МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК
СЕКЦІЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ПРОЕКТУВАННЯ

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему: «Інформаційна технологія виявлення мережевих атак в критичних інформаційних системах»

за спеціальністю 122 «Комп'ютерні науки та інформаційні технології», освітньо-професійна програма «Інформаційні технології проектування»

Виконавець роботи: студент групи ІТ-51-6 Холявка Євген Петрович

Кваліфікаційна робота бакалавра
захищена на засіданні ЕК
з оцінкою _____ «__» _____ 2019 р.

Науковий керівник _____ професор Лавров Є.А.
(підпис) (науковий ступінь, вчене звання, прізвище та ініціали)

Голова комісії _____ Шифрін Д. М.
(підпис) (науковий ступінь, вчене звання, прізвище та ініціали)

Засвідчую, що у цій дипломній роботі немає
запозичень з праць інших авторів
без відповідних посилань.

Студент _____
(підпис)

Суми-2019

Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра комп'ютерних наук
Секція інформаційних технологій проектування
Спеціальність 122 «Комп'ютерні науки та інформаційні технології»
Освітньо-професійна програма «Інформаційні технології проектування»

ЗАТВЕРДЖУЮ
Зав. секцією ІТП

_____ В. В. Шендрик
«____» _____ 2019 р.

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА СТУДЕНТУ

Холявці Євгену Петровичу

1 Тема роботи *Інформаційна технологія виявлення мережевих атак в критичних інформаційних системах*

керівник роботи *Лавров Євгеній Анатолійович, професор*,

затверджені наказом по університету від « 17 » травня 2019 р. № 0834-III
2 Строк подання студентом роботи «3» червня 2019 р.
3 Вхідні дані до роботи
мережеві атаки, набір даних NSL KDD, WEKA, кластеризація, OneR Attribute Evaluator, Correlation Attribute Evaluator, Information Gain Attribute Evaluator, Gain Ratio Attribute Evaluator, мережі Кохонена
4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
вступ, аналіз проблем виявлення мережевих атак, аналіз методів дослідження мережевих атак, проектування інформаційної системи виявлення мережевих атак, розробка інформаційної технології виявлення мережевих атак, висновки, список використаної літератури
5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
об'єкт, предмет, мета, актуальність, практична цінність, класифікація мережевих атак, методи виявлення мережевих атак, задача виявлення мережевих атак, як задача кластеризації, вибір засобів реалізації, реалізація технології кластеризації ситуацій, аналіз результатів, висновки

6. Консультанти розділів роботи:

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
Аналіз проблем виявлення мережевих атак	*Лавров Є.А.*		
Аналіз методів дослідження мережевих атак	*Лавров Є.А.*		
Проектування інформаційної системи виявлення мережевих атак	*Лавров Є.А.*		
Розробка інформаційної технології виявлення мережевих атак	*Лавров Є.А.*		

7.Дата видачі завдання_____

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Ознайомлення з поточним станом ринку	01.04.19-02.04.19	
2.	Визначення потреби виявлення мережевих атак і аномалій	03.04.19-03.04.19	
3.	Ідентифікація мети та задач	04.04.19-04.04.19	
4.	Аналіз документації	05.04.19-10.04.19	
5.	Визначення інструментарію	11.04.19-10.04.19	
6.	Ранжування за допомогою GAE алгоритму	19.04.19-23.04.19	
7.	Ранжування за допомогою GRAE алгоритму	24.04.19-26.04.19	
8.	Ранжування за допомогою OneR алгоритму	29.05.19-01.05.19	
9.	Ранжування за допомогою CAE алгоритму	02.06.05-06.05.19	
10.	Аналіз результатів	07.05.19-15.05.19	
11.	Оформлення документації	16.04.19-27.05.19	

Студент _____ Холявка Є.П.
(підпис)

Керівник роботи _____ професор Лавров Є.А.
(підпис)

РЕФЕРАТ

Тема бакалаврської роботи: «Інформаційна технологія виявлення мережевих атак в критичних інформаційних системах».

