Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
“____” _____ 2025 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за другим (магістерським) рівнем вищої освіти
на тему
“Дослідження та програмна реалізація системи поведінкового аналізу користувачів за допомогою концепції UEBA”

Виконав здобувач вищої освіти
ІІ курсу, групи КН-24М
ОПП «Комп’ютерні науки»
спеціальності 122 «Комп’ютерні науки»
_____ Ковальчук В.А.
«____» _____ 2025 р.

Керівник проекту
кандидат технічних наук
_____ Смірнова Т.В.
«____» _____ 2025 р.
Рецензент _____
_____

м. Кропивницький

АНОТАЦІЯ

Ковальчук В.А. Дослідження та програмна реалізація системи поведінкового аналізу користувачів за допомогою концепції UEBA. 122 Комп'ютерні науки. Центральноукраїнський національний технічний університет. Кропивницький. 2025.

В даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи поведінкового аналізу користувачів за допомогою концепції UEBA.

Метою розробки є дослідження та програмна реалізація системи поведінкового аналізу користувачів за допомогою концепції UEBA.

Об'єктом дослідження є процес поведінкового аналізу користувачів за допомогою концепції UEBA.

Предметом дослідження є методи поведінкового аналізу користувачів за допомогою концепції UEBA.

Методи дослідження базуються на методах машинного навчання, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація системи поведінкового аналізу користувачів за допомогою концепції UEBA.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ з ОС Windows 10/11.

Програму розроблено в середовищі PHP фреймворк Yii2.

Ключові слова: комп'ютерні науки, поведінковий аналіз, UEBA

ABSTRACT

Kovalchuk V.A. Research and software implementation of a user behavioral analysis system using the UEBA concept. 122 Computer Science. Central Ukrainian National Technical University. Kropyvnytskyi. 2025.

In this final qualification work for the second (master's) level of higher education, software has been developed, which is intended for a user behavioral analysis system using the UEBA concept.

The purpose of the development is the research and software implementation of a user behavioral analysis system using the UEBA concept.

The object of the research is the process of user behavioral analysis using the UEBA concept.

The subject of the research is the methods of user behavioral analysis using the UEBA concept.

The research methods are based on machine learning methods, mathematical statistics methods, and software development methods.

The result of the work is a software implementation of a user behavioral analysis system using the UEBA concept.

In the process of working on the software model, an analysis of existing hardware and software tools was performed. All components of the developed software are fully described.

A user-friendly user interface has been developed. Instructions for working with the software are provided.

The program can be used on a PC with OS Windows 10/11.

The program was developed in the PHP framework Yii2.

Keywords: computer science, behavioral analysis, UEBA

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

ЕОМ	– електрона обчислювальна машина
ІБ	– інформаційна безпека
ІТ	– інформаційні технології
КС	– комп’ютерна система
КТЗ	– комплекс технічних засобів
МЕТОЗ	– методичне забезпечення
ОС	– операційна система
ОРЗ	– організаційне забезпечення
ПЗ	– програмне забезпечення
ПК	– персональний комп’ютер
ПП	– програмний продукт
ППЗ	– прикладне програмне забезпечення
ППП	– пакет прикладних програм
БД	– база даних
DLL	– бібліотека динамічної компоновки
UEBA	– User and Entity Behavior Analytics

ВСТУП

Актуальність теми. UEBA, що розшифровується як User and Entity Behavior Analytics (аналіз поведінки користувачів та сутностей), – це технологія кібербезпеки, яка аналізує поведінку користувачів та сутностей для виявлення аномальної та потенційно шкідливої діяльності. Вона виходить за рамки традиційних заходів безпеки, зосереджуючись на моделях поведінки, а не лише на відомих загрозах, використовуючи машинне навчання та розширену аналітику для виявлення відхилень від звичайної активності. Це допомагає організаціям виявляти внутрішні загрози, скомпрометовані облікові дані та інші складні атаки.

Мета й завдання дослідження. Метою роботи є дослідження та програмна реалізація системи поведінкового аналізу користувачів за допомогою концепції UEBA.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

– Огляд існуючих систем поведінкового аналізу користувачів за допомогою концепції UEBA.

– Дослідження системи поведінкового аналізу користувачів за допомогою концепції UEBA.

– Програмна реалізація системи поведінкового аналізу користувачів за допомогою концепції UEBA.

Об’єктом дослідження є процес поведінкового аналізу користувачів за допомогою концепції UEBA.

Предметом дослідження є методи поведінкового аналізу користувачів за допомогою концепції UEBA.

Методи дослідження базуються на методах машинного навчання, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено метод поведінкового аналізу користувачів за допомогою концепції UEBA.

– Розроблено вітчизняний продукт поведінкового аналізу користувачів за допомогою концепції UEBA, який має більш широкі можливості, на відміну від існуючих аналогів.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі поведінкового аналізу користувачів за допомогою концепції UEBA.

Достовірність наукових результатів підтверджена теоретичними викладеннями, даними комп'ютерного моделювання, коректними дослідженнями параметрів на функціонуючій обчислювальній мережі, а також відповідністю отриманих результатів окремим результатам, наведеним у науковій літературі.

Робота апробована на LVII Науково-технічній конференції здобувачів вищої освіти LV науково-технічної конференції «Наука в ЦНТУ: основні досягнення та перспективи розвитку» (2025 р.), основні положення випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти надруковані у статті збірника праць молодих науковців ЦНТУ, випуск №15.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи поведінкового аналізу користувачів за допомогою концепції UEBA, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Зловмисники усередині компанії намагаються знайти інформацію, що не повинна їх цікавити за родом діяльності або просто проявляють незвичайну активність і зайву допитливість. Якщо співробітник протягом дня відкрив занадто багато мережних папок або роздрукував кілька сотень сторінок, рішення класу UEBA відстежить аномалію й повідомить про неї службі безпеки.

Для виявлення витоків даних застосовуються системи DLP, але контролювати всі канали вони не можуть – методи передачі даних стають усе складніші й трудомісткіші. Рішення UEBA набагато краще виявляють навіть спроби переслати конфіденційну інформацію з електронної пошти – за зрослим розміром й кількістю листів, а також по інших ознаках, задати які в статичних правилах DLP буде важко.

Дуже часто співробітникам дозволено підключатися до корпоративних ресурсів з вилучених площадок, але далеко не всі використовують цю можливість – якщо раптом головний бухгалтер увійде в систему в невизначений час або з незвичайного місця, система UEBA відреагує на інцидент. Те ж саме відбудеться у випадках, коли користувач ділиться реквізитами для доступу до корпоративних ресурсів з іншими співробітниками або при помилці настроювання прав доступу.

Звичайно ці речі контролюються системою керування обліковими даними (IDM – від англ. Identity management) або, у випадку із привілейованими користувачами – системами PUM (Privileged User Management). Рішення UEBA здатне частково взяти на себе й це завдання у випадку, якщо користувачеві помилково дали занадто широкі повноваження або, скажемо, системний адміністратор намагається виконати підозрілі дії: копіює занадто великі обсяги

даних, активно підключається до робочих станцій і т.д. Система PUM може служити джерелом відомостей для поведінкового аналізу, що виявить будь-яку аномалію.

1.2 Область застосування

Як ми вже писали, класичні системи включають елементи поведінкового аналізу, але вони відслідковують дії користувача по статично заданих правилах і далеко не завжди здатні відреагувати на аномалію. Важливою відмінністю систем UEBA є здатність до самонавчання (горезвісний machine learning): величезну роль тут грають кількість джерел інформації про типовий робочий день конкретного співробітника й динамічне коректування його профілю на основі методів математичної статистики для зменшення відсотка помилкових спрацьовувань. По суті, рішення UEBA призначені не тільки для визначення ризиків, їхнє завдання полягає в мінімізації, що відправляються на ручну перевірку офіцерами безпеки проблем – працюючі за твердими правилами класичні системи впоратися із цим у принципі не здатні.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи поведінкового аналізу користувачів за допомогою концепції UEBA, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти

Forcepoint UEBA

Торговельна марка Forcepoint з'явилася в 2016 році. Її утворення пов'язане з об'єднанням компаній Websense, Raytheon, Stonesoft, Sidewinder і створенням під єдиним брендом нової лінійки продуктів у сфері кібербезпеки. Випущені рішення включають більш ніж 20-літній досвід розробок попередників. Застосовуються одні з найефективніших методик детектування, блокування онлайн-загроз.

Компанія Forcepoint пропонує системно-орієнтований підхід до створення системи фільтрації інформаційно-контентного трафіку усередині організації. Подібний підхід дозволяє забезпечувати клієнтів ефективними інструментами з низькою вартістю впровадження.

Продукт Forcepoint UEBA є інструментом для поведінкового аналізу в системах з підключенням до корпоративних баз даних. Використовується з метою детектування й ідентифікації підозрілих дій користувачів, додатків. Проводить аналіз фактів змін реєстру, операцій з файлами, об'єктами операційної системи.

Крім фіксації будь-яких інцидентів, здійснюється виявлення скомпрометованих облікових записів, хостів в IT-інфраструктурі. Виміряється великий перелік факторів – від обсягу переписки до розподілу подій по групах особливого контролю.

Rapid7 InsightIDR

Rapid7 InsightIDR – це комбінована платформа SIEM та XDR, яка постійно відстежує звичайну активність користувачів поза межами визначених індикаторів компрометації. Вона розроблена для виявлення важковиявлюваних загроз, таких як зловмисники, які видають себе за співробітників компанії, або спроби витоку даних з боку зловмисних інсайдерів. Якщо ви шукаєте комбіноване рішення безпеки, InsightIDR – чудовий вибір. Його потужний набір функцій та адміністративні можливості роблять його нашим найкращим UEBA загалом.

Переваги:

- Безліч розширених функцій.
- Надана відносно прозора інформація про ціни.
- Безкоштовний пробний період триває місяць.

Недоліки:

- Вивчення платформи може зайняти деякий час.
- Нечітка функція пріоритезації інцидентів.
- Немає підтримки пристроїв Інтернету речей.

Основні характеристики:

- Автоматизована кореляційна активність: InsightIDR пов'язує події в мережі клієнта з конкретними користувачами та об'єктами, які стоять за цими подіями.

- Базові показники активності користувачів: Рішення адаптується до користувачів та об'єктів у мережі, щоб продовжувати визначати нормальну поведінку.

- Список спостерігається: Панель керування InsightIDR відстежує користувачів, які можуть становити значно підвищений ризик.

- Виявлення неправильної конфігурації: InsightIDR використовує візуальний пошук у журналах та попередньо створені картки відповідей для виявлення аномалії.

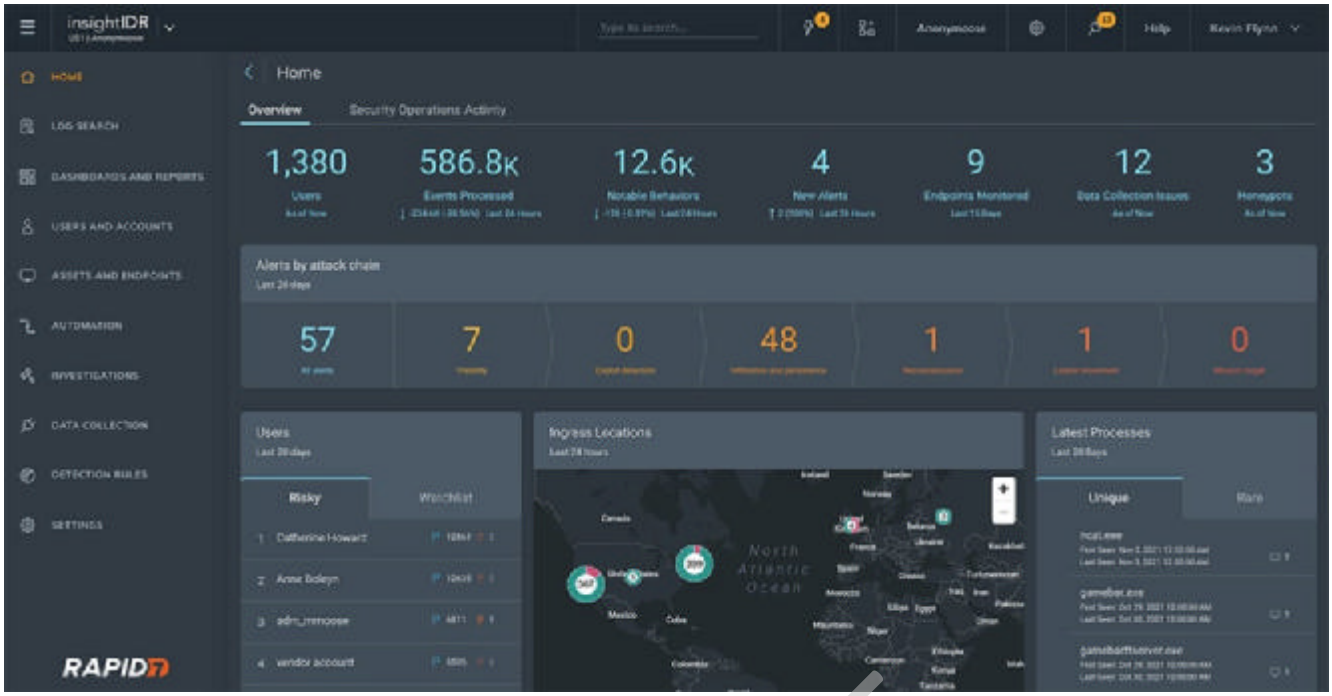

Рисунок 2.1 – Інтерфейс користувача Rapid7 InsightIDR

Microsoft Sentinel – це хмарне рішення SIEM, яке також надає функціональність UEBA. Його функції включають пріоритезацію інцидентів, групування однорангових пристроїв та часові рамки інцидентів. Окрім цих інструментів, Sentinel пропонує багато інших функцій UEBA та SIEM – розгляньте Sentinel, якщо ви особливо зосереджені на широких можливостях безпеки. Він також інтегрується з Defender, продуктом Microsoft XDR, і є чудовим вибором для компаній, що використовують хмарні технології Azure.

Переваги:

- Безліч основних та розширених функцій UEBA.
- Доступно як керована послуга.
- Чудово підходить для організацій Windows.

Недоліки:

- Обмежені варіанти розгортання.
- Бракує демонстрації продукту.
- Структура може не працювати для деяких команд.

Основні характеристики:

– Широкий збір даних: Sentinel отримує інформацію з усіх користувачів, пристроїв, програм та інфраструктури, як локально, так і в кількох хмарних сховищах.

– Виявлення бічного руху: Коли Sentinel позначає свідому поведінку, ваша команда може спостерігати за ефективним бічним рухом між програмами або службами.

– Розслідування на основі поведінкового інтелекту: штучний інтелект вам допоможе швидше дослідити загрози та виявити дивнуку, ніж ви могли б шукати вручну.

– Без обмежень щодо запитів: Оскільки Sentinel є хмарними рішеннями, він уникає обмежень, які іноді заважають локальним системам захисту підприємства.

FortiSIEM

FortiSIEM – це комплексний продукт SIEM, що пропонується відомим постачальником мережевої безпеки Fortinet. Він включає функції UEBA, такі як виявлення внутрішніх загроз, оцінка ризиків для користувачів та виявлення скомпрометованих облікових записів. FortiSIEM – це надійне рішення для будь-якого бізнесу. Тим не менш, він особливо корисний для команд, які вже використовують мережеві пристрої Fortinet, такі як брандмауери, оскільки він інтегрується з пристроями FortiGate, дозволяючи їм обмінюватися даними.

Переваги:

– Безліч адміністративних функцій, таких як API.

– Можна розгортати в хмарі та локально.

– FortiSIEM підтримує пристрої Інтернету речей.

Недоліки:

– Відсутність деяких функцій UEBA.

– Без безкоштовної пробної версії.

– Немає інструкцій з нативного реагування.

Основні характеристики:

– Виявлення аномальної поведінки кінцевої точки: дані кінцевої точки можуть відображати скомпрометовану систему чи обліковий запис, а також недбайливу чи зловмисну діяльність інсайдера.

– Аналітика загроз: FortiSIEM підтримує джерела загроз, які надають CSV-файл або підтримують стандарти STIC/TAXII 1.0, 1.1 та 2.0.

– Механізми кореляції в реальному часі: FortiSIEM може запустити всі активні правила кореляції на польоті.

– Виявлення на основі машинного навчання: адміністратори можуть переглядати аномальну поведінку та користувачів, не маючи потреби писати всі власні правила.

LogRhythm

LogRhythm пропонує потужну платформу SIEM з функціональністю UEBA, яка використовує машинне навчання для виявлення аномалій, таких як внутрішні загрози, атаки методом повного перебору та зловживання адміністратором. Функція моніторингу цілісності файлів LogRhythm виявляє несанкціонований доступ до файлів; розгляньте LogRhythm, якщо ви зберігаєте велику кількість файлів з конфіденційними даними компанії. Повне налаштування LogRhythm може зайняти місяці; це гарне рішення для досвідчених команд із розширеними вимогами до SIEM.

Переваги:

– Найкраща SIEM-платформа з можливостями UEBA.

– Моніторинг файлів допомагає командам захищати конфіденційні дані.

– Можна розгорнути локально або в хмарі.

Недоліки:

– Може бути занадто складним для менших команд.

– Бракує прозорої інформації про ціни.

– Без безкоштовної пробної версії.

Основні характеристики:

– Інтеграція з розвідкою загроз: LogRhythm SIEM інтегрується як з комерційними, так і з відкритими каналами даних про загрози.

– Індивідуальні оцінки аномалій: використовуйте ці оцінки та зведені оцінки користувачів, щоб налаштувати параметри критичних загроз для дослідження та усунення наслідків.

– Автоматизовані дії SmartResponse: LogRhythm допоможе зменшити ручну роботу, автоматизуючи реагування на загрози, такі як карантин файлів та блокування URL-адреси.

– Моделі аналізу: Платформа вибирає, коли ідентифікація користувача є аномальною залежністю від власного базового рівня, рівноправними особами або всіма контрольованими ідентифікаторами.

