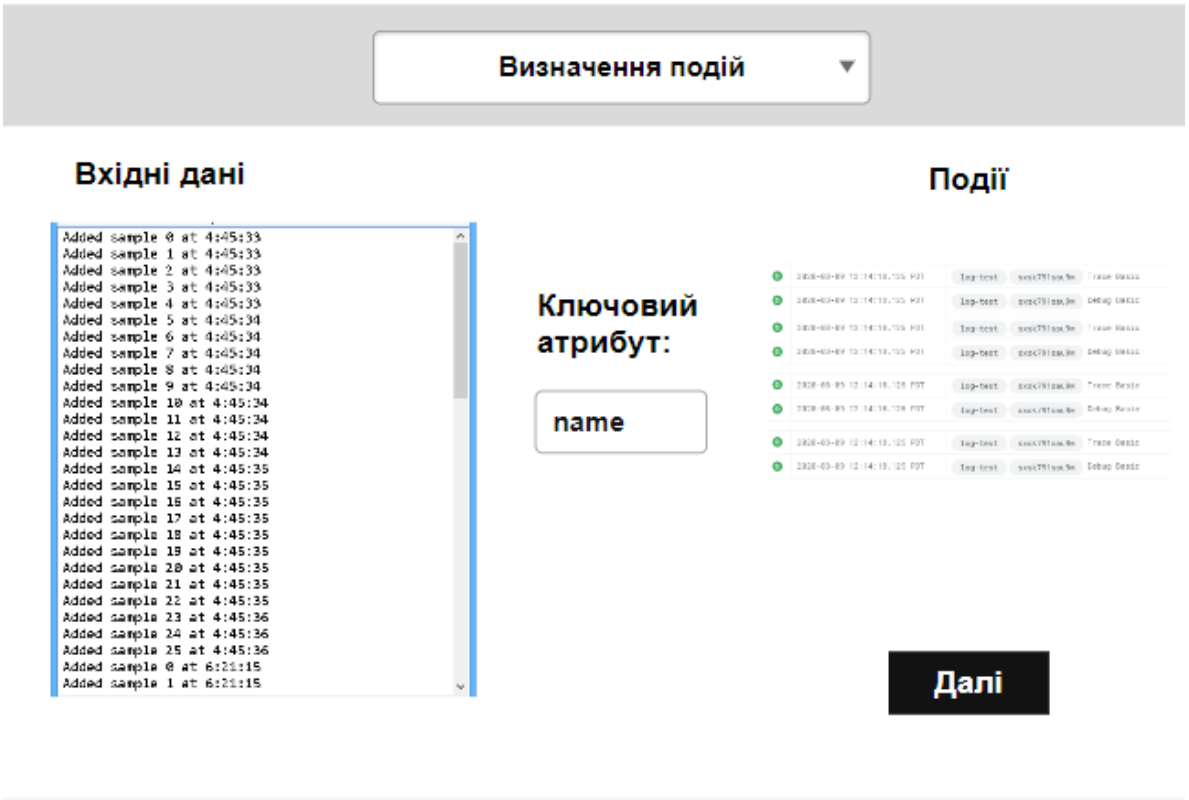


Рисунок 4.4 – Екранна форма прототипу програмі, що реалізує процес формування знань