Пояснювальна записка містить вступ, 4 розділи, висновки, список використаної літератури та додатки, включає 87 сторінок, 22 таблиці, 27 ілюстрації, 31 джерело інформації.

В першому розділі наведений огляд проблем, актуальність і перспективи розвитку систем виявлення мережевих атак. Проведено аналіз нинішнього стану технологій для вирішення проблем захисту інформації, проаналізовано методи, що застосовуються для рішення задач пов'язаних з кібербезпекою. Проведено огляд існуючих загроз, що впливають на працездатність інформаційних систем.

Другий розділ розкриває мету, поставлені задачі, аналіз методів дослідження. Наводиться порівняльна характеристика існуючих програмних засобів для проведення кластеризації, обґрунтовується використання обраного програмного забезпечення.

В третьому розділі наводиться опис інформаційної системи та розробляється функціональна модель. Наводиться діаграма варіантів використання інформаційної системи виявлення мережевих атак.

В четвертому розділі описується розробка інформаційної технології виявлення мережевих атак. Описується алгоритми вибору атрибутів, що дають кращі результати виявлення аномалій. Порівнюються результати виявлення мережевих атак з результатами аналогічних досліджень.

Результатом проведеної роботи є розроблена інформаційна технологія виявлення мережевих атак та обґрунтовано можливість використання її на практиці.

Ключові слова: мережеві атаки, мережеві аномалії, аналіз даних, WEKA, кластеризація, самоорганізуючі карти Кохонена.

ЗМІСТ

ВСТУП

Актуальність. Одною з найбільших загроз економіці і державності України є загрози, пов’язані з кібербезпекою. В останні роки збільшується кількість вірусів, що розповсюджуються в мережах, та мережевих атак. Незважаючи на велику кількість наукових робіт, задача виявлення мережевих атак в критичних інформаційних системах, де збитки можуть бути загрозливими, вирішена не до кінця.

Задача підвищення захищеності інформаційно-телекомунікаційні систем від різних загроз є важливою фундаментальною проблемою. В рамках даної роботи передбачається розробити новий підхід, в основу якого ляже комплексна система аналізу, заснована на методах інтелектуальної оцінки даних, що дозволяє врахувати різні характеристики вторгнення і прийняти ефективне рішення на основі його глибокого аналізу.

Предмет дослідження. Математична модель і інформаційна технологія виявлення атак.

Об’єкт дослідження. Мережеві атаки в комп’ютеризованих системах управління**.**

Мета. Розробити інформаційну технологію виявлення мережевих атак в критичних інформаційних системах та обґрунтувати можливість використання її на практиці.

Наукова новизна: На відміну від існуючих статистичних методів та методів орієнтованих на використання технології навчання з вчителем, запропонована технологія орієнтована на використання самоорганізуючих карт Кохонена.

Практична цінність. Метод може бути застосовано в системах підтримки прийняття рішень з питань кібербезпеки автоматизованих систем і забезпечити виявлення мережевих атак в критичних системах.

Апробації. Результати дослідження доповідались на конференції «Інформатика, математика, автоматика», 2019 р.

Публікації. За матеріалами дослідження було опубліковано наукові роботи. Список робіт та копії публікацій додаються.

1 АНАЛІЗ ПРОБЛЕМ ВИЯВЛЕННЯ МЕРЕЖЕВИХ АТАК

1.1 Мережеві атаки

Для організації комунікацій в неоднорідному мережевому середовищі, застосовуються набори протоколів TCP/IP. Вони забезпечують сумісність між комп'ютерами різних типів. Даний набір протоколів TCP/IP, популярний завдяки сумісності та наданні доступу до ресурсів глобальної мережі. Поширення стека протоколів TCP/IP показало його слабкі сторони. Віддаленим атакам піддаються розподілені системи, оскільки їх компоненти зазвичай використовують відкриті канали передачі даних, і порушник може не тільки проводити пасивне прослуховування переданої інформації, а й модифікувати переданий трафік [25].