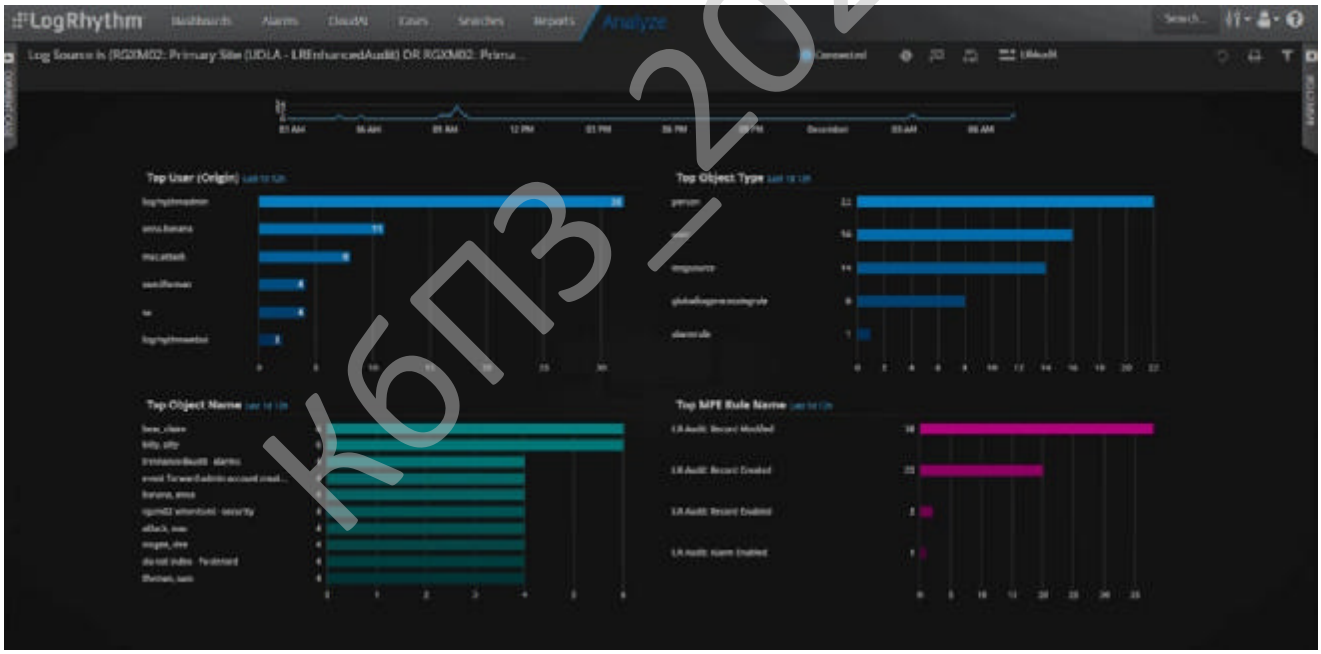

Рисунок 2.2 – Інтерфейс користувача LogRhythm SIEM.

Cynet 360 AutoXDR

Cynet 360 AutoXDR – це розширена платформа виявлення та реагування, яка пропонує організаціям єдину багатокористувацьку платформу, що об'єднує

функції безпеки кінцевих точок , користувачів та мережі в одному пакеті. UBA належить до мережевої безпеки платформи. Оскільки платформа XDR від Cynet пропонує широкий спектр функцій UBA та інші інструменти безпеки, вона є гарним вибором для великих організацій, особливо для команд, яким також потрібні засоби виявлення та реагування в мережі.

Переваги:

- Доступний як цілодобовий сервіс моніторингу.
- Розширені функції виявлення та реагування.
- Доступний API.

Недоліки:

- Вам потрібна повна платформа, щоб отримати доступ до функцій UEBA.
- Може бути непосильним для невеликих команд.
- Бракує навчальних відео з продукту.

Основні характеристики:

- Оркестрація реагування Cynet: Набір заходів з виправлення ситуації командою вирішувати проблеми із зараженими хостами, шкідливими файлами, мережевим трафіком та скомпрометованими обліковими записами користувачів.
- CyOps: Цілодобова служба MDR від Cynet, що складається з експертів SOC , завершує проведення поглиблених розслідувань, проактивування загроз та звітів про атаки.
- Моніторинг користувачів: Cynet шукає аномальну поведінку, яка вказує на скомпрометовані облікові записи користувачів.
- Визначення рівня ризику: Функціональність платформи UEBA використовує вичерпну інформацію про користувача для визначення загального рівня ризику цього користувача.

Exabeam

Exabeam – це постачальник послуг безпеки, який пропонує UEBA як одну зі своїх основних можливостей. Ключові функції включають графіки подій, контроль доступу на основі ролей та можливість зберегти вашу існуючу платформу SIEM. Exabeam інтегрується із сотнями сторонніх інструментів безпеки; також існує велика різноманітність, включаючи джерела даних, не пов'язані з безпекою, такі як Salesforce. Я рекомендую Exabeam організаціям, яким потрібен широкий спектр джерел даних.

Переваги:

- Безліч інтеграцій, включаючи ті, що не стосуються безпеки.
- Доступні «розумні» графіки подій.
- API доступний для розробників.

Недоліки:

- Можливості автоматизованого виправлення є нечіткими.
- Без безкоштовної пробної версії.
- Недоступно як керована послуга.

Основні характеристики:

- Інтеграція з постачальниками SIEM: Exabeam дозволяє клієнтам інтегрувати рішення UEBA з рішеннями SIEM, які вони, можливо, вже купують.
- Інші додаткові вбудовані інтеграції: вимкнення SIEM, Exabeam інтегрується з такими продуктами, як Microsoft 365, VMware ESXi, Salesforce та CrowdStrike.
- Аналітика поведінки: Exabeam має сумнівні сигнали від кількох продуктів для пошуку складних загроз.
- Групи однорангових користувачів: Exabeam сортує як користувачів, так і інші сутності, такі як пристрої, залежно від їхнього зв’язку, наприклад, внутрішніх користувачів в одному бізнес-відділі.

Splunk UBA

Splunk UBA – це рідкісний автономний продукт для поведінкової аналітики, який надає командам можливості, необхідні для моніторингу, аналізу та виявлення загроз користувачам. Організації можуть покращити свою безпеку за допомогою робочого процесу Splunk щодо загроз незалежно від розміру бізнесу чи набору навичок. Ця технологія звужує кількість первинних подій у масштабі бізнесу до кількох можливих загроз. Я рекомендую Splunk тим командам, яким потрібне автономне рішення, призначене виключно для аналітики користувачів.

Переваги:

- Обробляє та фільтрує мільярди необроблених подій.
- на основі споживання дозволяє вам платити по мірі використання.
- Можливість мати технічного менеджера облікового запису.

Недоліки:

- Бракує деяких функцій, які мають його конкуренти.
- Не зосереджено на аналітиці сутностей, лише на користувачах.
- Немає інтеграції з Active Directory.

Основні характеристики:

- Пріоритетність інцидентів: кожна аномалія отримує власний бал ризику, а Splunk допомагає вашій команді реагувати на найважливіші проблеми в першу чергу.

- Моделі численних аномалій та загроз: ці моделі Splunk зосереджені на виявленні загроз ззовні вашої організації.

- Розширене виявлення загроз: продукт розроблений для виявлення захоплення облікового запису (ATO), діяльності з керування та контролю, а також експлойтів браузера.

- Виявлення витоку даних: Splunk шукає вилучення або передачу конфіденційної інформації компанії зі своїх місць зберігання.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

PHP (англ. PHP: Hypertext Preprocessor – PHP: гіпертекстовий препроцесор), попередня назва: Personal Home Page Tools – скриптова мова програмування, була створена для генерації HTML-сторінок на стороні веб-сервера. PHP є однією з найпоширеніших мов, що використовуються у сфері веб-розробок (разом із Java, .NET, Perl, Python, Ruby). PHP підтримується переважною більшістю хостинг-провайдерів. PHP – проект відкритого програмного забезпечення.

PHP інтерпретується веб-сервером у HTML-код, який передається на сторону клієнта. На відміну від скриптової мови JavaScript, користувач не бачить PHP-коду, бо браузер отримує готовий html-код. Це є перевага з точки зору безпеки, але погіршує інтерактивність сторінок. Але ніщо не забороняє використовувати PHP для генерування і JavaScript-кодів які виконуються вже на стороні клієнта. PHP – мова, код якої можна вбудовувати безпосередньо в html-код сторінок, які, у свою чергу, будуть коректно оброблені PHP-інтерпретатором. Обробник PHP просто починає виконувати код після відкриваючого тегу (<?php) і продовжує виконання до того моменту, поки не зустріне закриваючий тег (?>).

Велика різноманітність функцій PHP дає можливість уникати написання багаторядкових функцій, призначених для користувача, як це відбувається в C або Pascal.

В PHP вбудовані бібліотеки для роботи з MySQL, PostgreSQL, mSQL, Oracle, dbm, Hyperware, Informix, InterBase, Sybase.

Мова PHP здаватиметься знайомою програмістам, що працюють в різних областях. Багато конструкцій мови запозичені з C, Perl. Код PHP дуже схожий на той, який зустрічається в типових програмах на C або Pascal. Це помітно знижує початкові зусилля при вивченні PHP. PHP – мова, що поєднує переваги Perl і C і спеціально спрямована на роботу в Інтернеті, мова з універсальним і зрозумілим

синтаксисом. І хоча PHP є досить молодою мовою, вона здобула таку популярність серед web-програмістів, що в наш час є мало не найпопулярнішою мовою для створення веб-застосунків (скриптів). [3]

Ефективність є дуже важливим чинником у програмуванні для середовищ розрахованих на багато користувачів, до яких належить і web. Важливою перевагою PHP є те, що ця мова належить до інтерпретованих. Це дозволяє обробляти сценарії з достатньо високою швидкістю. За деякими оцінками, більшість PHP-сценаріїв (особливо не дуже великих розмірів) обробляються швидше за аналогічні їм програми, написані на Perl. Проте хоч би що робили розробники PHP, виконавчі файли, отримані за допомогою компіляції, працюватимуть значно швидше – в десятки, а іноді і в сотні разів. Але продуктивність PHP достатня для створення цілком серйозних веб-застосунків.

Yii2 – це високопродуктивний та швидкодіючий веб-фреймворк, написаний на PHP, реалізує парадигму модель-вид-контролер. Yii – скорочення від «Yes It Is!».

Головними перевагами і можливостями фреймворку є:

– Реалізація архітектурного шаблону Модель-вид-контролер.
– Інтерфейси DAO та Active Record для роботи з базами даних.
– Підтримка інтернаціоналізації.
– Кешування сторінок та окремих фрагментів.
– Перехоплення та обробка помилок.
– Введення та валідація веб-форм.
– Генерація базового PHP-коду для CRUD-операцій.
– Використання AJAX та інтеграція з jQuery.
– Міграції бази даних.
– Можливість підключення сторонніх бібліотек [6].

Спільнота розробників використовуючих фреймворк активно розвивається та розроблює все більше нових корисних компонентів, які легко можно підключити і використовувати при роботі з фреймворком.

Перевага Yii над іншими фреймворками полягає у ефективності, широких можливостях та якісній документації. Yii з самого початку дуже ретельно спроектований для відповідності всім вимогам при розробці серйозних веб-додатків. Дякуючи своїй легковагості та наявності просунутих засобів кешування, Yii особливо підходить для розробки додатків із великим потоком трафіку, таких як портали, форуми, системи управління контентом (CMS), системи електронної коммерції та ін.

Подібно більшості інших PHP-фреймворків, Yii – це MVC-фреймворк.

MVC (або Модель-вид-контролер, англ. Model-view-controller, MVC) – архітектурний шаблон, який використовується під час проектування та розробки програмного забезпечення. Цей шаблон поділяє систему на три частини: модель даних, вигляд даних та керування.

Мета шаблону – гнучкий дизайн програмного забезпечення, який повинен полегшувати подальші зміни чи розширення програм, а також надавати можливість повторного використання окремих компонентів програми. Крім того використання цього шаблону у великих системах призводить до певної впорядкованості їх структури і робить їх зрозумілішими завдяки зменшенню складності.

Модель (model) – це екземпляр класу CModel або класу, успадкованого від нього. Модель використовується для зберігання даних або застосовних до них бізнес-правил.

Представлення – це PHP-скрипт, який складається переважно з елементів користувальницького інтерфейсу. Він може охоплювати вирази PHP, проте рекомендується, щоб ці вирази не змінювали дані і залишалися відносно простими. Згідно концепції розділення логіки та представлення, більша частина коду логіки повинна бути розміщена у контролері або моделі, а не у скрипті представлення.

Контролер (controller) – це екземпляр класу CController або успадкованого від нього класу. Він створюється об’єктом додатку тоді, коли користувач робить

відповідний запит. При запуску контролер виконує відповідну дію, що зазвичай передбачає створення відповідних моделей і рендеринг необхідних представлень. У найпростішому випадку дія – це метод класу контролера, назва якого починається на action.

Модель інкапсулює ядро даних і основний функціонал з їх обробки. Також компонент Модель не залежить від процесу введення або виведення даних. Компонент виводу Представлення може мати декілька взаємопов'язаних областей, наприклад, різні таблиці і поля форм, в яких відображається інформація. У функції Контролера входить моніторинг за подіями, що виникають в результаті дій користувача (зміна положення курсора миші, натиснення кнопки або введення даних в текстове поле).

Зареєстровані події транслюються в різні запити, що спрямовуються компонентам Моделі або об'єктам, відповідальним за відображення даних. Відокремлення моделі від вигляду даних дозволяє незалежно використовувати різні компоненти для відображення інформації. Таким чином, якщо користувач через Контролер внесе зміни до Моделі даних, то інформація, подана одним або декількома візуальними компонентами, буде автоматично відкорегована відповідно до змін, що відбулися. [4]

Для розробки програмного забезпечення було обрано середовище розробки JetBrains PhpStorm – крос-платформове інтегроване середовище розробки для PHP, яке розробляється компанією JetBrains на основі платформи IntelliJ IDEA.

PhpStorm являє собою інтелектуальний редактор для PHP, HTML і JavaScript з можливостями аналізу коду на льоту, запобігання помилок у сирцевому коді і автоматизованими засобами рефакторинга для PHP і JavaScript. Автодоповнення коду в PhpStorm підтримує специфікацію PHP 5.3, 5.4 та 5.5 (сучасні і традиційні проекти), включаючи генератори, співпрограми, простори імен, замикання, типажі і синтаксис коротких масивів. Присутній повноцінний SQL-редактор з можливістю редагування отриманих результатів запитів.

PhpStorm розроблений на основі платформи IntelliJ IDEA, написаної на Java. Користувачі можуть розширити функціональність середовища розробки за рахунок установки плаґінів, розроблених для платформи IntelliJ, або написавши власні плаґіни.

PhpStorm надає багатий і інтелектуальний редактор коду для PHP з підсвічуванням коду, розширеною конфігурацією форматування коду, перевіркою на наявність помилок на льоту і розумним автодоповненням.

Основними перевагами PhpStorm є:

– Підтримка PHP 5.3, 5.4 та 5.5, включаючи генератори, співпрограми, простори імен, замикання, типажі, синтаксис коротких масивів, доступ до члена класу при інстанціюванні, розіменування масиву при виклику функції, бінарні літерали, вираження в статичних виклики тощо. PhpStorm може використовуватися як для сучасних, так і для традиційних проектів на PHP.

– Автодоповнення коду фіналізують класи, методи, імена змінних, ключові слова PHP, а також широко використовувані імена полів і змінних залежно від їхнього типу.

– Підтримка стандартів оформлення коду (PSR1/PSR2, Drupal, Symfony2, Zend).

– Підтримка PHPDoc. PhpStorm надає відповідне автодоповнення коду, засноване на анотаціях @property, @method і @var.

– Детектор дубльованого коду.

– PHP Code Sniffer (phpcs), котрий перевіряє код на льоту

– Рефакторинги (перейменування, введення змінної/константи/поля, вбудовування змінної).

– Підтримка редагування шаблонів Smarty (підсвічування синтаксичних помилок, автодоповнення функцій і атрибутів Smarty, автоматична вставка парних дужок, лапок і закриваючих тегів тощо).

– MVC подання для фреймворків Symfony2 і Yii.

– Розпізнавання коду, запакованого в PHAR-архіви.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускну кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи поведінкового аналізу користувачів за допомогою концепції UEBA.

В процесі розробки випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформувати висновки про виконаний обсяг робіт та одержані результати.

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Ключові характеристики інструментів UEBA

Хоча UEBA як категорія безпеки зараз часто вписується в рамки більших платформ, кілька ключових можливостей є однаковими для всієї галузі. Основні функції UEBA включають моніторинг інфраструктури, аналітику, сповіщення та керування користувачами.

Моніторинг

В інфраструктурі безпеки мережі, пристрої та програми повинні контролюватися. Інструменти UEBA постійно спостерігають за IT-системами та повідомляють адміністраторів, коли мережевий трафік та поведінка пристроїв або програм не відповідають попередньо налаштованим стандартам.

Аналітика

У рішеннях UEBA поведінкова аналітика базується на технології машинного навчання. ML ідентифікує поведінку користувачів, щоб визначити, чи відповідає вона заздалегідь визначеним критеріям типових дій. Якщо інструмент UEBA вирішує, що нестабільна поведінка користувача є небезпечною, він виділяє цю закономірність на панелі інструментів, щоб адміністратори безпеки могли її переглянути.

Сповіщення та пріоритетність

Інструменти UEBA запускають сповіщення, коли виникає достатньо значна аномалія. Оскільки ці інструменти вивчають типові моделі поведінки користувачів і програм протягом певного часу, вони помічають, коли відбувається щось неочікуване. Інструменти UEBA часто надають пріоритет сповіщенням – ранжуючи рівень ризику, щоб IT-персонал та співробітники служби безпеки могли вирішити, з чим боротися в першу чергу.

Керування користувачами та об'єктами

Рішення UEBA контролюють дозволи користувачів і визначають, чи суперечить поведінка певного користувача призначеним йому правам. Це допомагає зменшити використання привілейованого доступу, а також може виявити зловмисну інсайдерську діяльність. Рішення UEBA також часто контролюють об'єкти або активи, такі як ноутбуки чи сервери, щоб визначити, чи є їхня поведінка аномальною та чи потребує карантину або вимкнення.

Розширена ідентифікація загроз

Коли інструменти UEBA моніторять системи та виявляють аномалії, вони часто визначають, який саме тип проблеми виникає. До них належать такі загрози, як горизонтальне переміщення та витік даних, а рішення UEBA також можуть повідомити вам, чи є загроза внутрішньою чи зовнішньою по відношенню до вашої організації. Ця інформація корисна для боротьби зі зловмисниками, особливо якщо це ваші власні співробітники.