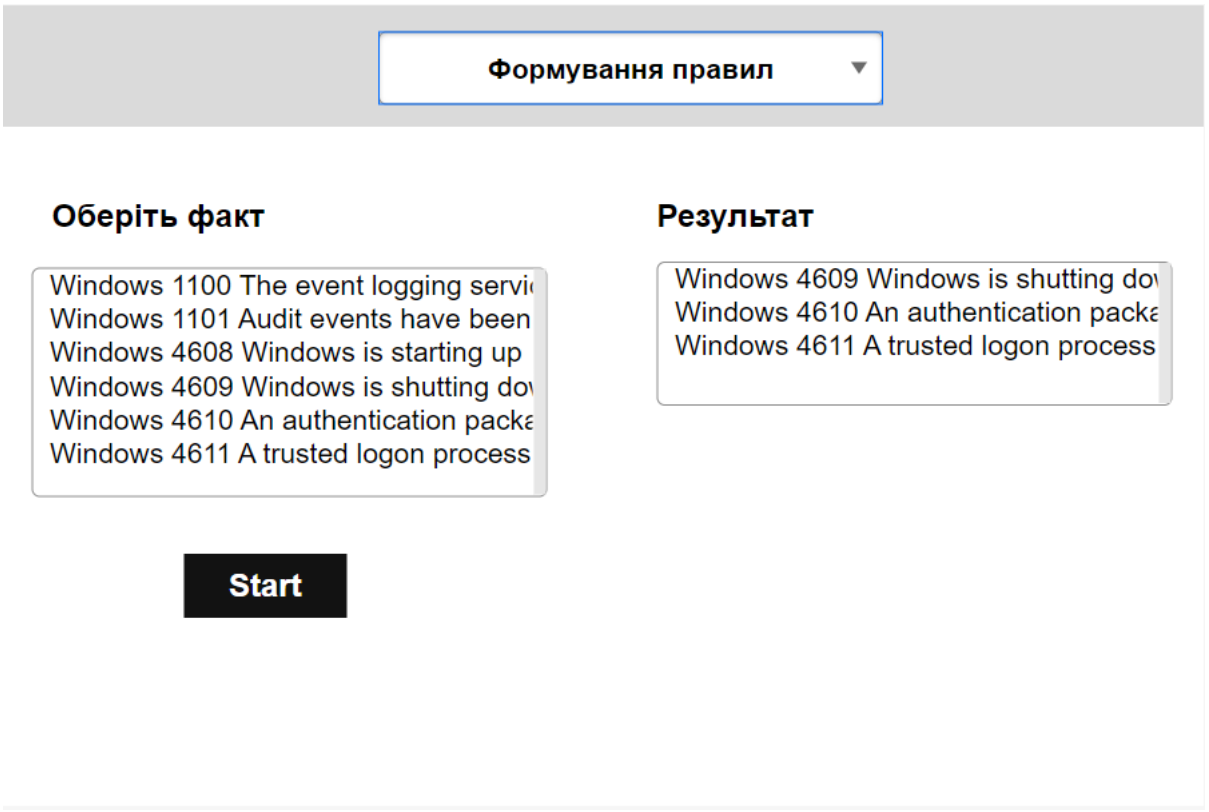


Рисунок 4.5 – Екранна форма прототипу програмі що відображає приклади правил

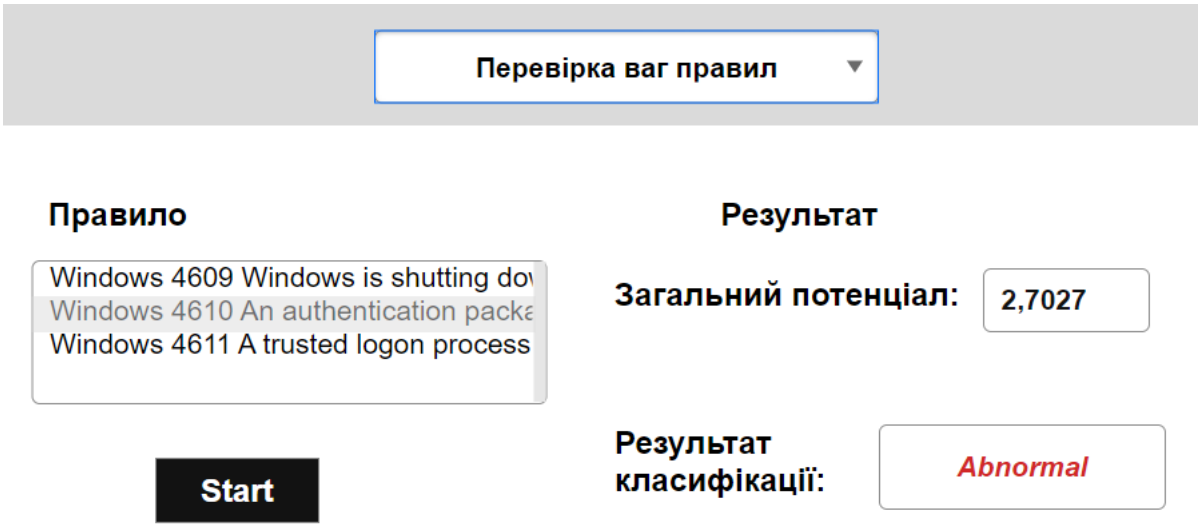


Рисунок 4.6 – Екранна форма прототипу програмі що відображає розрахунок ваг правил

Для простих математичних обчислень у Python можна використовувати вбудовані математичні оператори, такі як додавання (+), віднімання (-), ділення (/) і множення (*). Але більш складні операції, такі як експоненціальні, логарифмічні, тригонометричні чи степеневі функції, не вбудовані.