Труднощі виявлення проведеної віддаленої атаки і відносна простота проведення (через надмірну функціональності сучасних систем), виводить цей вид неправомірних дій на одне з перших місць за ступенем небезпеки і заважає своєчасному реагуванню на здійснену загрозу, результатом чого у порушника збільшуються можливість успішної реалізації атаки.

Під віддаленою мережевий атакою розуміємо вплив на програмні компоненти цільової системи за допомогою програмних засобів [1]. Метою атаки є отримання даних або здійснення проникнення.

Мережеві атаки за характером впливу поділяють на [2]:

- пасивні;
- активні.

Пасивний вплив на розподілену обчислювальну систему (РОС) представляє собою деяке втручання, яке надає прямого впливу на роботу обчислювальної системи, але й одночасно здатне втрутитися в її політику безпеки. Відсутність явного впливу на роботу РОС приводить до того, що пасивно віддалений вплив

(ПВВ) складно виявити. Одним із типових прикладів ПВВ в РОС служить прослуховування каналів зв'язку в мережі.

При пасивному впливі не відбувається прямого вторгнення на систему, через що виявлення такого виду атак складніше, ніж виявлення атак з активним впливом [1].

Активна дія на РОС – надає прямого впливу на роботу самої системи, тобто порушує працездатність або змінює конфігурації РОС, тобто порушує політику безпеки, прийняту в даній системі. Активними діями є майже всі типи віддалених мережевих атак. Очевидна відмінність активних впливів від пасивних – це принципова можливість його виявлення, так як в результаті здійснення активних впливів в системі відбуваються деякі зміни. При пасивному ж дії, не залишається абсолютно ніяких слідів (через те, що атакуючий перегляне чуже повідомлення в системі, в той же момент нічого не зміниться) [25].

При активному впливі на функціонування системи виявляється безпосередній вплив, яке може порушити її функціонування, змінити конфігурацію [31].

Класифікація мережевих атак за цілями впливу виділяються три типи [1]:

- атаки розвідки;
- атаки отримання доступу;
- атаки відмови в обслуговуванні.

Дані типи атак не часто застосовуються окремо, найчастіше для досягнення поставлених цілей зловмисники використовують їх в комплексі.

Атаки розвідки – перший крок для підготовки атаки на будь-яку мережу. Вони використовуються зловмисником для збору інформації, яка може забезпечити його даними про можливі вразливості системи, а також необхідними інструментами для їх експлуатації [2].

Атаки типу «відмова в обслуговуванні» (Denial of Service, DoS) вважаються найпоширенішим видом атак. Вони відрізняються від інших типів атак, так як спрямовані не на отримання доступу до мережі або до будь-якої інформації. Їх

мета – виведення системи з ладу або обмеження можливості використання шляхом створення умов, при яких сумлінні користувачі не можуть отримати доступу до наданих ресурсів або послуг. Популярність DoS атак обґрунтована простотою реалізації і великими масштабами завдається шкоди. У разі, якщо атака цього типу виробляється за допомогою великої кількості пристроїв, можна говорити про розподіленої атаки «відмова в обслуговуванні» (Distributed Denial of Sendee, DDoS) [31].

Атаки отримання доступу ґрунтуються на використанні прихованих можливостей або помилок, для отримання несанкціонованого доступу до системи. Засоби, що використовуються зловмисником, часто залежать від типу використовуваної уразливості. Дані про уразливість виходять при проведенні розвідки.

1.2 Аналіз методів виявлення мережевих атак та визначення наявних проблем

Мережеві атаки різноманітні за своєю структурою і складності виявлення. Завдання протидії атакам є важливою для коректного функціонування систем і запобігання порушенню безпеки.

Існує безліч методів виявлення атак. За способами виявлення мережевих атак існує загальноприйнята класифікація, в якій виділяють два класи [3]:

- методи виявлення зловживань;
- методи виявлення аномалій.