Основна функція UEBA включає:

– Поведінковий аналіз: Системи UEBA аналізують поведінку користувачів та сутностей, включаючи такі дії, як спроби входу, доступ до файлів, мережевий трафік та використання програм.

– Виявлення аномалій: встановлює базовий рівень нормальної поведінки для кожного користувача та сутності, а потім позначає відхилення від цього базового рівня як потенційні загрози.

– Машинне навчання: UEBA використовує алгоритми машинного навчання для виявлення тонких закономірностей та аномалій, які можуть бути пропущені традиційними методами безпеки.

Ключові переваги включають:

– Покращене виявлення загроз: UEBA може виявляти ширший спектр загроз, включаючи внутрішні загрози, скомпрометовані облікові записи та розширені постійні загрози (APT).

– Покращений рівень безпеки: Завдяки проактивному виявленню підозрілої активності, UEBA допомагає організаціям покращити загальний рівень безпеки та зменшити ризик витоків даних.

– Зменшення кількості хибних спрацьовувань: здатність UEBA аналізувати контекст і поведінку допомагає мінімізувати кількість хибних спрацьовувань, дозволяючи командам безпеки зосередитися на справжніх загрозах.

– Швидше реагування на інциденти: Завдяки швидкому виявленню та сповіщенню про аномальну поведінку, UEBA забезпечує швидше реагування на інциденти та їх локалізацію.

– Повна видимість: UEBA надає повне уявлення про активність користувачів та організацій, допомагаючи організаціям зрозуміти, як використовуються їхні системи, та виявити потенційні слабкі місця в безпеці.

Інструменти UEBA використовують інноваційні алгоритми, засновані на традиційному машинному навчанні та глибокому навчанні, для виявлення аномальної та ризикованої поведінки користувачів, машин та інших об'єктів у корпоративній мережі, часто у поєднанні з рішенням для управління інцидентами та подіями безпеки (SIEM).

Зростаюча потреба в UEBA: внутрішні ризики перевищують зовнішні загрози

Згідно з нашим нещодавнім звітом «Від людини до гібрида: як штучний інтелект та розрив в аналітиці підживлюють внутрішні ризики» , внутрішні ризики вже перевершили зовнішні загрози як головну проблему для команд безпеки. У нашому опитуванні 64% фахівців з кібербезпеки визначили зловмисних або скомпрометованих інсайдерів як більшу небезпеку, ніж зовнішніх нападників, порівняно з 36%, які вказали на зовнішніх суб'єктів.

З цих 64% 42% вважали зловмисних інсайдерів основною проблемою, а 22% – скомпрометованих інсайдерів. Понад половина (53%) повідомила, що

кількість інсайдерських інцидентів зросла за останній рік, а 54% очікують, що їхня кількість зростатиме ще більше протягом наступних 12 місяців.

Можливості виявлення залишаються недостатньо розвиненими. Лише 44% організацій використовують аналітику поведінки користувачів та об'єктів (UEBA), яка є критично важливою для виявлення аномальної активності. Хоча 88% кажуть, що мають програму боротьби з внутрішніми загрозами , багато з них є неформальними, недостатньо фінансованими або мають недостатню прозорість у системах. Узгодженість керівництва також є прогалиною: 74% фахівців з безпеки вважають, що керівники недооцінюють внутрішні ризики.

Генеративний штучний інтелект (ШІ) посилює проблему. 76% організацій стикалися з несанкціонованим використанням інструментів GenAI співробітниками. Фішинг та соціальна інженерія за допомогою ШІ (27%), а також несанкціоноване використання GenAI (22%) входять до числа основних векторів внутрішніх загроз, поряд із зловживанням привілеями (18%).

Керівники служб безпеки визнають необхідність кращого аналізу поведінки, але стикаються з технічними та організаційними перешкодами. Опір конфіденційності (20%), відсутність прозорості (16%) та фрагментовані інструменти (10%) створюють сліпі зони у зусиллях з виявлення.

Як працює UEBA

Аналіз поведінки користувачів та сутностей (UEBA) – це категорія рішень або можливостей кібербезпеки, які аналізують поведінку користувачів та сутностей і застосовують розширену аналітику та моделювання поведінки для визначення аномальної поведінки. UEBA використовується для виявлення розширених загроз безпеці, таких як зловмисні інсайдери та компрометація привілейованих облікових записів, які традиційні інструменти безпеки на основі правил не можуть побачити. Рішення UEBA отримують операційні дані з багатьох джерел та визначають, яка нормальна поведінка будь-якого користувача або нелюдської сутності. Суб'єкти можуть включати IT-активи, такі як хости, програми, мережевий трафік, облікові записи служб та сховища даних. З часом

рішення створює стандартні профілі поведінки для користувачів та сутностей у різних групах рівних, щоб створити базовий рівень для того, що є нормальним в організації. Коли виявляється аномальна активність, їй присвоюється оцінка ризику. Оцінка зростає зі збільшенням кількості аномальної поведінки, доки вона не перетне визначений поріг, що спрацьовує для аналітиків безпеки. Деякі рішення можуть автоматизувати дії реагування.

Ось більш детальний огляд основної функції UEBA:

– Машинне навчання: UEBA застосовує методи машинного навчання з вчителем та без вчителя для виявлення ледь помітних аномалій, які статичні правила не можуть вловлювати. Алгоритми можуть адаптуватися до зміни поведінки користувача, зменшуючи потребу в постійних ручних оновленнях. Це дозволяє системі виявляти приховані моделі атак, такі як повільне витікання даних або зловживання привілеями, які розгортаються поступово та в іншому випадку могли б уникнути виявлення.

– Поведінковий аналіз: UEBA збирає та зіставляє дані з кількох джерел, таких як журнали автентифікації, файлові системи, електронна пошта та хмарні додатки, для створення комплексного уявлення про активність. Він відстежує не лише окремі події, а й послідовності дій з плином часу, що дозволяє виявляти незвичайні робочі процеси, спроби доступу або моделі використання, які можуть свідчити про неправильне використання або компрометацію.

– Виявлення аномалій: Після встановлення профілів нормальної поведінки UEBA постійно порівнює нову активність з цими базовими показниками. Відхилення, такі як спроби входу з незвичайних місць, надмірне завантаження файлів або неочікуваний доступ до конфіденційних ресурсів, позначаються. Система призначає контекст цим аномаліям, допомагаючи аналітикам розрізняти нешкідливі відхилення та справжні загрози.

3.2 Розробка структурної схеми

Цілісний аналіз з використанням кількох джерел даних

Справжня сила рішення UEBA полягає в його здатності долати організаційні кордони, IT-системи та джерела даних й аналізувати всі дані, доступні для конкретного користувача чи організації.

Рішення UEBA повинно аналізувати якомога більше джерел даних, деякі приклади джерел даних включають:

– Системи автентифікації, такі як Active Directory.

– Системи доступу, такі як VPN та проксі-сервери.

– Бази даних керування конфігурацією.

– Дані про людські ресурси – нові співробітники, співробітники, що звільнилися, та будь-які дані, що надають додатковий контекст про користувачів.

– Брандмауер, системи виявлення та запобігання вторгненням (IDPS).

– Антивірусні та антивірусні системи.

– Системи виявлення та реагування на кінцеві точки.

– Аналіз мережевого трафіку.

– Стрічки інформації про загрози.

Наприклад, рішення UEBA повинно мати можливість ідентифікувати незвичний вхід через Active Directory, зіставляти його з критичністю пристрою, на який здійснюється вхід, конфіденційністю файлів, до яких здійснювався доступ, та нещодавньою незвичайною мережевою або шкідливою активністю, яка могла призвести до компрометації.

Поведінкове базове дослідження та оцінки ризику

Рішення UEBA вивчає нормальну поведінку, щоб виявити аномальну. Воно аналізує широкий набір даних, щоб визначити базовий або поведінковий профіль користувача.

Наприклад, система відстежує користувача та бачить, як він використовує VPN, о котрій годині приходить на роботу та в які системи входить, який принтер

використовує, як часто та якого розміру файли надсилає електронною поштою або завантажує на USB-накопичувач, а також багато інших даних, що визначають «нормальну поведінку» користувача. Те саме стосується серверів, баз даних або будь-якої значної IT-системи.

Коли відбувається відхилення від базового рівня, система додає до оцінки ризику цього користувача або машини. Чим незвичайніша поведінка, тим вищий бал ризику. Зі збільшенням кількості підозрілих випадків накопичення бал ризику збільшується, доки не досягне певного порогу, що призводить до передачі інформації аналітику для розслідування.

Такий аналітичний підхід має кілька переваг:

– Агрегація – оцінка ризику складається з численних подій, тому аналітикам не потрібно вручну переглядати велику кількість окремих сповіщень та подумки об’єднувати їх для виявлення загрози.

– Зменшення кількості хибнопозитивних результатів – одна незначна аномальна подія сама по собі не призведе до спрацювання сповіщення системи безпеки. Для створення сповіщення системі потрібні кілька ознак аномальної поведінки, що зменшує кількість хибнопозитивних результатів та заощаджує час аналітиків.

Більше контексту – традиційні правила кореляції, визначені адміністраторами безпеки, могли бути правильними для однієї групи користувачів або систем, але не для інших. Наприклад, якщо відділ починає наймати працівників, що працюють позмінно, або працівників, що працюють за кордоном, вони почнуть входити в систему в незвичний час, що постійно призводитиме до спрацьовування сповіщень на основі правил. UEBA розумніша, оскільки встановлює контекстно-залежний базовий рівень для кожної групи користувачів. Вхід працівника, що працює за кордоном, о 3:00 ранку за місцевим часом не вважатиметься аномальною подією.

Аналіз часової шкали та зшивання сесій

Під час аналізу інцидентів безпеки часова шкала є критично важливим поняттям, яке може пов'язати, здавалося б, не пов'язані між собою дії. Сучасні атаки – це процеси, а не ізольовані події.

Передові рішення UEBA можуть «зшивати» дані з різних систем та потоків подій, щоб побудувати повну часову шкалу інциденту безпеки.

Наприклад, розглянемо користувача, який увійшов у систему, виконав підозрілу активність, а потім зник із журналів. Чи була та сама IP-адреса використана для підключення до інших організаційних систем невдовзі після цього? Якщо так, це може бути частиною того самого інциденту, коли той самий користувач продовжував спроби проникнення в систему. Додатковим прикладом може бути вхід зловмисника в систему на одному комп’ютері кілька разів, використовуючи різні облікові дані. Це також вимагає «зшивання» даних про різні спроби входу та позначення їх як одного інциденту.

Після того, як рішення UEBA об'єднає всі відповідні дані, воно може призначити оцінки ризику будь-якій діяльності вздовж часової шкали події. Засвоюється нормальна поведінка для всіх користувачів і машин. Оцінка ризику додається для високоризикової та аномальної поведінки.

Внутрішні загрози

Існує три типи внутрішніх загроз:

1. Недбалий інсайдер – недбалий інсайдер – це працівник або підрядник із привілейованим доступом до IT-систем, який ненавмисно наражає свою організацію на небезпеку, не дотримуючись належних IT-процедур. Наприклад, той, хто залишає свій комп’ютер, не вийшовши з системи, або адміністратор, який не змінив пароль за замовчуванням або не встановив патч безпеки. Визначення нормальної та аномальної активності користувача є ключовим для виявлення користувача, який був скомпрометований через недбалість.

2. Зловмисний інсайдер – Зловмисний інсайдер – це співробітник або підрядник із привілейованим доступом до IT-систем, який має намір здійснити

кібератаку на організацію. Важко виміряти зловмисний намір або виявити його за допомогою лог-файлів або регулярних подій безпеки. Рішення UEBA допомагають, встановлюючи базову лінію типової поведінки користувача та виявляючи аномальну активність.

3. Скомпрометований інсайдер – Зловмисники часто проникають в організацію та компрометують обліковий запис привілейованого користувача або довірений хост у мережі, а потім продовжують атаку звідти. Рішення UEBA можуть допомогти швидко виявити та проаналізувати шкідливу діяльність, яку зловмисник здійснює через скомпрометований обліковий запис.

Традиційним засобам безпеки важко виявити скомпрометованого інсайдера, якщо схема атаки або ланцюжок знищення наразі невідомі (наприклад, під час атаки нульового дня), або якщо атака поширюється латерально через організацію, змінюючи облікові дані, IP-адреси або машини. Однак технологія UEBA може виявляти ці типи атак, оскільки вони майже завжди змушують активи поводитися інакше, ніж встановлені базові показники.

Пріоритетність інцидентів

SIEM збирає події та журнали з кількох інструментів безпеки та критично важливих систем, а також генерує велику кількість сповіщень, які мають розслідувати співробітники служби безпеки. Це призводить до втоми від сповіщень, що є поширеною проблемою Центрів операцій безпеки (SOC).

Рішення UEBA можуть допомогти зрозуміти, які інциденти є особливо ненормальними, підозрілими або потенційно небезпечними в контексті вашої організації. UEBA може вийти за рамки базових показників та моделей загроз, додаючи дані про організаційну структуру, наприклад, критичність активів, ролі та рівні доступу до певних функцій організації. Невелике відхилення від норми для критично захищеної системи або адміністратора вищого рівня може бути вартим уваги для слідчого; для звичайного співробітника лише значне відхилення отримає високий пріоритет.

Запобігання втраті даних (DLP) та запобігання витоку даних

Інструменти запобігання втраті даних (DLP) використовуються для запобігання витоку даних або незаконній передачі даних за межі організації. Традиційні інструменти DLP повідомляють про будь-яку незвичайну активність, що здійснюється з конфіденційними даними, – вони створюють велику кількість сповіщень, з якими може бути важко впоратися командам безпеки.

Рішення UEBA можуть приймати сповіщення DLP, визначати їх пріоритети та консолідувати, розуміючи, які події являють собою аномальну поведінку порівняно з відомими базовими показниками. Це заощаджує час слідчим та допомагає їм швидше виявляти реальні інциденти безпеки.

Аналітика сутностей (IoT)

UEBA може бути особливо важливим у боротьбі з ризиками безпеки Інтернету речей (IoT). Організації розгортають великі парки підключених пристроїв, часто з мінімальними заходами безпеки або без них. Зловмисники можуть скомпрометувати пристрої IoT, використовувати їх для крадіжки даних або отримання доступу до інших IT-систем, або, що ще гірше, використовувати їх для DDoS-атаки чи інших атак на третіх осіб.

Дві чутливі категорії Інтернету речей – це медичні прилади та виробниче обладнання. Підключені медичні прилади можуть містити критично важливі дані та можуть становити загрозу для життя, якщо їх використовувати безпосередньо для догляду за пацієнтами. Виробниче обладнання може спричинити великі фінансові втрати у разі його збою, а в деяких випадках може загрожувати безпеці працівників.

UEBA може відстежувати підключені пристрої, встановлювати базову поведінку для кожного пристрою або групи подібних пристроїв і негайно виявляти, чи пристрій поводиться поза межами своїх звичайних меж. Наприклад:

– Підключення до або з незвичайних адрес чи пристроїв.
– Активність у незвичний час.
– Активовані функції пристрою, які зазвичай не використовуються.

Конвергенція UEBA та SIEM

Існує тісний зв'язок між технологіями UEBA та SIEM, оскільки UEBA спирається на міжорганізаційні дані безпеки для виконання свого аналізу, і ці дані зазвичай збираються та зберігаються SIEM.

Gartner розглядає UEBA як функцію, інтегровану в SIEM. Аналіз поведінки – це одна з можливостей, за допомогою якої Gartner оцінює постачальників у Магічному квадранті для управління інформацією та подіями безпеки. Gartner окреслює такі можливості для SIEM:

– Сукупні дані про події, що генеруються пристроями безпеки, мережевою інфраструктурою, системами та програмами.

– Поєднуйте дані про події з контекстною інформацією про користувачів, активи, загрози та вразливості з метою оцінювання, визначення пріоритетів та пришвидшення розслідувань.

– Нормалізуйте дані для ефективнішого аналізу.

– Пропонуйте аналіз подій у режимі реального часу для моніторингу безпеки, розширений аналіз поведінки користувачів та об'єктів, аналітику запитів, підтримку розслідування та управління інцидентами, а також звітність.

UEBA проти аналогічних технологій

UEBA проти NTA

Аналіз мережевого трафіку (NTA) зосереджений на моніторингу та аналізі мережевого зв'язку для виявлення аномалій або ознак компрометації. Хоча UEBA та NTA виявляють аномальну поведінку, їхні області застосування відрізняються.

UEBA досліджує поведінку користувачів та об'єктів у різних системах, включаючи кінцеві точки, програми та каталоги, а не лише мережеву активність. NTA обмежується мережевими даними та чудово справляється з виявленням таких загроз, як горизонтальне переміщення та витік даних. Натомість UEBA може виявляти загрози, пов'язані з зловживанням привілейованим доступом, незвичайною поведінкою під час входу або змінами в шаблонах доступу до файлів. NTA зазвичай підтримує аналіз трафіку в режимі реального часу, тоді як

UEBA працює як з даними в реальному часі, так і з історичними даними для довгострокового моделювання поведінки.

Разом UEBA та NTA можуть доповнювати один одного: NTA виділяє підозрілі мережеві шляхи, тоді як UEBA надає поведінковий контекст щодо того, хто або що задіяно.

UBA проти UEBA

Аналітика поведінки користувачів (UBA) – це попереднє покоління технологій, орієнтованих виключно на користувачів-людей. Вона аналізує поведінку користувачів для виявлення таких ризиків, як неправомірне використання облікових даних, внутрішні загрози або підозрілі моделі доступу.

UEBA (аналітика поведінки користувачів та об'єктів) розширює цю концепцію, включаючи нелюдські об'єкти, такі як сервери, програми та пристрої Інтернету речей. Цей ширший охоплення є критично важливим, оскільки багато атак спрямовані на об'єкти, що не є користувачами-людьми, або походять від них. UEBA також зазвичай включає більш просунуту аналітику, таку як моделі машинного навчання, здатні виявляти складні поведінкові аномалії в гібридних середовищах.

Коротше кажучи, UEBA базується на UBA, надаючи більш повне уявлення про всі сутності в IT-середовищі та взаємозв'язки між ними.