Python надає модуль, спеціально розроблений для математичних операцій вищого рівня: математичний модуль. Python – це високорівнева динамічно типізована багатопарадигмальна мова програмування. Часто кажуть, що код Python майже схожий на псевдокод, оскільки він дозволяє висловлювати дуже потужні ідеї в невеликій кількості рядків коду, при цьому він дуже читабельний.

Математичний модуль Python є важливою функцією, призначеною для виконання математичних операцій. Він постачається разом із стандартним випуском Python і існує з самого початку. Більшість функцій математичного модуля є тонкими обгортками навколо математичних функцій платформи C. Оскільки основні функції написані на CPython, математичний модуль ефективний і відповідає стандарту C.

Математичний модуль Python пропонує вам можливість виконувати типові та корисні математичні обчислення у програмі. Ось кілька практичних застосувань математичного модуля:

- обчислення комбінацій і перестановок з використанням факторіалів;
- обчислення висоти стовпа за допомогою тригонометричних функцій;
- розрахунок радіоактивного розпаду за експоненціальною функцією;
- розрахунок кривої підвісного моста за допомогою гіперболічних функцій;
- розв’язування квадратних рівнянь;
- моделювання періодичних функцій, таких як звукові та світлові хвилі, за допомогою тригонометричних функцій.

Окрім власне Python використовуються «numpy» – основний пакет для наукових обчислень на Python. Використовується для роботи перш за все із

матрицями, які необхідні у обраному алгоритмі. Використано також Pandas – швидкий, потужний, гнучкий і простий у використанні інструмент аналізу та обробки даних із відкритим кодом, створений на основі мови програмування Python.

NumPy – це розширення для Python, який здебільшого написаний мовою C. Це гарантує, що скомпільовані математичні та числові функції та функції гарантують найвищу можливу швидкість виконання.

NumPy також збагачує мову програмування Python потужними структурами даних для ефективних обчислень із великими масивами та матрицями.

Реалізація націлена навіть на надзвичайно великі («великі дані») матриці та масиви. Крім того, модуль надає величезну кількість високоякісних математичних функцій для роботи з цими матрицями та масивами.

SciPy (науковий Python) часто згадується одночасно з NumPy. SciPy розширює можливості NumPy за допомогою інших корисних функцій, таких як мінімізація, регресія, перетворення Фур'є та багато інших.

І NumPy, і SciPy зазвичай не встановлюються на стандартній установці Python. Однак NumPy та всі інші згадані модулі є частиною дистрибутива Anaconda.

В принципі, Python у поєднанні з NumPy, SciPy, Matplotlib і Pandas можна використовувати як повноцінну заміну MATLAB. Python і його модулі є вільним програмним забезпеченням («безкоштовним програмним забезпеченням» або «з відкритим вихідним кодом»), що означає «безкоштовне», а не «безкоштовне» пиво, навіть якщо Python безкоштовний.

Незважаючи на те, що для MATLAB доступна величезна кількість додаткових наборів інструментів, Python у поєднанні зі згаданими вище модулями має ту перевагу, що Python є більш сучасною та повною мовою програмування.

SciPy додає до Python більше функцій, подібних до MATLAB. Модуль Matplotlib пропонує необхідні функції для створення графіків. Pandas є

наймолодшим членом цього сімейства модулів. Pandas ідеально підходить для роботи з табличними даними, як відомо з програм для роботи з електронними таблицями, таких як Excel.

Модуль pickle реалізує двійкові протоколи для серіалізації та десеріалізації об’єктної структури Python. «Вибір» – це процес, за допомогою якого ієрархія об’єктів Python перетворюється на потік байтів, а «відбір» – це зворотна операція, за допомогою якої потік байтів (з двійкового файлу або байт-подібного об’єкта) перетворюється назад в ієрархію об’єктів.

SciPy надає алгоритми для оптимізації, інтеграції, інтерполяції, задач на власні значення, алгебраїчних рівнянь, диференціальних рівнянь, статистики та багатьох інших класів задач. Алгоритми та структури даних, надані SciPy, широко застосовуються в різних доменах. Високорівневий синтаксис SciPy робить його доступним і продуктивним для програмістів із будь-яким рівнем підготовки та досвіду.

4.2 Експериментальна перевірка удосконаленого методу

Необхідно певним чином керувати сервером і визначити, кому дозволено що, де та коли змінювати за допомогою перевірених бізнес-процесів.

Для виконання експерементальної перевірки методу використовується файл Windows Event log, Його приклад наведено нижче (рисунок 4.6).