Методи виявлення зловживань базуються на порівняння поточного стану системи з образом, званим сигнатурою. Сигнатура – безліч умов, при задоволенні яких настає подія, яке визначається як атака або вторгнення [1]. Основний недолік методів виявлення зловживань – неможливість опису всіх можливих

атак, до того ж, навіть невелика зміна в структурі атаки призводить до неможливості виявлення даними методами.

Загальна схема роботи методів виявлення зловживань приведена на рисунку 1.1.

Рисунок 1.1 – Схема роботи методів виявлення зловживань

Методи виявлення зловживань можна розділити на методи на основі знань, методи машинного навчання і методи штучного інтелекту [3].

Методи на основі знань діють на основі закладених правил і механізмів пошуку, в яких відображаються ознаки атак. Серед цих методів можна виділити:

- сигнатурні методи;
- мови опису сценаріїв;
- методи на основі кінцевих автоматів;
- експертні системи;
- мережі Петрі;
- методи перевірки на моделі.

Найчастіше для оптимальної роботи систем виявлення атак і гарантованого виявлення атак і порушень, методи виявлення зловживань використовуються спільно з методами виявлення аномалій [5].

Аномальна поведінка в інформаційних системах найчастіше є наслідком дій зловмисників. Загальновідома поняття «аномалія» з грецької – відхилення від норми. Поняття ж «виявлення аномалій» є відносно новим, але при цьому воно відразу привернуло увагу фахівців в області інформаційної безпеки [6].

Методи виявлення аномалій ґрунтуються на побудові образу нормальної поведінки системи, при відхиленні від якого поведінка буде вважатися аномальною, тобто буде фіксуватися факт вторгнення або атаки. Недоліком методів виявлення аномалій є помилкові спрацьовування, тобто не кожна аномалія може бути потенційною атакою або загрозою, а система розпізнає її як загрозу.

Схема роботи методів виявлення аномалій наведена на рисунку 1.2

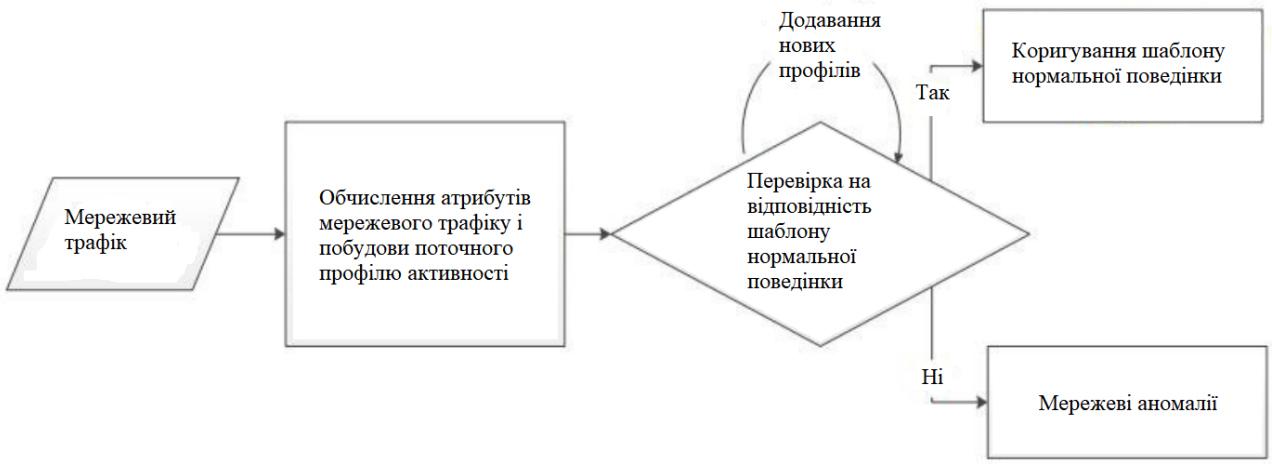


Рисунок 1.2 – Схема виявлення аномалій

Для прийняття рішення про подальші дії або протидії, слід визначити природу виникання аномалії, наслідки та можливі небезпеки, які вона несе. Щоб спростити цей процес, необхідна якась класифікація мережевих аномалій. Але в силу великої кількості і різновидів аномалій, немає загальноприйнятого підходу до цього процесу. Авторами статті [6] пропонується підхід до класифікації мережевих аномалій з точки зору об'єкта впливу. При такому підході мережеві

аномалії поділяють па програмно-апаратні відхилення і порушення мережевої безпеки (рисунок 1.3).