Методи аналітики UEBA

Деякі рішення UEBA покладаються на традиційні методи виявлення підозрілої активності. До них можуть належати правила, визначені вручну, кореляції між подіями безпеки та відомими шаблонами атак. Обмеження традиційних методів полягає в тому, що вони ефективні лише настільки, наскільки ефективні правила, визначені адміністраторами безпеки, і не можуть адаптуватися до нових типів загроз або поведінки системи.

Розширена аналітика включає кілька сучасних технологій, які можуть допомогти виявити аномальну поведінку навіть за відсутності відомих закономірностей:

– Машинне навчання з вчителем – набори відомої хорошої та відомої поганої поведінки подаються в систему. Інструмент навчається аналізувати нову поведінку та визначати, чи є вона «схожою» на набір відомої хорошої чи відомої поганої поведінки.

– Баєсівські мережі – можуть поєднувати машинне навчання з вчителем та правила для створення поведінкових профілів.

– Самонавчання – система вивчає нормальну поведінку та здатна виявляти аномальну поведінку й попереджати про неї. Вона не зможе визначити, чи є аномальна поведінка хорошою чи поганою, лише те, що вона відхиляється від норми.

– Підсилене/напівавторизоване машинне навчання – гібридна модель, де основою є навчання без вчителя, а фактичні рішення щодо сповіщень передаються назад у систему, щоб забезпечити точне налаштування моделі та зменшити співвідношення сигнал/шум.

– Глибоке навчання – дозволяє проводити віртуальне сортування та розслідування тривог. Система навчається на наборах даних, що представляють тривоги безпеки та їх результати сортування, виконує самоідентифікацію ознак та здатна прогнозувати результати сортування для нових наборів тривог безпеки.

Традиційні методи аналітики є детермінованими в тому сенсі, що якщо певні умови були виконані, генерувалося сповіщення, а якщо ні, система вважала, що «все гаразд». Розширені методи аналітики, перелічені вище, відрізняються тим, що вони є евристичними. Вони обчислюють оцінку ризику, яка є ймовірністю того, що подія являє собою аномалію або інцидент безпеки. Коли оцінка ризику перевищує певний поріг, система створює сповіщення безпеки.

Проблеми розгортання UEBA

Інтеграція та масштабування даних

Однією з головних проблем розгортання UEBA є інтеграція різноманітних джерел даних. Системи UEBA покладаються на комплексні, високоякісні дані із систем керування ідентифікацією, журналів програм, мережевого трафіку,

телеметрії кінцевих точок тощо. Інтеграція цих джерел – часто в різних форматах і обсягах – може бути складною та трудомісткою.

Масштабованість – ще одна проблема. Зі зростанням організацій та додаванням нових пристроїв, програм і користувачів обсяг даних зростає експоненціально. Рішення UEBA повинні обробляти ці дані майже в режимі реального часу, зберігаючи при цьому продуктивність. Без належного планування вузькі місця в продуктивності та збільшена затримка можуть погіршити можливості виявлення та робочі процеси аналітиків.

Хибнопозитивні результати

Незважаючи на розширену аналітику, хибнопозитивні результати залишаються значною проблемою в розгортанні UEBA. Якщо система генерує забагато сповіщень про нешкідливі аномалії, такі як робота легітимного користувача з нового місця розташування, аналітики безпеки можуть бути перевантажені або втратити чутливість.

Ця проблема часто пов'язана з незрілим базовим підходом або недостатнім контекстом у моделях поведінки. З часом, коли система навчається та налаштовує оцінку ризиків, кількість хибнопозитивних результатів може зменшитися. Однак на ранніх етапах розгортання або в динамічних середовищах підтримувати прийнятну якість сповіщень може бути складно.

Вимоги до навичок та ресурсів

Платформи UEBA потребують кваліфікованого персоналу для конфігурації, налаштування та обслуговування. Організаціям потрібні аналітики зі знаннями поведінкової аналітики, виявлення загроз та реагування на інциденти. Крім того, можуть знадобитися інженери обробки даних, щоб забезпечити належне отримання та нормалізацію даних. Меншим організаціям може бракувати досвіду або кількості персоналу для підтримки повномасштабного впровадження UEBA. Навіть для великих підприємств інтеграція UEBA в існуючі операції безпеки може вимагати значних часових витрат та постійних зусиль для підтримки точності та ефективності моделей.

Ключові найкращі практики впровадження UEBA

1. Забезпечення комплексної та високоякісної інтеграції даних

Системи UEBA покладаються на багаті, різноманітні дані для точного моделювання поведінки. Почніть з визначення ключових джерел даних від постачальників ідентифікації (наприклад, Active Directory, LDAP), журналів кінцевих точок, хмарних додатків, VPN, проксі-серверів та мережевого трафіку. Отримуйте як структуровані, так і неструктуровані дані для створення повних профілів поведінки.

Використовуйте конектори, API або відправники журналів для автоматизації збору даних та забезпечення узгодженої синхронізації часу між джерелами – розбіжності в часі можуть перешкоджати точному аналізу часової шкали. Інвестуйте в процеси нормалізації та збагачення даних, щоб стандартизувати формати, вирішувати неоднозначності та позначати відповідні метадані, такі як ролі користувачів або класифікації активів.

Високоякісні дані – це не просто технічна вимога, вони є основоположними для здатності UEBA генерувати змістовні та практичні висновки. Низька якість даних призводить до спотворення базових показників, неефективного виявлення аномалій та збільшення кількості хибнопозитивних результатів.

2. Встановлення надійних базових показників поведінки

Ефективність UEBA залежить від сили її поведінкових моделей. Почніть з надання системі періоду спостереження – зазвичай кілька тижнів – протягом якого вона відстежує активність без попереджень. Протягом цього періоду система встановлює базові показники для користувачів та об'єктів, вивчаючи моделі використання, час доступу, мережеву взаємодію та поведінку системи.

Для більшої точності базові показники повинні враховувати взаємодію з колегами, включаючи порівняння поведінки користувачів, які виконують схожі ролі або мають схожі системи в одній операційній категорії. Врахування організаційного контексту, такого як відділ або місцезнаходження, допомагає

адаптувати базові показники та запобігти неправильній класифікації. Регулярно переглядайте та вдосконалюйте ці базові показники. Якщо бізнес-операції змінюються, наприклад, додавання віддалених команд або сезонний сплеск активності, переконайтеся, що система адаптується. Статичні базові показники в динамічному середовищі призводять до появи сліпих зон або втоми від оповіщень.

3. Ретельно налаштуйте порогові значення та оцінку ризику

Не всі аномалії заслуговують на однакову стурбованість. Системи UEBA використовують оцінки ризику для оцінки серйозності відхилень, але їх необхідно ретельно калібрувати. Почніть з консервативних порогових значень, щоб уникнути перевантаження аналітиків, і коригуйте їх на основі оперативного зворотного зв'язку та аналізу інцидентів.

Оцінка ризику повинна враховувати частоту аномалій, їх серйозність та критичність для ураженої системи чи користувача. Наприклад, незвичайний вхід адміністратора на цінний сервер повинен мати більшу вагу, ніж така сама дія на звичайній робочій станції.

Використовуйте динамічні порогові значення, де це можливо – адаптивні системи, які з часом вивчають прийнятну дисперсію, можуть забезпечити більш нюансовані сповіщення. Також визначте шляхи ескалації та автоматизуйте дії реагування на події з високою достовірністю та високим ризиком, щоб пришвидшити пом’якшення наслідків.

4. Збагачуйте сповіщення контекстом та інформацією про загрози

Контекст є важливим для скорочення часу сортування та покращення прийняття рішень. Збагачуйте сповіщення метаданими з HR-систем (наприклад, статус зайнятості, відділ), інвентаризації активів (наприклад, критичність системи) та історичних моделей поведінки. Включайте такі деталі, як час входу, ідентифікатори пристроїв, геолокацію та журнали доступу до даних.

Рисунок 3.1 – Структурна схема системи.

Інтегруйте канали аналітики загроз для зіставлення аномалій з відомими індикаторами компрометації (IOC) або тактиками, методами та процедурами зловмисників (TTP). Це допомагає розрізняти випадкові аномалії та цілеспрямовані загрози.

Представляйте збагачені сповіщення у зручному для аналітиків форматі, пов’язуючи їх із відповідними подіями в ланцюжку атаки. Це мінімізує час ручного розслідування та покращує якість дій реагування.

5. Інтеграція відповідно до стеку безпеки та робочих процесів

Щоб максимізувати цінність, UEBA має працювати в рамках ширшої екосистеми безпеки. Інтегруватися з платформами SIEM для використання існуючих можливостей збору та кореляції журналів. Передавати оцінки ризиків та сповіщення в системи SOAR, щоб забезпечити автоматизоване виконання сценаріїв, таких як ізоляція пристроїв або скидання облікових даних.

Забезпечте відповідність результатів UEBA вашим існуючим робочим процесам реагування на інциденти, системам видачі заявок та панелям звітності. Це забезпечує безперешкодну передачу між групами виявлення та розслідування та уникає дублювання зусиль.

Ретельно протестуйте інтеграції – системи UEBA повинні не лише надавати точні сповіщення, а й відповідати операційним реаліям. Обсяг сповіщень, час реагування та зручність використання так само важливі, як і точність виявлення. Прагніть до тісно пов’язаної архітектури, де поведінкова аналітика стає природною частиною життєвого циклу ваших операцій безпеки.

3.3 Розробка функціональної схеми

Функціональна схема містить інформацію про способи реалізації пристроєм заданих функцій. За такою схемою можна визначити, як здійснюються перетворення і які для цього необхідні функціональні елементи. Дана схема розробляється на основі структурної схеми для кожного блоку, в результаті з окремих функціональних елементів складається загальна функціональна схема програмного забезпечення.

ПЗ, що підлягає розробці, має забезпечувати виконання всіх функцій, що закладені в системі.

Таким чином, визначивши структуру майбутньої системи та функції, які будуть виконувати її складові, маємо всі необхідні дані для побудови функціональної схеми системи, яка представлена на рисунку 3.2.

Рисунок 3.2 – Функціональна схема системи

Серед блоків схеми перераховані функції, які виконує програма, що гарно ілюструє її функціональність.

До складу головної програми входить чотири основні блоки системи:

– блок головної сторінки системи поведінкового аналізу користувачів за допомогою концепції UEBA;

– блок сторінки профілю користувача;

– блок сторінки запитань для поведінкового аналізу користувачів за допомогою концепції UEBA;

– блок бази даних системи поведінкового аналізу користувачів за допомогою концепції UEBA.

Блок головної сторінки системи поведінкового аналізу користувачів за допомогою концепції UEBA забезпечує ознайомлення користувача з функціоналом веб-сервісу.

Блок головної сторінки системи поведінкового аналізу користувачів за допомогою концепції UEBA містить в собі такі модулі:

– модуль реєстрації користувачів;

– модуль авторизації користувача;

– модуль, який містить нові запитання;

– модуль інформації про розробника.

Модуль реєстрації користувачів має забезпечувати можливість зареєструватись. На сторінці має відображатись форма для введення email та паролю. Після того як користувач введе і вони пройдуть перевірку на коректність, інформація користувача буде збережена в БД.

Модуль авторизації користувачів повинен здійснювати процедуру встановлення належності користувачеві інформації в системі залежно від логіна і пароля користувача, а також керування рівнями та засобами доступу до веб-сервісу. Також необхідно реалізувати валідацію введених в форму даних.

Модуль інформації про розробника забезпечує виведення даних про розробника, а також форму зворотного зв’язку. Форма зворотного зв’язку потрібна, користувачі могли звернутись з своїми пропозиціями і скаргами до розробників.

Модуль нових запитань здійснює відображення останніх доданих запитань.

Блок сторінки профілю користувача надає можливість перегляду активності користувача, а саме: інформація про користувача, запитання, відповіді та коментарі, які він задавав, час коли користувач останній раз з'являвся на сайті та теги.

Також має бути реалізована можливість редагувати основну інформацію на сторінці свого профілю та функція видалення свого профілю.

Блок сторінки запитань складається з таких модулів:

– модуль сторінки окремого запитання;

– модуль виведення списку тегів.

Модуль сторінки окремого запитання виконує такі функції: додавання і перегляд коментарів, перегляд повного вмісту запитання, можливість збільшення рейтингу відповіді, можливість відмічення коментарю як відповіді.

Модуль списку тегів: виведення списку тегів, пошук по тегам.

Користувацький інтерфейс повинен бути розроблений на дуже простому рівні та бути інтуїтивно зрозумілим, що спростить використання розробленого програмного забезпечення.

Розглянувши усі блоки функціональної схеми перейдемо до розгляду діаграми взаємодії процесів, які відбуваються у системі.

3.4 Розробка діаграми процесів

Діаграма процесів розробленої системи зображена на рисунку 3.3. При детальному її розгляді можна побачити як саме проходе взаємодія у розробленій системі. Використовується модель проектування, графічне представлення «потоків» даних в інформаційній системі.

Діаграма взаємодії процесів використовується для візуалізації процесів обробки даних (структурне проектування). Для розробника вважається звичним спочатку креслити діаграму взаємодії процесів даних рівня контексту, завдяки чому буде показано взаємодію системи. Ця діаграма в подальшому підлягає

уточненню шляхом деталізації процесів та потоків даних з метою показати систему що розробляється.

Діаграми потоків даних містять чотири типи елементів:

– Процеси які являють собою трансформацію даних в рамках описуваної системи.

– Сховища даних (репозиторії).

– Зовнішні по відношенню до системи сутності.

– Потоки даних між елементами трьох попередніх типів.

Рисунок 3.3 – Діаграма взаємодії процесів

Таким чином, розглянувши опис системи, структурну, функціональну схеми системи, та діаграму взаємодії процесів перейдемо до опису блок-схем основної програми, та підпрограм, які використовуються, для реалізації системи.

4 РЕАЛІЗАЦІЯ ПРОЕКТУ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ПРАВИЛЬНІСТЬ ПРОЕКТНИХ РІШЕНЬ

4.1 Блок-схеми та опис алгоритмів функціонування системи

Під час роботи над бакалаврською дипломною роботою було створено блок-схеми. Перед їх розглядом необхідно провести роз’яснення який саме тип блок-схем використовується.

Блок-схема це представлення задачі для її аналізу або розв'язування за допомогою спеціальних символів (геометричних образів), які позначають такі елементи, як операції, потік, дані тощо. Блок вхідних та вихідних даних прийнято позначати паралелограмом, блок обчислень (обробки) даних – прямокутником, блок прийняття рішень – ромбом, еліпсом – початок та кінець алгоритму.

У інформаційних технологіях функціональна схема складається з функціональних блоків, які являють собою конструктивно відособлені частини (елементи або пристрої) автоматичних систем, які виконують певні функції. Функціональні блоки на схемі позначають прямокутниками, всередині яких надписують їх найменування відповідно до функцій, що виконуються. Зв'язки між функціональними блоками (внутрішні впливи) позначаються лініями зі стрілками, які вказують напрям впливів.

Функціональні схеми можуть виконуватися в укрупненому і розгорненому вигляді. У першому випадку на схемі зображають найважливіші блоки системи і зв'язки між ними.

У другому варіанті схема відображається більш детально, що полегшує її читання та ілюструє принцип роботи.

Основні елементи схем алгоритму це термінатор, процес, рішення, зумовлений процес (підпрограма), дані та з'єднувач.

Термінатор це елемент відображає вхід із зовнішнього середовища або вихід з неї (найчастіше застосування – початок і кінець програми). Всередині фігури записується відповідна дія.

Процес це виконання однієї або кількох операцій, обробка даних будь-якого виду (зміна значення даних, форми подання, розташування). Всередині фігури записують безпосередньо самі операції.

Рішення це показує рішення або функцію перемикального типу з одним входом і двома або більше альтернативними виходами, з яких тільки один може бути обраний після обчислення умов, визначених всередині цього елементу. Вхід в елемент позначається лінією, що входить зазвичай у верхню вершину елементу. Якщо виходів два чи три то зазвичай кожен вихід позначається лінією, що виходить з решти вершин (бічних і нижній). Якщо виходів більше трьох, то їх слід показувати однією лінією, що виходить з вершини (частіше нижній) елемента, яка потім розгалужується. Відповідні результати обчислень можуть записуватися поруч з лініями, що відображають ці шляхи.

Зумовлений процес (підпрограма) це символ відображає виконання процесу, що складається з однієї або кількох операцій, що визначені в іншому місці програми (у підпрограмі, модулі). Всередині символу записується назва процесу і передані в нього дані.

Дані це перетворення у форму, придатну для обробки (введення) або відображення результатів обробки (виведення). Цей символ не визначає носія даних (для вказівки типу носія даних використовуються специфічні символи).

З'єднувач це символ відображає вихід в частину схеми і вхід з іншої частини цієї схеми. Використовується для обриву лінії та продовження її в іншому місці (приклад: поділ блок-схеми, що не поміщається на листі). Відповідні сполучні символи повинні мати одне (при тому унікальне) позначення.

Також при розробці бакалаврської дипломної роботи було використано наступні підходи UML: діаграма діяльності (діаграми поведінки типу); діаграма прецедентів (діаграми поведінки типу).

Діаграма діяльності. Це візуальне представлення графу діяльностей. Граф діяльностей є різновидом графу станів скінченного автомату, вершинами якого є певні дії, а переходи відбуваються по завершенню дій. Дія є фундаментальною одиницею визначення поведінки в специфікації. Дія отримує множину вхідних сигналів, та перетворює їх на множину вихідних сигналів.

Одна із цих множин, або обидві водночас, можуть бути порожніми. Виконання дії відповідає виконанню окремої дії. Подібно до цього, виконання діяльності є виконанням окремої діяльності, буквально, включно із виконанням тих дій, що містяться в діяльності. Кожна дія в діяльності може виконуватись один, два, або більше разів під час одного виконання діяльності. Щонайменше, дії мають отримувати дані, перетворювати їх та тестувати, деякі дії можуть вимагати певної послідовності.

Специфікація діяльності (на вищих рівнях сумісності) може дозволяти виконання декількох (логічних) потоків, та існування механізмів синхронізації для гарантування виконання дій у правильному порядку.

Діаграма прецедентів це діаграма, на якій зображено відношення між акторами та прецедентами в системі. Також, перекладається як діаграма варіантів використання.

Діаграма прецедентів є графом, що складається з множини акторів, прецедентів (варіантів використання) обмежених границею системи (прямокутник), асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами. Діаграми прецедентів відображають елементи моделі варіантів використання.