Event Log	Level	ID	Error Name	Source
Security	Informational	4740	Account Lockouts	Microsoft-Windows-Security-Auditing
Security	Informational	4728, 4732, 4756	User Added to Privileged Group	Microsoft-Windows-Security-Auditing
Security	Informational	4735	Security-Enabled Group Modification	Microsoft-Windows-Security-Auditing
Security	Informational	4724	Successful User Account Login	Microsoft-Windows-Security-Auditing
Security	Informational	4625	Failed User Account Login	Microsoft-Windows-Security-Auditing
Security	Informational	4648	Account Login with Explicit Credentials	Microsoft-Windows-Security-Auditing

Рисунок 4.6 – Приклад вхідного файлу журналу

Початкова інформація включає часові дані та тимчасові правила. Дані описують поведінку об'єкта керування. Початкові часові дані містять послідовності подій. Необхідно визначити, чи є цей стан аномальним.

За звичайних умов роботи критичні параметри системи не можна змінити, якщо користувачі не мають певних привілеїв, тому моніторинг використання привілеїв і змін в облікових записах і групах користувачів може свідчити про атаку. Наприклад, додавання користувачів до привілейованих груп, таких як адміністратори домену, має відповідати запиту на зміну (RFC).

Якщо у файлі логу знаходиться запис про додання якихось привілеїв може свідчити про зловмисницьку атаку.

Категорії завдань «Керування обліковими записами користувачів» і «Керування групами» засобу перегляду подій.

Коли аудит увімкнено на рядовому сервері, зміни локальних користувачів і груп реєструються, а на контролері домену змінюється Active Directory.

Експериментальна перевірка методу формування знань з урахуванням подій та атрибутів:

– попередня обробка даних;

– створення тестової та навчальної виборок;

– оцінка результату на тестових виборках.

Програмно реалізовані кроки попередньої обробки даних; виявлення подій, побудови правил між подіями, виялвлення підозрілої активності на цих подіях; оцінки отриманих рекомендацій за показником AUC ROC.

Для оцінки роботи методу обрано 4 файлі логу із кількістю записів 100, 500, 1000, 1500.

Результати видалення непотрібних подій наведено у таблиці 4.1.

Таблиця 4.1 – Видалення смислових подій

До обробки	100	500	1000	1500
Після обробки	76	489	820	1429

Набір даних, що використовується має часину даних, яка стосується взаємодії між користувачем та Windows. Це зроблено для подальшого аналізу роботи системи та виявлення підозрілої активності.

Далі за допомогою розробленого програмного модуля було проведено виявлення на подіях знань щодо зловмисницької діяльності. Результат порівняння існуючого методу та удосконаленого наведено у таблиці 4.2.

Таблиця 4.2 – Оцінка виявлення підозрілої активності

Кількість логів	100	500	1000	1500
Метод до удосконалення	83,2%	85%	84,9%	81%
Удосконалений метод	98%	97,5%	97,8%	98,5%

Проведено порівняння методів і зображено результат у вигляді графіку (рисунок 4.7)

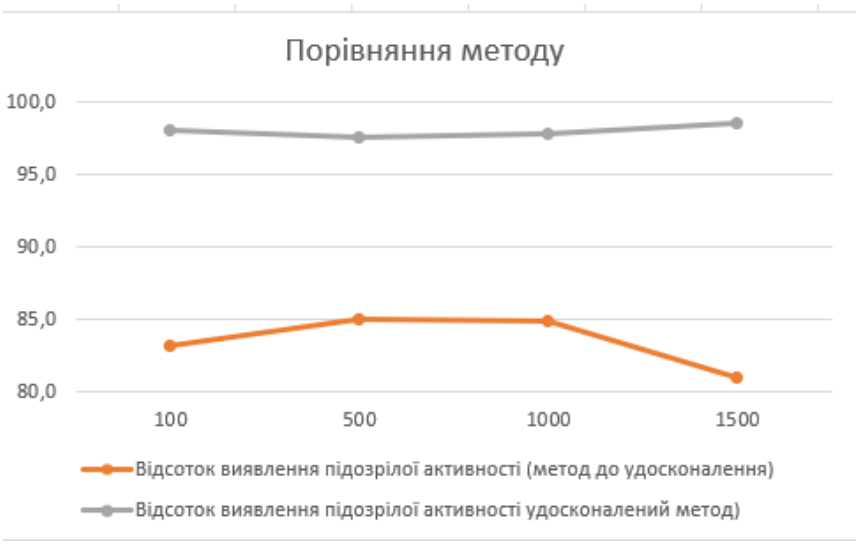


Рисунок 4.7 – Порівняльна характеристика удосконаленого методу на реальних вибірках логів

ВИСНОВКИ

В результаті виконання магістерської роботи був виконаний огляд методів формування знань в процесах підтримки прийняття рішень щодо виявлення атак на комп’ютерні системи.