Рисунок 1.3 – Схема класифікації мережевих аномалій

До програмно-апаратних несправностей відносяться:

- апаратні несправності;
- помилки конфігурації;
- помилки програмного забезпечення;
- проблеми продуктивності обладнання.

Апаратні несправності найчастіше пов'язані з поломкою комплектуючих, дефектами заводської збірки, неправильною експлуатацією або ж виникненням збоїв, що виникають під дією будь-яких зовнішніх факторів (скачки напруги, зростання температури).

Помилки конфігурації виникають в результаті невідповідності між встановленими параметрами системи та обладнанням, встановленим в системі. Найчастіше вони виникають після додавання в систему нових пристроїв.

Помилки програмного забезпечення тягнуть за собою аномальну поведінку, що, в свою чергу, тягне за собою несподіваний результат роботи. Зловмисники можуть застосовувати ці помилки і уразливості в своїх цілях [3].

В основі методів виявлення аномалій лежить модель передбачуваної нормальної і очікуваної поведінки користувачів або програмних засобів, інтерпретація відхилень, як можливі аномалії і порушення безпеки. Тобто основна ідея полягає в тому, що атаки – відрізняються від нормальної поведінки.

Методи виявлення аномалій, як і методи виявлення зловживань, включають в себе методи штучного інтелекту, методи машинного навчання і поведінкові методи [4].

Поведінкові методи ґрунтуються на порівнянні поточного поведінки системи з певною моделлю нормальної поведінки.

Поведінкові методи включають в себе [5]:

- вейвлет-аналіз;
- статистичний аналіз;
- аналіз ентропії;
- спектральний аналіз;
- фрактальний аналіз;
- кластерний аналіз.

Загальна схема методів виявлення мережевих атак [26] наведена на рисунку 1.4.