Суть даної діаграми полягає в наступному: проектована система представляється у вигляді безлічі сутностей чи акторів, що взаємодіють із системою за допомогою так званих варіантів використання. Варіант використання (use case) використовують для описання послуг, які система надає актору. Іншими словами, кожен варіант використання визначає деякий набір дій, який виконує система при діалозі з актором.

При цьому нічого не говориться про те, яким чином буде реалізована взаємодія акторів із системою.

У мові UML є кілька стандартних видів відношень між акторами і варіантами використання:

– асоціації (association relationship);

– включення (include relationship);

– розширення (extend relationship);

– узагальнення (generalization relationship).

При цьому загальні властивості варіантів використання можуть бути представлені трьома різними способами, а саме – за допомогою відношень включення, розширення і узагальнення.

Відношення асоціації – одне з фундаментальних понять у мові UML і в тій чи іншій мірі використовується при побудові всіх графічних моделей систем у формі канонічних діаграм.

Включення (include) у мові UML – це різновид відношення залежності між базовим варіантом використання і його спеціальним випадком. При цьому відношенням залежності (dependency) є таке відношення між двома елементами моделі, при якому зміна одного елемента (незалежного) приводить до зміни іншого елемента (залежного).

Відношення розширення (extend) визначає взаємозв'язок базового варіанта використання з іншим варіантом використання, функціональна поведінка якого задіюється базовим не завжди, а тільки при виконанні додаткових умов.

Блок-схеми є першоджерелами стратегії розвитку ПЗ. Тому від точності і детальної блок-схеми залежить результат всієї програми.

При виборі початкової точки відліку при побудові схем було враховано, що виходячи з вибору мови програмування і інших технічних засобів, програма буде об’єктно-орієнтована що вимагає оптимізації програми високого рівня, також те, що при розробці програми слід надати особливу увагу модулю поведінкового аналізу користувачів за допомогою концепції UEBA.

На рисунку 4.1 зображена основна блок-схема програми, на рисунку 4.2 зображено роботу підпрограми.

З яких видно що робота основної програми складається з початкових етапів ініціалізації ПЗ, перевірки наявності ресурсів системи, блоку початку основного циклу з чеканням запиту від користувача в якому відбувається виклик підпрограми та останньої стадії – перевірка поточного стану з завершенням роботи розробленого ПЗ. При роботі підпрограми виконується основний функціонал системи з циклічними послідовностями, перевіркою поточного стану та поверненням в основну програму прапорів стану виконання.

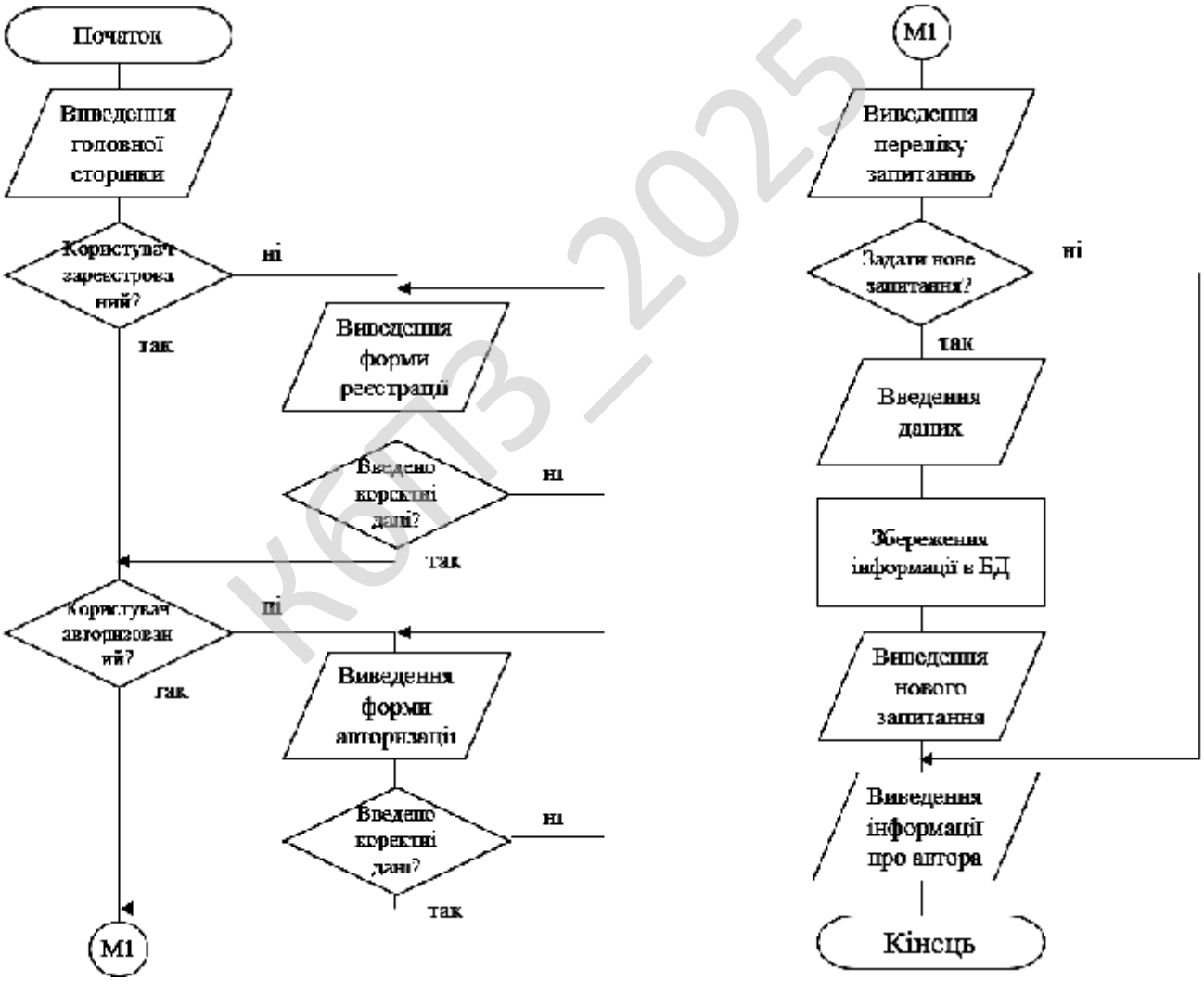

Рисунок 4.1 – Блок-схема основної програми

Подійно-орієнтована архітектура (Event-driven architecture, надалі EDA) – шаблон архітектури програмного забезпечення, який призначений для створення подій, їх виявлення, споживання і реагування на них.

Подія може бути визначена як значна зміна стану. Наприклад, коли споживач купує автомобіль, стан автомобіля змінюється з "на продаж" до "продано". Архітектура системи дилера автомобілів може трактувати цю зміну стану як подію, поява якої може стати відомою іншим програмам даної архітектури.

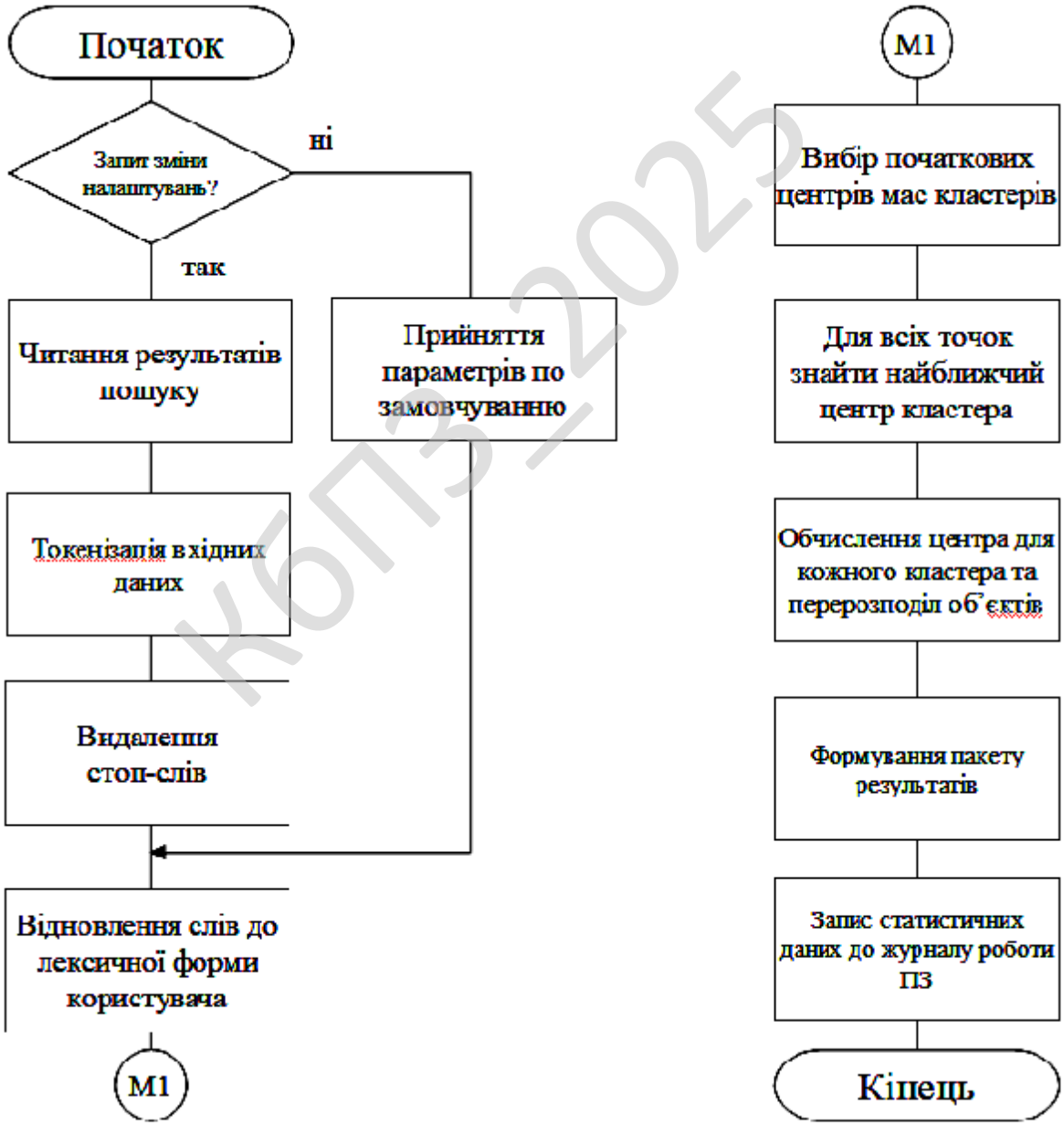

Рисунок 4.2 – Блок-схема роботи підпрограми

З формальної точки зору, те, що виробляється, публікується, поширюється, виявляється і споживається (як правило, асинхронно) є повідомленням, яке називають сповіщенням про подію (або нотифікацією), а не самою подією, яка є зміною стану, що викликає появу повідомлення.

Події не подорожують, вони просто відбуваються. Проте термін подія часто використовується метонімічно для позначення самого нотифікаційного повідомлення, що може призвести до певної плутанини.

Цей архітектурний шаблон може застосовуватися при проектуванні і реалізації ПЗ і систем, які передають події між слабкозв'язаними компонентами програмного забезпечення і сервісами (службами).

Подійно-орієнтована система як правило складається з емітерів подій (або агентів) і споживачів подій (або стоків).

Стоки несуть відповідальність за здійснення реагування на появу події. Реакція не завжди може бути повністю забезпечена самим стоком. Наприклад, стік, може бути відповідальним лише за фільтрацію, трансформацію і відправку події до іншого компонента або він може забезпечити повністю самостійну реакцію на таку подію. Перша категорія стоків може бути заснована на традиційних компонентах, таких як проміжне програмне забезпечення, орієнтоване на обробку повідомлень (message oriented middleware, MOM), в той час, як друга категорія стоків (самостійна реакція в режимі он-лайн) може вимагати більш придатної платформи (фреймворку) для виконання транзакцій.

Розробка ПЗ і систем в подійно-орієнтованій архітектурі дозволяє їм бути сконструйованими способом, який більш відповідає вимогам до їх створення, оскільки такі системи в більшій мірі пристосовуються до непередбачуваних і асинхронних середовищ.

Подійно-орієнтована архітектура (EDA) може доповнювати сервісно-орієнтовану архітектуру (SOA), оскільки сервіси (служби) можуть бути активовані тригерами, які ініціюються при настанні подій.

Ця парадигма особливо корисна, коли стік не забезпечує власного виконання будь-яких дій.

Подійно-орієнтована SOA (SOA-2) розвиває архітектури SOA і EDA для забезпечення більш глибокого і надійного рівня сервісів за рахунок використання раніше невідомих причинно-наслідкових зв'язків, щоб сформувати новий шаблон подій. Цей новий шаблон бізнес-аналітики дає поштовх до небаченого раніше зростання рівня автоматизації підприємства за рахунок привнесення додаткової цінної інформації в описану раніше модель діяльності.

Обчислювальна техніка та сенсорні пристрої (сенсори, датчики, контролери) можуть виявляти зміни стану об'єктів або умов і створювати події, які потім можуть бути оброблені сервісом (службою) або системою.

Подія може складатися з двох частин: заголовка події та тіла події. Заголовок події може включати в себе інформацію таку як, наприклад, назва події, часова мітка події і тип події. Тіло події – це частина, яка описує факт, що стався в дійсності. Тіло події не слід плутати з шаблоном або логікою, яка може бути застосована як реакція на саму події.

Архітектура, керована подіями, складається з чотирьох логічних рівнів (шарів). Вона починається з виявлення факту, його технічного подання у формі події і закінчується непустою множиною реакцій на цю подію.

Генератор подій

Першим логічним шаром є генератор подій, який виявляє факт і представляє цей факт подією. Оскільки фактом може бути практично все, що може бути сприйнято, то ним може бути і генератор подій. Наприклад, генератором може бути клієнт електронної пошти, система електронної комерції або певний тип датчика.

Перетворення різних даних, отриманих від датчиків, в єдину стандартизовану форму даних, які можуть бути оцінені, є основною проблемою при розробці та реалізації цього шару. Однак, враховуючи, що подія є строго

декларативною, можна легко застосовувати будь-які операції трансформації, тим самим усуваючи необхідність забезпечення високого рівня стандартизації.

Канал подій

Канал подій – це механізм, через який інформація від генератора подій передається до обробника подій (event engine) або стоку.

Це може бути з'єднання TCP/IP або вхідний файл будь-якого типу (простий текст, формат XML, e-mail тощо). В один і той же час може бути відкрито кілька каналів подій. Як правило, оскільки обробник подій повинен працювати в режимі, наближеному до реального часу, канали подій зчитуються асинхронно. Події зберігаються в черзі, очікуючи наступної обробки механізмом обробки подій.

Механізм обробки подій

Механізм обробки подій (event processing engine) є місцем, де подія ідентифікується і вибирається відповідна реакція на нього, яка потім виконується. Це також може призвести до породження ряду тверджень. Якщо подія, яка надійшла до механізму обробки подій, є наприклад такою «Запаси продукту ID досягли нижнього допустимого рівня», це може ініціювати, наприклад, такі реакції як «Замовити продукт ID» і «Сповістити персонал».

Наступна подійно-орієнтована дія (післядія)

Щодо того, як можуть проявлятися наслідки події, слід відмітити, що вони можуть проявитись багатьма різними способами і у різноманітних формах (наприклад, повідомлення електронної пошти, надіслане комусь, або ПЗ, що виводить деяке попередження на екран). Залежно від рівня автоматизації, який забезпечується стоком (механізмом обробки подій), ці дії можуть виявитись зайвими.

Є три основні стилі обробки подій: простий, потоковий і складний. Часто ці три стилі використовуються спільно у розвинутій подійно-орієнтованій архітектурі.

Проста обробка подій

Проста обробка подій стосується подій, які безпосередньо належать до специфічних вимірних змін умов. У випадку простої обробки подій, мають справу з появою відомих подій, що ініціюють післядію (післядії). Проста обробка подій зазвичай використовується для управління потоком робіт в реальному часі, скорочуючи тим самим час затримки і вартість робіт.

Наприклад, прості події можуть створюватись (породжуватись) датчиком, що виявляє зміну тиску в шині або температуру навколишнього середовища.

Обробка потоку подій

При обробці потоку подій (event stream processing, далі ESP) відбуваються як звичайні, так і відомі події. Звичайні події (заявки, передачі RFID) перевіряються на те, чи є вони відомими, і передаються інформаційним передплатникам. Обробка потоку подій зазвичай використовується для управління потоком інформації в реальному часі і на рівні підприємства, що дозволяє своєчасно приймати рішення.

Обробка складних подій

Обробка складних подій (Complex event processing (CEP)) дозволяє за шаблонами простих і звичайних подій проводити аналіз того, чи наступила складна подія. Обробка складних подій полягає в оцінюванні взаємного впливу подій і в наступному виконанні дій. При цьому, типи подій (відомих або звичайних) можуть перетинатись, а події можуть виникати протягом тривалого періоду часу.

Кореляція подій може бути причинною, тимчасовою або просторовою. CEP вимагає використання складних інтерпретаторів подій, визначення і підбору шаблонів подій, а також відповідних кореляційних методів. Обробка складних подій зазвичай використовується для виявлення і реагування на аномальну поведінку, загрози і можливості у бізнесі.