Під час аналізу існуючих методів вирішення поставленої задачі був наведений загальний опис систем підтримки прийняття рішень, їх класифікація та задачі, які вирішуються для виявлення атак на комп’ютерні системи.

Аналіз способів вдосконалення існуючого методу наведений у розділі два пропонує спосіб формування правил для виявлення аномальних ситуацій завдяки подіям у логах.

У ході виконання роботи було досліджено процес підтримки прийняття рішень, було проаналізовано задачі підтримки прийняття рішень при виявленні атак в комп’ютерних системах, досліджено методи формування знань, удосконалено методу формування знань для підтримки прийняття рішень з виявлення атак в комп’ютерній системі, експериментальна перевірка удосконаленого методу

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Методичні вказівки щодо розробки та оформлення кваліфікаційної роботи (для студентів усіх форм навчання другого (магістерського) рівня програми"Інформаційні управляючі системи та технології) / Упоряд.:Петров К.Е., Левикін В.М., Чалий С.Ф., Євланов М.В., Саєнко В.І., Міхнов Д.К., Міхнова А.В., Чала О.В. - Харків: ХНУРЕ,2021.- 30с.

2. Chalyi S., Pribylnova I. The method of constructing recommendations online on the temporal dynamics of user interests using multilayer graph. EUREKA: Physics and Engineering. 2019. Vol. 3. P. 13-19. (SCOPUS)

3. Chala O. Method for detecting anomalous states of a control object in information systems based on the analysis of temporal data and knowledge. EUREKA. Physics and Engineering. 2018. Vol. 6. P. 28-35. DOI: 10.21303/2461-4262.2018.00787.

4. Чала О.В. Визначення інтегральної оцінки відхилень траєкторій у журналі подій в інформаційних системах процесного управління. ProfIT Conference: Тези допов. І Міжнар. наук.-практ. конф. ІТ-професіоналів та аналітиків комп'ютерних систем (Харків, 24 – 26 квіт. 2018). Харків, НАУ ім. Жуковського, ХАІ, 2018. С. 34-35.

5. Чалий С. Ф. Темпорально-об'єктні моделі події та процесу у завданнях моделювання міркувань на основі прецедентів/С. Ф. Чалий, І. Б. Прибильнова//Уральський науковий вісник. - 2015. -№ 19 (150). - С. 71-75.

6. Чалий С. Ф. Метод адаптивного процесного управління на основі прецедентного підходу/ С. Ф. Чалий, І. В. Левикін//Наукоємні технології № 4 (32), 2016 – С. 410-414.

7. Chalyi S., Leshchynskyi V., Leshchynska I. Method of forming recommendations using temporal constraints in a situation of cyclic cold start of the

recommender system. EUREKA: Physics and Engineering Vol. 4. P. 34-40. (SCOPUS)

8. Segal T. Decision Support System (DSS): What It Is and How Businesses Use Them [Електронний ресурс] / Troy Segal. – 2022. – Режим доступу до ресурсу: https://www.investopedia.com/terms/d/decision-support-system.asp.

9. Chalyi S., Levykin I., Biziuk A., Vovk A., Bogatov I. Development of the technology for changing the sequence of access to shared resources of business processes for process management support. Eastern-European Journal of Eenterprise Technologies, 2020. Vol 2, NO 3 (104). C. 22-29. (SCOPUS).

10. Waltower S. Decision Support Systems Applications and Uses [Електронний ресурс] / Shayna Waltower – Режим доступу до ресурсу: https://www.business.com/articles/decision-support-systems-dss-applications-and-uses/.

11. Чала О. В. Метод побудови контекстно-орієнтованих правил в темпоральній базі знань. Системи управління, навігації та зв'язку. 2018. № 5(51). С. 115-120. DOI: 10.26906/SUNZ.2018.5.115.

12. Чала О. В. Принципи автоматизованої побудови та використання темпоральної бази знань. Системи управління, навігації та зв'язку. 2018. № 6 (52). С. 122-125. DOI: 10.26906/SUNZ.2018.6.122

13. Левикін В. М., Чала О.В. Модель бази знань інформаційної системи процесного управління. Вісник Національного технічного університету «ХПІ». Системний аналіз, управління та інформаційні технології. 2017. № 28(1250). С. 74-78.