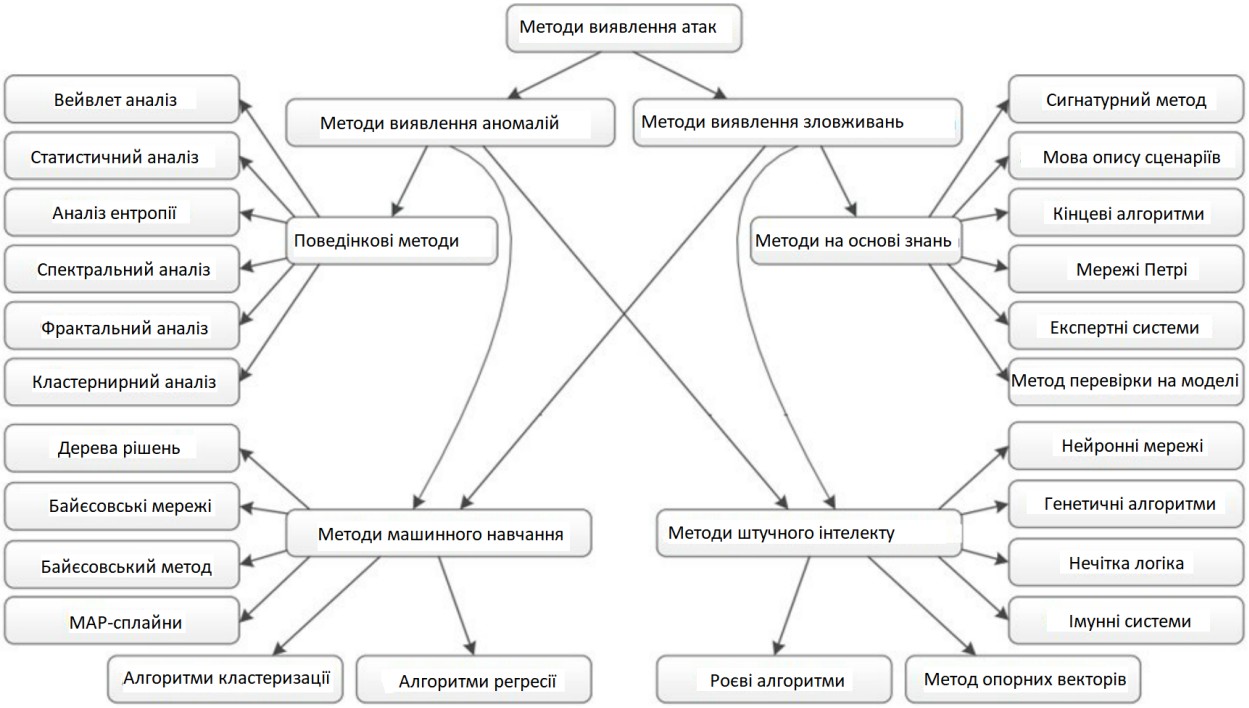


Рисунок 1.4 – Класифікація методів виявлення мережевих атак

Існує поширена помилка, що єдиним і самодостатнім засобом захисту корпоративної обчислювальної мережі підприємства є міжмережевий екран, але це не так. Основна функція брандмауера - фільтрація мережевого трафіку, в результаті якої мережеві пакети або надходять у внутрішню мережу підприємства, або відкидаються відповідно до заданих правил. Сучасні системи виявлення вторгнень (СВВ) здатні контролювати в реальному масштабі часу корпоративну мережу підприємства і діяльність окремих її вузлів, виявляти несанкціоновані дії, і автоматично реагувати на них практично в реальному масштабі часу. Крім того, СВВ можуть аналізувати поточні події, беручи до уваги вже відбулися події, що дозволяє ідентифікувати атаки, рознесені в часі, і, тим самим, прогнозувати майбутні події [26].

Сьогодні є два основних підходи до побудови СВВ: аналіз пакетів, переданих по корпоративної мережі підприємства і аналіз журналів реєстрації операційних систем, систем управління базами даних і додатків. Також ведуться розробки в області експертних систем, які намагаються аналізувати мережевий трафік, «навчатися» на ньому і виявляти аномалії. Це перспективний, але значно

більш складний підхід, який складніше не тільки реалізувати, а й застосовувати на практиці. Найбільша проблема експертних систем - велике число помилкових спрацьовувань. Для їх зменшення потрібна ретельна попередня настройка [28].

У даний час виробники пропонують досить велику кількість продуктів (як програмно-апаратних, так і програмних), що реалізують функції виявлення вторгнень в комп'ютерну систему з використанням зазначених вище підходів, тому проблема вибору постає досить гостро. Крім того, на відміну від міжмережевих екранів, основною складністю застосування яких є грамотне налаштування, використання систем виявлення вторгнень має ряд специфічних особливостей, зв’язаних з різними типами систем і способами їх застосування.

1.3 Постановка задачі

У зв’язку з описаними проблемами визначено мету і задачі роботи.

Мета – розробити технологію виявлення мережевих атак в критичних інформаційних системах управління та обґрунтувати можливість використання її на практиці.

Задля досягнення встановленої мети в роботі необхідно вирішити такі задачі:

1) Розробити алгоритми для виявлення атак на основі застосування моделі кластеризації з використанням технологій Data mining.

2) Запропонувати інформаційну технологію для виявлення атак, провести обчислювальні експерименти з метою оцінки ефективності запропонованих алгоритмів виявлення атак.