4.2 Захист розробленого програмного забезпечення

Для захисту розробленого програмного забезпечення запропоновано використовувати алгоритм SEED – у криптографії симетричний блоковий криптоалгоритм на основі Мережі Фейстеля, розроблений Корейським агентством інформаційної безпеки (Korean Information Security Agency, KISA) в 1998 році. В алгоритмі використовується 128-бітний блок і ключ довжиною 128 біт. Алгоритм одержав широке поширення й використовується фінансовими й банківськими структурами, виробничими підприємствами й бюджетними установами Південної Кореї, оскільки 40-бітний SSL не забезпечує на даний момент мінімально необхідного рівня безпеки. Агентством по захисту інформації специфіковане використання шифру SEED у протоколах TLS і S/MIME. У той же час, алгоритм SEED не реалізований у більшості сучасних браузерів і інтернет-додатків, що утрудняє його використання в даній сфері поза межами Південної Кореї. SEED являє собою мережу Фейстеля з 16 раундами, 128-бітовими блоками й 128-бітовим ключем. Алгоритм використовує дві 8×8 таблиці підстановки, які, як такі з Safer, виведені з дискретного зведення в ступінь (у цьому випадку, x^{247} і x^{251} – плюс деякі «несумісні операції»). Це є деякою подібністю с MISTY1 у рекурсивності його структури: 128-бітовий повний шифр – мережа Фейстеля з F-функцією, що впливає на 64-бітові половини, у той час як сама F-функція – Мережа Фейстеля, складена з G-функції, що впливає на 32-розрядні половини. Однак рекурсія не простягнеться далі, тому що G-функція – не Мережа Фейстеля. В G-функції 32-розрядне слово розглядають як чотири 8-бітових байта, кожний з яких проходить через одну або іншу таблицю підстановки, потім поєднується в помірковано комплексному наборі булевих функцій таким чином, що кожний біт виводу залежить від 3 з 4 вхідних байтів. SEED має складний ключовий розклад, генеруючи тридцять два 32-розрядних додаткових символу, використовуючи G-функції на серіях обертань вихідного неопрацьованого ключа, комбінованого зі спеціальними раундовими константами (як в TEA) від «Золотого співвідношення» (англ. Golden ratio).

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

Розглянемо розроблене ПЗ поведінкового аналізу користувачів за допомогою концепції UEBA яке зображено на рисунку 5.1. З рисунку можна побачити що інтерфейс програми.

Розроблена програма має дуже простий і інтуїтивно зрозумілий інтерфейс з користувачем. Кожен, хто в достатньому обсязі володіє операційним середовищем Windows без особливих складностей освоїть і цю програму, оскільки її інтерфейс інтуїтивно зрозумілий.

Якщо програма не видала ніяких помилок, і працює, то можна використовувати, інакше слід слідувати інструкціям, які пропонує програма.

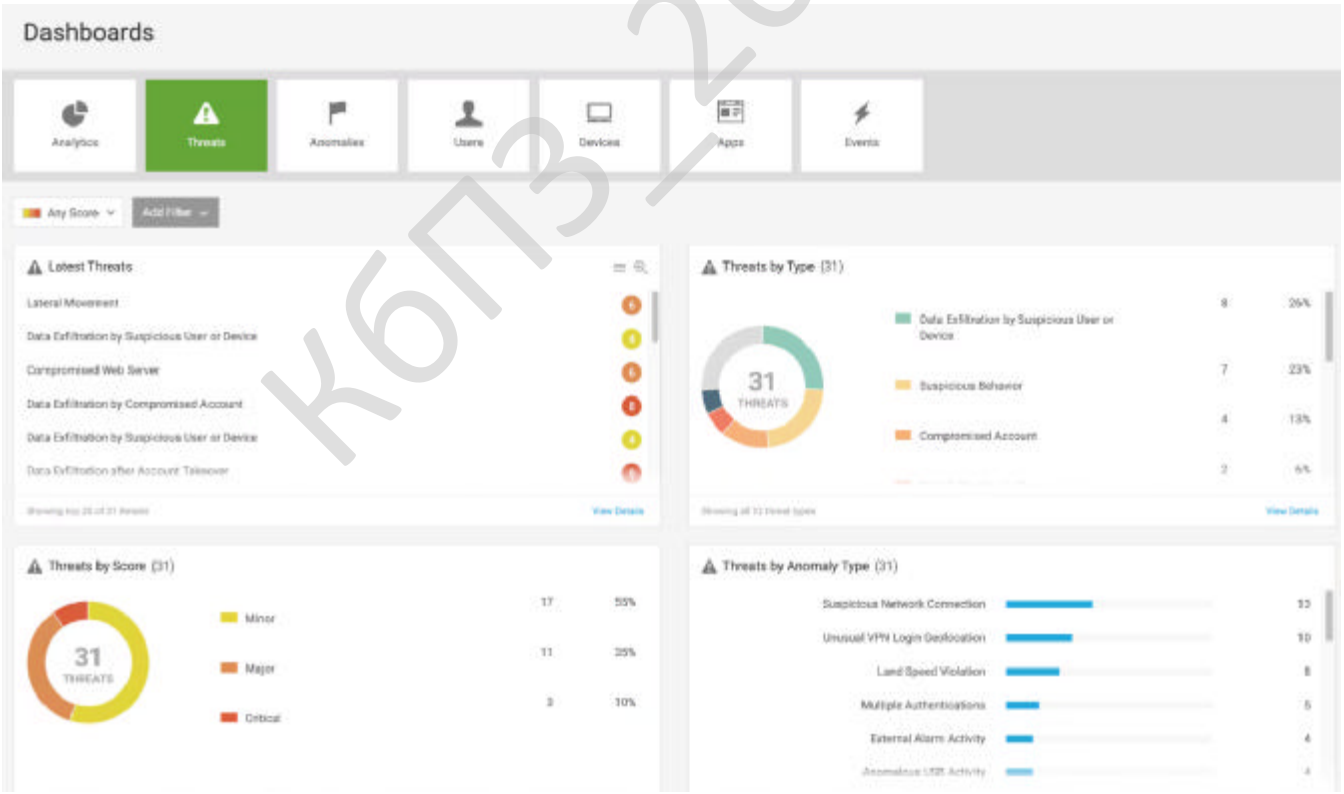

Рисунок 5.1 – Головне вікно ПЗ

На рисунку 5.2 зображено авторські дані розробленого програмного забезпечення.

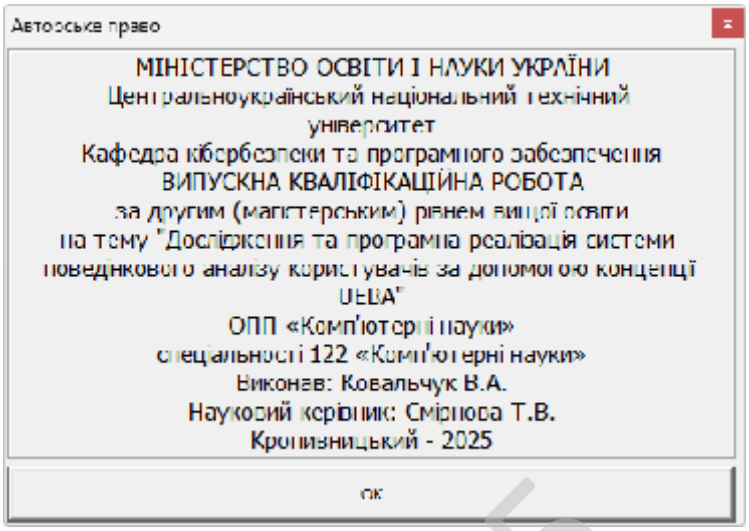

Рисунок 5.2 – Авторське право

Розглянемо процес впровадження програмного забезпечення, це процес налаштування програмного забезпечення під певні умови використання, а також навчання користувачів роботі з програмним продуктом. Впровадження програмного забезпечення це усі дії, що роблять розроблену програмну систему готовою до використання. Даний процес є частинною життєвого циклу програмного забезпечення.

Загалом процес розгортання складається з кількох взаємопов'язаних дій із можливими переходами між ними. Ця активність може відбуватися як з боку виробника так і з боку споживача. Оскільки кожна програмна система є унікальною, то усі процеси та процедури під час розгортання важко передбачити. Тому, "розгортання" можна трактувати як загальний процес відповідно до певних вимог та характеристик. Розгортання може здійснюватись програмістом і в процесі розробки програмного забезпечення.

До діяльностей пов'язаних із розгортанням програмного забезпечення відносять:

– Випуск.

– Встановлення та активація.

– Деактивація.

– Адаптація.

– Обновлення.

– Вмонтування.

– Відстежування версій.

– Видалення.

– Вилучення з обігу.

При впровадженні програмного забезпечення потрібно урахувати наступні дії:

– Виділення критичних, з точки зору загального результату, процедур в діяльності організації. Коли набір таких процедур визначений, необхідно в першу чергу використовувати IT рішення для автоматизації операцій усередині саме цих процедур. Таким чином, розроблене IT рішення автоматично стає життєво важливим і затребуваним для організації, а також буде забезпечена публічність процесу впровадження;

– Розширення нормативної бази організації шляхом включення до неї регламентів, що описують порядок виконання процедур автоматизованих процесів. В іншому випадку є небезпека виникнення неузгодженості між автоматизованими процедурами та іншими процесами організації.

– Виконання робіт з загальної стандартизації існуючої діяльності організації, коли виділяються кращі практики виконання процедур і включаються в IT рішення за принципом найбільшої корисності для більшості учасників. Відсоток таких процедур щодо загального обсягу автоматизації може бути невеликий, але це надає процесу побудови рішення вагу в організації за рахунок збільшення його необхідності.

Під час роботи над програмою було проведено тестування програмного забезпечення, тобто технічне дослідження, призначене для виявлення інформації про якість продукту відносно контексту, в якому воно має використовуватись.

Тестування включає як процес пошуку помилок або інших дефектів, так і випробування програмних складових з метою їх оцінки.

Проводилась оцінка:

– відповідності поставленим вимогам;

– правильна відповідь для усіх можливих вхідних даних;

– виконання функцій за прийнятний час;

– практичність;

– сумісність з ОС та стороннім ПЗ.

Оскільки число можливих тестів для програмних компонент практично нескінченне, тому стратегія тестування полягала в тому, щоб провести всі можливі тести з урахуванням наявного часу та ресурсів.

Як результат ПЗ тестувалось стандартним виконанням програми з метою виявлення помилок або інших дефектів.

Проводилось тестування форматом білої скриньки засноване на аналізі керуючої структури програми. Програма вважається повністю перевіреною, якщо проведено вичерпне тестування маршрутів (шляхів) її графа управління.

У цьому випадку формуються тестові варіанти, в яких:

– Гарантується перевірка всіх незалежних маршрутів програми.

– Знаходяться гілки True, False для всіх логічних рішень.

– Виконуються всі цикли (у межах їхніх кордонів та діапазонів).

– Аналізується правильність внутрішніх структур даних.

Недоліки тестування "білої скриньки":

– Кількість незалежних маршрутів може бути дуже велика.

– Повне тестування маршрутів не гарантує відповідності програми вихідним вимогам до неї.

– У програмі можуть бути пропущені деякі маршрути.

– Не можна виявити помилки, поява яких залежить від даних.

Переваги тестування "білої скриньки" пов'язані з тим, що принцип «білої скриньки» дозволяє врахувати особливості програмних помилок:

– Кількість помилок мінімально в «центрі» і максимально на «периферії» програми.

– Попередні припущення про ймовірність потоку керування або даних у програмі часто бувають некоректними. У результаті типовим може стати маршрут, модель обчислень за яким опрацьована слабо.

– При записі алгоритму програмного забезпечення у вигляді тексту на мові програмування можливе внесення типових помилок трансляції (синтаксичних та семантичних).

– Деякі результати в програмі залежать не від вихідних даних, а від внутрішніх станів програми.

Проводилось тестування чорної скриньки.

Основне місце програми тестів «чорної скриньки» – інтерфейс ПЗ. Відомі: функції програми. Досліджується: робота кожної функції на всій області визначення.

Ці тести демонструють:

– Як виконуються функції програми.

– Як приймаються вихідні дані.

– Як виробляються результати.

– Як зберігається цілісність зовнішньої інформації.

При тестуванні «чорної скриньки» розглядаються системні характеристики програм, ігнорується їхня внутрішня логічна структура. Вичерпне тестування, як правило, неможливе.

Наприклад, якщо в програмі 10 вхідних величин і кожна приймає по 10 значень, то кількість тестових варіантів становитиме 10^10. Тестування «чорної скриньки» не реагує на багато особливостей програмних помилок.

Тестування «чорної скриньки» (функціональне тестування) дозволяє отримати комбінації вхідних даних, які забезпечують повну перевірку всіх функціональних вимог до програми.

Програмний виріб тут розглядається як «чорна скринька», чию поведінку можна визначити тільки дослідженням його входів та відповідних виходів. При такому підході бажано мати:

– Набір, утворений такими вхідними даними, які призводять до аномалій у поведінці програми (назвемо його ITc).

– Набір, утворений такими вхідними даними, які демонструють дефекти програми (назвемо його OT).

Будь-який спосіб тестування «чорної скриньки» повинен:

– Виявити такі вхідні дані, які з високою ймовірністю належать набору ITc;

– Сформулювати такі очікувані результати, які з високою імовірністю є елементами набору OT.

Принцип «чорної скриньки» не альтернативний принципу «білої скриньки». Скоріше це доповнює підхід, який виявляє інший клас помилок.

Тестування «чорної скриньки» забезпечує пошук наступних категорій помилок:

– Некоректних чи відсутніх функцій.

– Помилок інтерфейсу.

– Помилок у зовнішніх структурах даних або в доступі до зовнішньої бази даних.

– Помилок характеристик (необхідна ємність пам'яті і т.д.).

– Помилок ініціалізації та завершення.

Обрано умови розповсюдження – commercial software. Програмне забезпечення, створене комерційною організацією з метою отримання прибутку від його використання іншими, наприклад, шляхом продажу копій.

Найважливішою особливістю комерційний програмних продуктів є підтримка великих компаній, прямо зацікавлених у поширенні програм. Багато

організацій надають виключно платну підтримку своїх продуктів, такий підхід, як правило, використовують організації надають відкриті вихідні коди. Для продуктів, що розповсюджуються на комерційній основі діють зазвичай безкоштовні служби підтримки, покликані збільшити рівень довіри у клієнтів і потенційних покупців.

Далеко не завжди, але як правило терміни критично важливих змін в комерційних продуктах значно менше, ніж у некомерційних проектів. Це пов'язано з тим, що над комерційним продуктом працюють цілі групи розробників і ця робота є їх основним заняттям. Розробникам-початківцям як правило доводиться шукати додаткові способи заробітку, і це збільшує час, що витрачається на доповнення і зміни програм. Так як основним рушійним фактором створення комерційного ПЗ є одержання прибутку, то комерційні програмні продукти першими заповнюють вільні ніші та пропонують варіанти вирішення завдань відразу по мірі виявлення вакууму в будь-якому секторі ринку.

Окремий вид комерційних програм, коли їх розробка оплачується безпосередньо замовником. Такі програми найчастіше позбавлені всіх переваг комерційних продуктів, оскільки мають обмежений бюджет, але більш адаптовані до вимог замовника, ніж аналоги.

6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи поведінкового аналізу користувачів за допомогою концепції UEBA.

Метою розробки є дослідження та програмна реалізація системи поведінкового аналізу користувачів за допомогою концепції UEBA.

Об'єктом дослідження є процес поведінкового аналізу користувачів за допомогою концепції UEBA.

Предметом дослідження є методи поведінкового аналізу користувачів за допомогою концепції UEBA.

Методи дослідження базуються на методах машинного навчання, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено метод поведінкового аналізу користувачів за допомогою концепції UEBA.

– Розроблено вітчизняний продукт поведінкового аналізу користувачів за допомогою концепції UEBA, який має більш широкі можливості, на відміну від існуючих аналогів.

7 МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ ІТ-ПРОЄКТУ

7.1 Визначення цільової аудиторії кінцевого готового продукту

Результати дослідження та впровадження системи поведінкового аналізу користувачів на основі концепції UEBA насамперед можуть зацікавити компанії, які мають великий обсяг даних і високу кількість користувачів – фінансові установи, банки, телекомунікаційні оператори, освітні заклади та державні органи. У таких організаціях питання безпеки даних стоїть особливо гостро, адже будь-яка підозріла активність користувачів або порушення політики доступу може призвести до серйозних фінансових та репутаційних втрат.

Для ІТ-компаній та аналітичних центрів, що займаються питаннями кіберзахисту, система UEBA стане основою для створення більш адаптивних і “розумних” рішень у сфері інформаційної безпеки. Завдяки можливості аналізу патернів поведінки користувачів у реальному часі такі системи дозволяють не лише фіксувати інциденти, а й передбачати потенційні загрози. Це відкриває широкі перспективи для досліджень і вдосконалення механізмів безпеки.

Крім того, дослідження може бути корисним для освітніх установ, які готують фахівців з кібербезпеки, адже воно демонструє практичне застосування машинного навчання у виявленні аномалій поведінки користувачів. Для державних структур, що відповідають за цифрову трансформацію, UEBA може стати інструментом у боротьбі з кіберзлочинністю, зокрема при моніторингу критичної інфраструктури.

7.2 Оцінка привабливості шляхом застосування методів експертних оцінок

Для визначення привабливості системи UEBA доцільно застосувати метод експертних оцінок, який базується на залученні фахівців із кібербезпеки, IT-аналітиків та керівників безпекових підрозділів підприємств. Кожен експерт оцінює проєкт за критеріями – технологічна інноваційність, потенціал ринку, економічна вигода, складність впровадження, можливість інтеграції з існуючими системами безпеки та рівень автоматизації процесів.

Наприклад, якщо за десятибальною шкалою більшість експертів оцінює інноваційність на 9 балів, а економічну вигоду – на 8, то з урахуванням вагових коефіцієнтів (0,3 для інноваційності, 0,25 для економіки, 0,2 для інтеграції тощо) середній інтегральний показник привабливості може становити понад 8,5 бала з 10. Це свідчитиме про високу перспективність проєкту в ринкових умовах.

Такий підхід дає змогу не лише оцінити рентабельність і конкурентоспроможність продукту, а й виявити ключові фактори ризику, які можуть знизити його привабливість – наприклад, складність адаптації до великих корпоративних систем або потребу в спеціалізованому персоналі.

7.3 Вибір методу оцінки вартості ПЗ

Найбільш доцільним методом оцінки вартості для реалізації системи UEBA є метод дисконтованих грошових потоків (DCF), який враховує очікувані вигоди від впровадження системи в майбутньому. Цей метод дозволяє оцінити, як інвестиції у систему сьогодні вплинуть на економію витрат і запобігання збиткам у наступні роки.

Наприклад, зменшення кількості внутрішніх інцидентів на 70% або скорочення часу реагування на них може прямо впливати на фінансові результати компанії. DCF дозволяє розрахувати теперішню вартість таких переваг з

урахуванням фактору часу, дисконту та ризику. Також доцільно поєднати цей метод із витратним підходом, який визначає первісну вартість створення системи, включно з розробкою, впровадженням, тестуванням і подальшою підтримкою. Це дає змогу створити комплексну модель оцінки, яка враховує як фінансову сторону інвестицій, так і довгострокові переваги, що проявляться в зниженні ризиків і підвищенні стабільності бізнесу.