14. Петров К. Э., Дейнеко А. А., Чалая О. В., Панферова И. Ю. Метод ранжирования альтернатив при проведении процедуры коллективного экспертного оценивания. Радіоелектроніка, інформатика, управління. № 2(53). 2020. С. 84-94. DOI: 10.15588/1607-3274-2020-2-9 (входить до міжнародної наукометричної бази Web of Science).

15. Chala O. Development of information technology for the automated construction and expansion of the temporal knowledge base in the tasks of

supporting management decisions. Technology audit and production reserves. 2019. № 1/2(45). P. 9-14. DOI: 10.15587/2312-8372.2019.160205.

16. Левикін В. М., Чала О. В. Підтримка прийняття рішень в інформаційно-управляючих системах з використанням темпоральної бази знань. Сучасні інформаційні системи. 2018. Том 2, № 4. С. 101-107. DOI: 10.20998/2522-9052.2018.4.17

17. Chala O. Models of temporal dependencies for a probabilistic knowledge base. Econtechmod. An International Quarterly Journal. 2018. Vol. 7, No. 3. P. 53 – 58.

18. Левикін В. М., Чала О. Концепція автоматизованої побудови бази знань у системі процесного управління. Біоніка інтелекту. 2017. № 2(89). С. 77-83.

19. Чала О.В. Еволюційний підхід до управління життєвим циклом знання-ємних бізнес-процесів. Наукоємні технології. 2017. № 1(33). С. 53-59

20. Чалая О.В. Модель неявных реляционных зависимостей в знаниеемких бизнес-процессах. Проблеми інформаційних технологій. 2016. № 2(020). С. 111-118.

21. О. В. Чала, «Метод ієрархічного виведення в базі знань інформаційноуправляючої системи в парадигмі «Enterprise 2.0», Системи управління навігації та зв’язку, т. 4, №. 50, с. 86–90, 2018.

22. F. Niu, C. Zhang, C. Re, and J. W. Shavlik, «DeepDive: Web-scale knowledge-base construction using statistical learning and inference», in VLDS 2012. 2012, pp. 1-4.

23. X. Song, M. Wu, C. Jermaine, and S. Ranka, «Conditional Anomaly Detection», IEEE Transaction on Knowledge and Data Engineering, vol. 19, no. 5, pp. 631–645, 2007

24. О. В. Чала, «Побудова темпоральних правил для представлення знань в інформаційно-управляючих системах», Сучасні інформаційні системи, т. 2, №3, с. 54–59, 2018. doi: 10.20998/2522-9052.2018.3.09.

25. R. Das, A. Godbole, N. Monath, M. Zaheer, and A. McCallum, «Probabilistic case-based reasoning for open-world knowledge graph completion», in Findings of the association for computational linguistics: EMNLP 2020, 2020, pp. 4752–4765. doi: 10.18653/v1/2020.findingsemnlp.427

26. O. Chala, «Logical-probabilistic representation of casual dependencies between events in business-process management», Advanced information systems, vol. 2, no. 2, pp. 40–44, 2018.

27. S. Wu et al., «Fonduer: Knowledge Base Construction from Richly Formatted Data», in Proceedings of the 2018 International Conference on Management of Data, Houston TX USA, 2018, pp. 1301–1316. doi: 10.1145/3183713.3183729.

28. J. Han, W. Gong, and Y. Yin, «Mining segment-wise periodic patterns in time-related databases», Proc Fourth Intl Conf Knowl. Discov. Data Min., pp. 214-218, 1998.

29. О. В. Чала, «Побудова подієвої складової бази знань в рамках системи управління підприємством», на V Всеукр. наук.-практ. конф. Інформаційні технології 2018 (IT-2018), 2018, с. 143-145.

30. V. M. Levykin and O. V. Chala, «Automated knowledge base construction using process logs», in XIII International scient.- pract. confer. Scientific progress news, 2017, pp. 26-28.

31. ДСТУ 3008:2015. Інформація та документація. Звіти у сфері науки і техніки. Структура і правила оформлювання. . – Чинний від 22.06.2015. – Київ: ДП «УкрНДНЦ», 2016. – 31 с.

32. ДСТУ 8302:2015. Інформація та документація. Бібліографічні посилання. Загальні положення та правила складання. – Чинний від 04.03.2016. – Київ: ДП «УкрНДНЦ», 2016. – 20 с.