7.4 Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості

Підприємство, яке має розгалужену корпоративну мережу, стикається з проблемою виявлення інсайдерських загроз, зловживань доступом і нетипової активності користувачів. Традиційні системи контролю доступу (наприклад, SIEM) ефективно фіксують відомі інциденти, але не здатні аналізувати контекст поведінки співробітників у динаміці.

Для вирішення цієї проблеми пропонується впровадження системи UEBA, яка аналізує поведінкові патерни користувачів і виявляє аномалії на основі машинного навчання. Це дає змогу запобігати інцидентам ще до їхнього прояву – наприклад, коли користувач раптово починає завантажувати великі обсяги даних у незвичний час або підключається з нетипового пристрою.

Мета проєкту – зменшити збитки від внутрішніх загроз, скоротити витрати на розслідування інцидентів і підвищити загальний рівень інформаційної безпеки без збільшення чисельності IT-відділу. Вхідні дані зафіксовано в таблиці 7.1.

Таблиця 7.1 – Вихідні дані для розрахунку

Показник	**До впровадження**	**Після впровадження**	**Економічний ефект**
Кількість інсайдерських інцидентів на рік	8	2	–6
Середній збиток від одного інциденту, грн	500 000	200 000	–300 000
Річні збитки від інцидентів, грн	4 000 000	400 000	–3 600 000
Витрати на розслідування інцидентів, грн/рік	1 200 000	600 000	–600 000
Початкові інвестиції у систему (розробка, інтеграція, навчання персоналу)	—	2 000 000	—
Щорічні витрати на підтримку, аналітику та оновлення	—	300 000	—

Розрахунок економічного ефекту демонструє наступне: зниження збитків від внутрішніх загроз – 3 600 000 грн/рік, економія на розслідуванні інцидентів –

600 000 грн/рік, підвищення продуктивності відділу безпеки (економія робочого часу становить близько 400 людино-годин на рік, що еквівалентно 200 000 грн/рік), сукупний річний економічний ефект – 4 400 000 грн/рік, чистий економічний ефект – 4 100 000 грн/рік, термін окупності ≈ 0,49 року (≈ 6 місяців), рентабельність інвестицій – 205 %.

Додаткові нефінансові вигоди: підвищення рівня довіри до корпоративної безпеки: менше інцидентів означає стабільнішу репутацію підприємства, автоматизація аналізу поведінкових ризиків: зменшення навантаження на аналітиків безпеки, своєчасне реагування на підозрілу активність: UEBA виявляє загрози ще до того, як вони переходять у фазу інциденту, покращення політик доступу: система аналізує закономірності користувачів і пропонує оптимізацію рівнів доступу, інтеграція з SIEM: UEBA доповнює наявні системи безпеки, не потребуючи повної заміни існуючих інструментів.

Таким чином, впровадження UEBA є не лише технічно виправданим кроком, а й економічно вигідним рішенням, що поєднує аналітичну потужність штучного інтелекту з бізнес-логікою управління ризиками.

7.5 Пропозиція алгоритму просування проєкту розробки ПЗ

Перший етап просування має передбачати розробку демонстраційної версії продукту, яку можна показати потенційним клієнтам на прикладі реальних сценаріїв виявлення аномалій. Це дозволить продемонструвати переваги UEBA у порівнянні зі звичайними системами контролю доступу. Важливо не просто описати функціональні можливості, а показати, як система запобігає витоку даних і підвищує безпеку без надлишкового навантаження на ІТ-відділ.

Наступним кроком може стати участь у конференціях і форумах з інформаційної безпеки, де можна представити кейси використання системи в корпоративному середовищі. Це допоможе здобути увагу ринку та потенційних

партнерів. Далі потрібно запустити пілотні проєкти у кількох організаціях, які погодяться протестувати систему у своїй мережі.

Завершальним етапом має стати створення маркетингової кампанії з акцентом на практичну вигоду – наприклад, на реальну економію від скорочення кількості інцидентів і підвищення ефективності реагування. У поєднанні з публікацією аналітичних звітів і відгуків клієнтів це створить довіру до продукту та допоможе масштабувати проєкт.

7.6 Оптимізація каналів збуту та шляхів реалізації ПЗ

Для оптимізації збуту можна зосередитися на партнерських програмах із постачальниками рішень у сфері інформаційної безпеки, які вже мають усталені зв'язки з корпоративними клієнтами. Інтеграція UEBA як додаткового модуля до існуючих SIEM або SOC-систем може суттєво спростити процес виходу на ринок і збільшити охоплення цільової аудиторії.

Також можна запропонувати гнучкі моделі ліцензування – наприклад, підписку за схемою "software as a service" (SaaS). Це дозволить зменшити поріг входу для малих і середніх підприємств, які не мають змоги інвестувати значні суми в кіберзахист одразу.

Ще одним напрямом може бути створення спільних проєктів із державними структурами або освітніми установами, що займаються дослідженнями в галузі штучного інтелекту. Це дозволить не лише розширити мережу контактів, а й підвищити репутаційну привабливість продукту як технологічно передового рішення у сфері кібербезпеки.

7.7 Визначення ключових факторів успіху конкретного проєкту

Основним фактором успіху є точність роботи системи – здатність UEBA коректно розпізнавати відхилення у поведінці користувачів без створення великої

кількості хибнопозитивних сповіщень. Якщо система здатна правильно визначати ризики й одночасно не заважати звичним робочим процесам, вона швидко отримає довіру користувачів і аналітиків безпеки.

Не менш важливим чинником є швидкість адаптації та масштабованість системи. Оскільки корпоративні середовища постійно змінюються, UEBA має легко підлаштовуватись під нові умови – додавання користувачів, нові типи доступу, хмарні сервіси. Це забезпечить довготривалу актуальність рішення.

Крім того, успіх визначатиметься рівнем підтримки з боку керівництва компанії, готовністю інвестувати у розвиток системи та навчання персоналу. Сучасний бізнес розуміє, що безпека – це не витрати, а інвестиція в стабільність і довіру клієнтів. Якщо UEBA зможе показати реальні результати у вигляді зменшення інцидентів і підвищення ефективності реагування, вона стане невід'ємним елементом цифрової стратегії будь-якої організації.

8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

8.1 Вступ

Охорона праці – система збереження життя і здоров'я працівників у процесі трудової діяльності, що включає правові, соціально-економічні, організаційні, технічні, санітарно-гігієнічні, лікувально-профілактичні, реабілітаційні та інші заходи.

Згідно закону України “Про охорону праці” [3] кожна компанія впроваджує заходи з охорони праці. Реалізується трудові відносини з вживанням необхідних засобів з охорони праці та розробки відповідних документів: інструкцій з охорони праці по кожній професії і загальні; положення про охорону праці; накази з охорони праці; журнали реєстрації та інструктажу.

Роботодавець створює відділ який працює відповідно до типового положення, яку затверджується центральним органом виконавчої влади і забезпечує виконання вимог державної політики у сфері охорони праці.

За недотриманням вимог, керівники ІТ-компаній можуть бути притягнуті до відповідальності, яка виглядає у виді накладання штрафу. Якщо в результаті порушення умов охорони праці є постраждалі працівники то керівні особи ІТ-компаній притягуються до кримінальної відповідальності.

Законом України “Про охорону праці” [3] регламентуються загальні положення державної політики в галузі охорони праці, а конкретизуються ці положення нормативно-правовими актами про охорону праці, зокрема Наказом Міністерства соціальної політики України 14.02.2018 № 207, який зареєстровано в Міністерстві юстиції України 25 квітня 2018 р. за №508/31960 «Про затвердження Вимог щодо безпеки та захисту здоров’я працівників під час роботи з екранними пристроями» [5], яким затверджено нормативно-правовий акт з охорони праці НПАОП 0.00-7.15-18, «Правила охорони праці під час експлуатації

електронно-обчислювальних машин», та «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» ДСанПіН 3.3.2-007-98 [2].

Науково-технічний прогрес привніс серйозні зміни в умови виробничої діяльності робітників розумової діяльності. Їх праця стала більш інтенсивною, напруженою і вимагає значних витрат розумової, емоційної і фізичної енергії. Це призвело до необхідності у знаходженні комплексного рішення проблем ергономіки, гігієни і організації праці, регламентації режимів праці та відпочинку. Охорона здоров'я робітників, забезпечення безпеки умов праці, ліквідація та профілактика професійних захворювань і виробничого травматизму складає одну з головних турбот людського суспільства.

8.2 Пожежна безпека

Вимоги до пожежної безпеки на підприємстві неухильно повинен дотримуватися кожен співробітник, а організаційна складова при цьому покладається на посадових осіб за відповідним рішенням керівництва і прописується в посадових інструкціях і положеннях по структурним підрозділам. Зокрема, вказуються конкретні території, ділянки, зони, об’єкти, цілі будівлі і їх частини, поверхи, на яких відповідального співробітника повинне проводити такі організаційні роботи.

Відповідальні особи зобов’язуються розробити, впровадити та підтримувати в певному інструкцією і положенням на ввірених їм об’єктах протипожежний режим і інструкції відповідно до вимог, викладених в нормативних актах. Передбачено також створення підрозділу добровільної пожежної охорони та пожежно-рятувальної команди в його складі.

Встановлений режим включає порядки з описом місць спеціального призначення та правила їх користування та утримання, наприклад:

– евакуаційних шляхів;

– так званих «курилок»;

– місць складування продукції та сировини;

– стоянки транспорту.

Також встановлюється порядок роботи та технічного обслуговування:

– вентиляційного устаткування;

– засобів пожежогасіння і захисту від загорянь;

– нагрівальних приладів;

– електрообладнання.

Розробляються і впроваджуються правила роботи з відкритим вогнем і горючими матеріалами. Створюються графіки проходження інструктажів з пожежної безпеки співробітників, а також порядок і терміни перевірок знань пожежно-технічного мінімуму, в тому числі, тих працівників, які відповідальні за цю ділянку роботи на підприємстві. При цьому можуть передбачатися внутрішні лекції, семінари, тренінги та практичні заняття на підприємстві, а також зовнішні – на базі спеціалізованих навчальних центрів з професійними викладачами. Важливою складовою протипожежного режиму на будь-якому об’єкті є розробка і впровадження порядку дій при виникненні пожежі. Неодмінно має бути план евакуації, описано, як повинні відключатися електроустановки, що і в якій послідовності необхідно робити співробітникам.

Відповідно, для кожного об’єкта, кожного приміщення (крім коридорів, санвузлів, басейнів і подібних приміщень), окремих видів робіт складаються інструкції, за якими повинен працювати персонал, залучений на певних ділянках і в виконанні окремих видів робіт. За інструкціями проводиться навчання (інструктаж) персоналу з подальшим контролем знань.

Детально про те, як розробити протипожежний режим, прописати порядки та інструкції, пояснюють на тематичних курсах і семінарах. [4]

8.3 Пропозиції щодо підвищення працездатності ІТ-фахівців

Поява та впровадження нових інформаційно-комунікаційних технологій зумовлює необхідність подальшого вдосконалення охорони праці фахівців ІТ-індустрії. Все це потребує розробки нових нормативно-правових актів з регламентації праці та відпочинку фахівців ІТ-індустрії і стандартів підприємств, центрів комп'ютерної техніки, центрів інформаційних технологій, сучасних комп'ютерних класів. Для підвищення розумової працездатності то зорової роботи повинна здійснюватися ергономічна оптимізація в рамках системи «оператор-термінал», яка сприятиме результативній фізичній та інтелектуальний працездатності і відновленню психосоматичного здоров'я фахівців ІТ-індустрії.

Особливе значення у соціальному захисті цієї категорії працівників належить прийняття комплексного договору, який може забезпечити фахівців додатковими пільгами та компенсаціями.

Пропозиції щодо підвищення працездатності ІТ-фахівців, поділимо на декілька категорій:

1) Середовище і розпорядок праці. Для мінімізації негативних ефектів, що пов'язані з перевтомленням ІТ-фахівців, потрібно чітко прописати і реалізувати графік періодів праці-відпочинку, щоб фахівець міг можливість переключити увагу, дати можливість відпочити очам, мозку, елементарно, встати розім'яти ноги. Також потрібно зробити максимально комфортними умови мікроклімату у офісному приміщенні, де працюють ІТ-фахівці. Мається на увазі встановлення і експлуатація, коли виникає необхідність, кондиціонерів, опалення, та системи вентиляції, задля попередження перегрівання, переохолодження ІТ-фахівців, і подальшої неможливості ними виконувати свої функції. Також, за можливості, нами пропонується введення практики віддаленої праці ІТ-фахівцями, якщо роботодавець не може забезпечити оптимальні і безпечні умови в офісному приміщенні, або якщо фахівця вони не влаштовують із певних причин.

2) Фізичні і психоемоційні чинники. Першим і найважливішим чинником, що впливає на працездатність ІТ-фахівців є робоче місце, і саме тому, роботодавець має забезпечити максимальний його комфорт і безпеку. Гарантією цих факторів може слугувати сертифікація меблів, що використовуються на підприємстві ІТ-галузі. Тому нами пропонується закупівля тільки меблів, які пройшли сертифікацію на відповідність. Під психоемоційними чинниками ми розуміємо гарне самопочуття фахівців, позитивний настій, гарний психологічний клімат у колективі, тощо. Задля того, щоб психоемоційні чинники мали максимально позитивний ефект, керівництву слід поводити заходи, які сприятимуть укріпленню і покращенню міжособистісних стосунків у колективі, таких як психологічні тренінги, таймбілдінг, спортивні змагання і естафети. Також, сюди можна віднести розробку і впровадження системи мотивації працівників, як фінансової, так моральної і адміністративної.

8.4 Розробка заходів з умов поліпшення охорони праці

Згідно аналізу умов праці в розглянутому приміщенні, ми одержали наступні результати:

– розмірі приміщення, у розрахунку на одному працюючого, відповідають нормативам;

– мікроклімат відповідає нормативному значенню;

– акустичні умови роботи не перевищують нормативних значень.

Таким чином можна припустити, що основною причиною можливого зниження працездатності програміста є психофізіологічний фактор, тому основна пропозиція буде така: дотримання позитивної психологічної атмосфери в колективі та регламентованого режиму праці та відпочинку, організація робочого місця з урахуванням ергономічних вимог.

Рекомендовані заходи: регулярні періодичні наочні огляди персоналом шляхів для евакуації людей із приміщення, відповідно до плану евакуації (який

повинен розташовуватись на видному місці у приміщенні), включення до колективного договору мінімально можливого вмісту аптечок з обов'язковою наявністю масок-клапанів, або іншого спорядження для штучного дихання. Регулярна періодична перевірка параметрів заземлення та занулення (вимірювання опору ланцюга) [9].

Регулярна наочне знайомство персоналу із шляхами для евакуації людей із приміщення відповідно до плану евакуації, забезпечення розподільних щитів спеціальними розетками з заземлюючими контактами; організація заземлення всіх приладів і пристроїв, які працюють при напрузі вище 36 В.

Так як при ураженні електричним струмом у людини може статися фібриляція шлуночків серця, в організації бажано мати дефібрилятор і підготовлений персонал для роботи з ним.

8.5 Розрахункова частина

Для захисного штучного заземлення будемо застосовувати вертикальні електроди з сталевого прокату круглого перерізу діаметром 35 мм, довжиною L=2 м, та горизонтальний електрод – металева полоса з перетином 35·4 мм. Напруга – 220/380 В. Розрахункова схема розташування заземлюючих електродів – по контуру прямокутником (рис. 8.1).

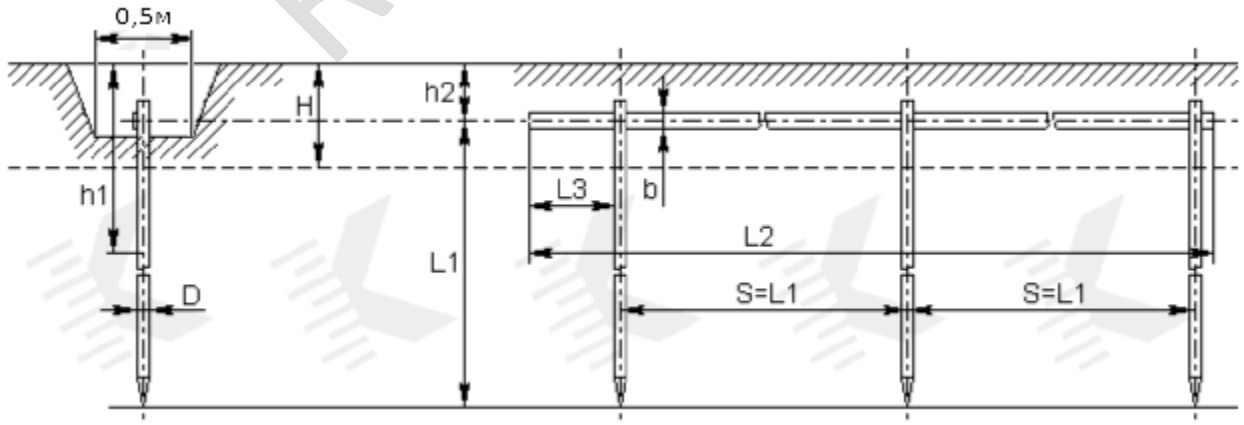

Рисунок 8.1 – Схема штучного заземлення

Розрахунок проводиться за допустимим опором розтіканню струму заземлювача.

Початкові дані для розрахунку захисного заземлення: тип верхнього шару ґрунту – чорнозем, нижнього шару ґрунту – глина (питомий опір ρ_2 = 40 Ом·м). Умовна товщина верхнього шару ґрунту: H=0,6 м. Відстань між вертикальними заземлювачами (електродами) A=3 м. Глибина закладення горизонтального контура заземлення t=0,75 м. Опір заземлювача, який нормується: $R_{ЗН}$ = 4 Ом. Необхідно визначити необхідну кількість вертикальних заземлювачів та довжину полоси (горизонтального заземлювача).

Виконуємо розрахунок.

Відстань від центра вертикального заземлювача до поверхні землі:

$$T = t + L/2 = 0,75 + 2/2 = 1,75 \text{ м.}$$

Розрахунковий питомий опір ґрунту (з врахуванням того, що фактично вся конструкція заземлювача розташовується у нижньому шарі ґрунту):

$$\rho = \psi \cdot \rho = 1,36 \cdot 40 = 54,5 \text{ Ом·м.}$$

де ψ = 1,36 – табличне значення коефіцієнта сезонності для відповідної кліматичної зони у багатошаровому гранті [10]; ρ_2= 40 Ом·м – табличне значення питомого опору нижнього шару ґрунту (глина) [10].

Діаметр вертикального електрода (заданий) Dв= 35 мм = 0,035 м.

Відношення A/L=3/2=1,5.

Опір розтіканню електричного струму одного електрода вертикального заземлювача з урахуванням заглиблення заземлювача [10]:

$$R_O = 0,366 \cdot (\rho/L) \cdot [\lg(2L/Dв) + (1/2) \cdot \lg((4T+L)/(4T-L))] =$$

$$= 0,366 \cdot (54,5/2) \cdot [(\lg(2 \cdot 2/0,035) + (1/2) \cdot \lg((4 \cdot 1,75+2)/(4 \cdot 1,75-2))] = 21,7 \text{ Ом.}$$

Визначаємо коефіцієнт екранування вертикальних електродів Кев=0,53 при орієнтовній кількості вертикальних електродів, яке дорівнює 5 [10].

Визначаємо необхідну кількість вертикальних електродів заземлювача (без врахування горизонтального заземлювача), при $R_{ЗН}$ = 4 Ом:

$$N = R_O / (Кев \cdot R_{ЗН}) = 21,7 / (0,53 \cdot 4) = 10,2 \approx 10 \text{ шт.}$$

Визначаємо довжину з’єднуючої полоси:

$$L_П = 1{,}05 \cdot A \cdot N = 1{,}05 \cdot 3 \cdot 10 = 32{,}3 \approx 32 \text{ м.}$$

Опір розтіканню електричного струму з’єднуючої полоси з урахуванням кліматичного коефіцієнта питомого опору ґрунту $K_П$ [10]:

$$R_П = 0{,}366 \cdot (\rho \cdot K_П / L_П) \cdot \lg(2(L_П \cdot L_П)/(B \cdot t)) =$$

$$= 0{,}366 \cdot (40 \cdot 5/40) \cdot \lg((2 \cdot 40^2)/(0{,}035 \cdot 0{,}75)) = 11{,}14 \text{ Ом.}$$

де $K_П$=5 – табличне значення кліматичного коефіцієнта питомого опору ґрунту для відповідної кліматичної зони для з’єднуючої полоси [10]:

B = 35 мм. = 0,035 м.- ширина з’єднуючої полоси (задана).

Загальний опір розтіканню електричного струму заземлювача [10]:

$$R = (R_O \cdot R_П)/(R_O \cdot \eta_П + N \cdot R_П \cdot Кев) =$$

$$= (21{,}7 \cdot 11{,}14)/(21{,}7 \cdot 0{,}55 + 10 \cdot 11{,}14 \cdot 0{,}53) = 3{,}4 \text{ Ом.}$$

де $\eta_П$ = 0,55 – табличне значення коефіцієнта екранування з’єднуючої полоси [10].

Умова $R \le R_{ЗН}$ виконується ($3{,}4 \le 4$).

Так як при 10 вертикальних електродах R суттєво більше $R_{ЗН}$, зменшимо кількість вертикальних електродів N до 9 і виконаємо перерахунок. У результаті остаточно отримали: R = 3,9 Ом. при кількості вертикальних електродів N = 9.

Висновки до розділу

Дотримання всіх необхідних умов праці не лише сприяє збереженню здоров’я працівників, а також підвищує ефективність виробництва в цілому.

З цих міркувань було здійснено аналіз приміщення, призначеного для праці програмістів, проведено розгляд питань пожежної безпеки, небезпечних та шкідливих факторів, що негативно впливають на програмістів під час роботи.

Тільки повна усвідомленість працівника про можливі небезпеки, що можуть підстерігати його на робочому місці та дотримання вимог нормативних актів о питань охорони праці та відповідних рекомендацій фахівців, дозволять значною мірою знизити негативний вплив шкідливих та небезпечних факторів при роботі з комп’ютером на організм людини. Виконано розрахунок захисного штучного заземлення, як одного з ключових факторів безпеки програміста.

9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для системи поведінкового аналізу користувачів за допомогою концепції UEBA.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти наведені теоретичне узагальнення й рішення наукового завдання дослідження методів поведінкового аналізу користувачів за допомогою концепції UEBA.

Рішення даного завдання полягало у вирішенні наступних задач:

– Був проведений огляд існуючих систем поведінкового аналізу користувачів за допомогою концепції UEBA.

– Досліджена система поведінкового аналізу користувачів за допомогою концепції UEBA.

– На основі отриманих результатів досліджень створена програмна реалізація системи поведінкового аналізу користувачів за допомогою концепції UEBA.

Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання поведінкового аналізу користувачів за допомогою концепції UEBA.

Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

При створені програмного забезпечення було використано об'єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня PHP фреймворк Yii2. Дана мова програмування дозволяє найбільш ефективно обробляти дані. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм SEED.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

Проведено маркетингове та економічне обґрунтування ІТ-проєкту, що дозволило визначити ключові фактори успіху даного проекту.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ковальчук В.А. Дослідження та програмна реалізація системи поведінкового аналізу користувачів за допомогою концепції UEBA // Збірник праць молодих науковців ЦНТУ. – Вип. 15. – Кропивницький: ЦНТУ, 2025.

2. Cormen T.H., Leiserson C.E., Rivest R.L., Stein C. Introduction to Algorithms, 3rd Edition (The MIT Press) 3rd Edition – The MIT Press, 2019. – 1292 p.

3. Fenner M. Machine Learning with Python for Everyone (Addison-Wesley Data & Analytics Series) 1st Edition, Kindle Edition. – Addison-Wesley Professional, 2019. – 586 p.

4. Foreman J.W. Data Smart: Using Data Science to Transform Information into Insight 1st Edition. – Wiley, 2013. – 432 p.

5. Hurbans R. Grokking Artificial Intelligence Algorithms. – Manning, 2020. – 631 p.

6. Gusfield D. Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology 1st Edition. – Cambridge University Press, 2008. – 556 p.

7. Kotu V., Deshpande B. Data Science: Concepts and Practice. – Elsevier Science, 2018. – 953 p.

8. Knowledge Base A Complete Guide – 2021 Edition // The Art of Service – Knowledge Base Publishing, 2020. – 306 p.

9. Knuth D. The Art of Computer Programming, Vol. 1: Fundamental Algorithms, 3rd Edition 3rd Edition. – Addison-Wesley Professional, 2019. – 672 p.

10. Mattmann C. Machine Learning with TensorFlow, Second Edition. – Manning, 2020. – 1124 p.

11. Mueller J.P., Massaron L. Machine Learning For Dummies. – Wiley, 2016. – 714 p.

12. Teofili T. Deep Learning for Search. – Manning, 2019. – 695 p.

13. Rungta K. TensorFlow in 1 Day: Make your own Neural Network. – Publishdrive, 2019. – 587 p.

14. Weidman S. Deep Learning from Scratch: Building with Python from First Principles. – O'Reilly. 2019. – 252 p.

15. Rajasekaran S., Vijayalakshmi Pai G.A. Neural networks, fuzzy logic, and genetic algorithms: synthesis and applications (with cd-rom) Kindle Edition. – PHI, 2013. – 628 p.

16. Усік, П.С., Смірнова, Т.В., Буравченко, К.О., Смірнов, О.А., Улічев, О.С., Смірнов, С.А. «Дослідження технологій забезпечення кібербезпеки банківських систем з використанням штучного інтелекту». *Кібербезпека: освіта, наука, техніка.* 2025. Том 1 № 29. С.704–716, 2025

17. Kuznetsov, O., Frontoni, E., Kuznetsova, K., Arnesano, M., Smirnov, O. «A secure biometric authentication architecture for blockchain-driven cyber-physical systems». *Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications*, 2025, pp. 193–224.

18. Kuznetsov, O., Smirnov, O., Akhmetov, B., Alimseitova, Z., Imoize, A.L. «Deep Learning Frontiers in Copy-Move Forgery Detection: Advances, Challenges, and Future Directions». *Advancements in Cybersecurity Next Generation Systems and Applications,* 2025. 202-229.

19. Вінтенко Б., Смірнов О., Миронець І., Смірнова Т., Смірнов С. «Імітаційна модель шляхів вхідних даних комп'ютерної інтелектуальної системи підтримки оператора енергоблоку АЕС». *Комбінаторні конфігурації та їхні застосування: Матеріали XXVII Міжнародного науково-практичного семінару, присвяченого 125-річчю Національного університету «Запорізька політехніка»* (Запоріжжя-Кропивницький-Київ, 4-6 червня 2025 р.). Запоріжжя: НУ «Запорізька політехніка», 2025. С.82-91.

20. Al-Azzeh, J., Ayyoub, B., Mesleh, A., Smirnova, T., Gnatyuk, S., Drieiev, O., Smirnov, O., Dorenskyi, O. «Cloud-Based Information System for

Evaluating Caverns in the Process of Blasting Metal Surfaces of Details». *International Review on Modelling and Simulations* 18 (1), 2025. pp. 32-42.

21. Вінтенко Б.Ю., Смірнов О.А., Миронець І.В., Смірнова Т.В., Коваленко О.В., Мацуй А.М. «Модель шляхів отримання вхідних даних комп'ютерної інтелектуальної системи підтримки оперативного персоналу АЕС». *Центральноукраїнський науковий вісник. Технічні науки.* 2025. Вип. 11(42), ч. II. С.52-62.

22. Вінтенко Б.Ю., Смірнов О.А., Миронець І.В., Смірнова Т.В. «Методи забезпечення відмовостійкості інтелектуальних систем підтримки оператора». *VIII міжнародна науково-практична конференція "Інформаційна безпека та комп'ютерні технології",* м. Кропивницький. 24-25 квітня 2025 р. – Кропивницький: ЦНТУ. – 2025. – С. 44-46.

23. Смірнов, О.А., Константинова, Л.В., Конопліцька-Слободенюк, О.К., Козірова, Н.В, Якименко, Н.М., Доренський, О.П., Буравченко, К.О. «Дослідження інструментів штучного інтелекту для роботи з базами даних та аналізу даних». *Кібербезпека: освіта, наука, техніка.* 2025. №3(27), С. 429–448.

24. Lakhno, V., Malyukov, V., Smirnov, O., Bebeshko, B., Chubaievskiy, V., Zhumadilova, M., Malyukova, I., Smirnov, S. «Multifactorial Model for Targeted Attacks Counteracting Within the Framework of a Multi-Step Quality Game with Fuzzy Information». *8th International Symposium on Intelligent Informatics, ISI 2023*, 2025. vol 389. pp 377-389. Springer, Singapore.

25. Smirnov O., Fedorov E., Neskorodieva A., Neskorodieva T. «Intellectual Classification method of Gymnastic Elements Based on Combinations of Descriptive and Generative Approache». *CEUR Workshop Proceedings* Volume 3664, 2024, Pages 11-23.

26. Kuznetsov, O., Kryvinska, N., Ilchenko, O., Smirnova, T., Ulianovska, Y. «Comparative Analysis of Cryptocurrency Trading Platforms Using the Analytic Hierarchy Process». *CEUR Workshop Proceedings*, 2023, 3628, pp. 106-115.

27. Malyukov V., Bebeshko B., Lakhno V., Smirnov O., Malyukova I., Mohylnyi H. «Managing the Purchase-Sale Process of Digital Currencies Under Fuzzy Conditions». *Lecture Notes in Networks and Systems,* 2023, 729 LNNS, pp. 104–112.

28. Al-Mudhafar Aqeel, A.M., Smirnova, T., Buravchenko, K., Smirnov, O. «The method of assessing and improving the user experience of subscribers in software-configured networks based on the use of machine learning». *Advanced Information Systems*, 2023, 7(2), pp. 49-56.

29. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yenchev, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». *CEUR Workshop Proceedings,* Volume 3530, 2023, pp. 256-265.

30. Аль-Мудхафар Акіл Абдулхуссейн М., Смірнова Т.В., Буравченко К.О., Смірнов О.А. «Метод оцінки та підвищення користувальницького досвіду абонентів в програмно-конфігурованих мережах на основі використання машинного навчання». Сучасні інформаційні системи, 2023, том 7, № 2, С. 49-56.

31. Smirnov, O., Karapetyan, A., Fedorov, E., «Creating Neural Network and Single Solution Human-Based Metaheuristic Methods of Solving the Traveling Salesman Problem». CEUR Workshop Proceedings, Volume 3312, 2022, pp. 47-58.

32. Smirnov, O., Neskorodieva, T., Fedorov, E., Rudakov, K., Neskorodieva, A. «Method Detection Audit Data Anomalies on Basis Restricted Cauchy Machine» CEUR Workshop Proceedings, Volume 3187, 2022, pp. 1-12.

33. Smirnov O., Smirnova T., Anas M. Al-Oraiqat, Drieiev O., Polishchuk L., Sheroz Khan, Yassin M. Y. Hasan, Aladdein M. Amro, Hazim S. AlRawashdeh «Method for Determining Treated Metal Surface Quality Using Computer Vision Technology». Sensors (Basel, Switzerland) Volume 22, Issue 16, 6223, 2022.

34. Smirnov, O., Lakhno, V., Akhmetov, B., Chubaievskyi, V., Khorolska, K., Bebeshko, B. «Selection of a Rational Composition of İnformation Protection Means Using a Genetic Algorithm». In: Rajakumar, G., Du, KL., Vuppalapati, C., Beligiannis, G.N. (eds) Intelligent Communication Technologies and Virtual Mobile

Networks. Lecture Notes on Data Engineering and Communications Technologies, vol 131. 2023. Springer, Singapore. pp. 21-34.

35. Kuznetsov, A., Oleshko, I., Chernov, K., Bagmut, M., Smirnova, T. «Biometric authentication using convolutional neural networks». Lecture Notes in Networks and Systems. Volume 152, 2021, Pages 85-98.

36. Smirnov O., Kuznetsov A., Kryvinska N., Kiian A., Kuznetsova K. «Full Non-Binary Constant-Weight Codes». SN Computer Science, Vol 2, 337, 2021. https://doi.org/10.1007/s42979-021-00739-w.

37. Smirnov O., Neskorodieva T., Fedorov E., Rymar P. «Neural Network Modeling Method of Transformations Data of Audit Production with Returnable Waste». CEUR Workshop Proceedings Volume 3101, 2021, Pages 192-207.

38. Smirnov, O., Kuznetsov, A., Potii, O., Poluyanenko, N., Stelnyk, I., Mialkovsky, D. «Combining and filtering functions in the framework of nonlinear-feedback shift register». International Journal of Computing; 2020, Volume 19, Issue 2 – Research Institute for Intelligent Computer Systems – 2020. – P. 247-256.

39. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». CEUR Workshop Proceedings. Volume 2740, 2020, Pages 102-114.

40. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), Ukraine, Kyiv, May 14-18. 2020. P. 172-177.

41. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhiienko R., Babenko V., Kuznetsova T., «Representation of Cascade Codes in the Frequency Domain». In: Radivilova T., Ageyev D., Kryvinska N. (eds) Data-Centric Business and Applications. Lecture Notes on Data Engineering and Communications Technologies, vol 48. Springer, Cham. 2021. pp 557-587.

42. Smirnov, O., Drieieva, H., Drieiev, O., Polishchuk, Y., Brzhanov, R., Aleksander, M. «Method of fractal traffic generation by a model of generator on the graph». CEUR Workshop Proceedings Volume 2616, 2020, Pages 366-379.

43. Smirnov, O., Drieieva, H., Drieiev, O., Simakhin, V., Bondar, S., Odarchenko, R. «Managing multifractal properties of the binary sequence generated with the Markov chains», CEUR Workshop Proceedings Volume 2608, 2020, Pages 633-645.

44. Smirnov O. Kuznetsov A., Zaicenko Yu., Pastukhov M., Oleshko O., Kuznetsova K., «Formation of Discrete Signals with Special Correlation Properties». International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019; Odessa; Ukraine; 9-13 September 2019. P.22-28.

45. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». International Journal of Computing; 2019, Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019. – P. 393-407.

46. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), Kyiv, Ukraine, 8 – 11 October 2019 . P.517-522.

47. Smirnov, O., Ulichev, O., Meleshko, Y., Khokh, V., Goncharenko, I. «Method of Choosing Objects for Informational Influence in Social Networks during Information Campaign Based on the Analytic Hierarchy Process». CEUR Workshop Proceedings, Vol 2588, P. 215-227, 2019.

48. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». CEUR Workshop Proceedings, Vol 2588, P. 90-106, 2019.

49. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Averchev, A., Pastukhov, M., Kuznetsova, K., «Formation of Pseudorandom Sequences with Special Correlation

Properties», 2019 3rd International Conference on Advanced Information and Communications Technologies, AICT -2019/ Lviv, Ukraine, 2-6 July, 2019, P. 395-399.

50. Smirnov, O., Kuznetsov, A., Kiian, A., Zamula, A., Rudenko, S., Hryhorenko, V., «Variance Analysis of Networks Traffic for Intrusion Detection in Smart Grids», 2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS), Kyiv, Ukraine April 17-19, 2019 P. 353-358.

51. Smirnov, O., Kuznetsov, A., Kavun, S., Babenko, B., Nakisko, O., Kuznetsova, K., «Malware Correlation Monitoring in Computer Networks of Promising Smart Grids», 2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS), Kyiv, Ukraine April 17-19, 2019 P. 347-352.

52. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Pastukhov, M., Kuznetsova, K., Prokopovych-Tkachenko, D., «Discrete Signals with Special Correlation Properties», CEUR Workshop Proceedings Volume 2353, CEUR Workshop Proceedings 2019, Pages 618-629.

53. Smirnov, S., Bulekbaeva, G., Kikvidze, O.G., Lakhno, V., Brzhanov, R., Tabylov, A. «Computer simulation in the MathCAD package of plastic deformation of the deposited layer on the flat surface of the part». Journal of Theoretical and Applied Information Technology Volume 97, Issue 20, 2019, Pages 2467-2484. (Scopus).

54. Smirnov A.A., Kuzne tsov A.A., Danilenko D.A., Berezovsky A., «The statistical analysis of a network traffic for the intrusion detection and prevention systems», Telecommunications and Radio Engineering. – Volume 74, Issue 1. – Begel House Inc. – 2015. – P. 61-78.