2 АНАЛІЗ МЕТОДІВ ДОСЛІДЖЕННЯ МЕРЕЖЕВИХ АТАК

2.1 Кластерний аналіз, як метод виявлення мережевих атак

Проведений аналіз в розділі 1.2 дозволив виявити перспективність використання, для поставлених задач, технології кластерного аналізу.

Кластеризацією називається процес поділу безлічі вихідних даних за деякими критеріями «схожості» на групи, які називаються кластерами. На відміну від класифікації, при якій відбувається розподіл даних по заздалегідь визначених класів, при кластеризації розподіл відбувається одночасно з формуванням класів, тобто класи не були попередньо визначені [8].

Кластеризація допомагає вирішити ряд завдань:

– змістовний аналіз. За рахунок формування груп дозволяє виявити і відслідковувати закономірності в даних і отримати статистику;

– прогнозування. Випливає з попереднього пункту. Відносячи об'єкт до певної групи, можна висунути припущення про подібний поведінці об'єкта з об'єктами групи (кластера);

– виявлення аномалій. Виходячи з припущення, що аномальних даних значно менше нормальних, можна зробити висновок, що влучень в кластер з аномальними даними буде значно менше, ніж в кластер з нормальними даними.

Самоорганізуючі карти Кохонена (Self-Organizing Map, SOM) – це різновид нейронних мереж, які навчаються на основі методу навчання без учителя. При навчанні без учителя результати навчання залежать тільки від структур вхідних даних, навчальна множина складається лише з вхідних векторів і перевірки з будь-якими еталонними значеннями не проводиться [31].

Самоорганізуючі карти вирішують задачу кластеризації і візуалізації вхідних даних, що дозволяє визначати наявність або ж відсутність зв'язку в даних [7].

Самоорганізуючі карти представляють собою безліч нейронів, кількість яких збігається з кількістю кластерів [9]. Нейрони є деякий вектор-стовпець виду:

$$w_j = [w_{j1}, w_{j2}, \dots w_{jn}]^T, \tag{2.1}$$

де n - визначається, виходячи з розмірності вхідних векторів. Крім цього нейрони впорядковані в деяку структуру (найчастіше це двовимірна сітка).

В основі алгоритму побудови систем самоорганізуючих карт лежить три основних процеси [7]:

1) конкуренція. В процесі навчання при подачі вектору даних на вхід вибирається так званий «нейрон-переможець». Їм буде такий нейрон, вектор ваг якого буде мінімально відрізнятися від вхідного вектору:

$$d(x, w_j) = \min_{1 \le i \le j} d(x, w_j), \tag{2.2}$$

де j - номер нейрона-переможця, n-кількість нейронів, $d(x, w_j)$ – відстань в деякій метриці між вектором вхідних даних і вектором нейрона;

2) кооперація. В процесі навчання зміни зачіпають не тільки нейрон-переможець, а й деяку топологічну околиця, розмір якої зменшується з часом. Коригування ваг нейронів в околиці здійснюється за формулою:

$$w_i^{k+1} = w_i^k + \eta_i^k(d, k) \cdot a(k) \cdot \left[x(k) - w_i^k\right], \tag{2.3}$$

де $a(k)$ функція швидкості навчання, спадна від номера циклу навчання. Найчастіше використовується функція виду $a(k) = \frac{A}{k+B}$, де A і B – деякі константи.

$\eta_i^k(d,k)$ функція сусідства, в якій зі зростанням d виконується умова $\eta_i^k \to 0$, де $d_i = \left|\left|r_i - r_{c_j}\right|\right|$ – відстань між *i -м* нейроном і нейроном-переможцем *Cj*.

В якості опції сусідства хороші результати виходять при використанні функція Гауса:

$$\eta_i^k(d,k) = e^{\frac{d_i}{2\sigma(k)}}, \tag{2.4}$$

де σ - деяка функція, спадна від номера циклу;

3) адаптація. Цей механізм дозволяє нейронам топологічного околу за рахунок корекції ваг посилювати відгук при аналогічних вхідних прикладах.

Структурна схема систем самоорганізуючих карт Кохонена зображена на рисунку 2.1

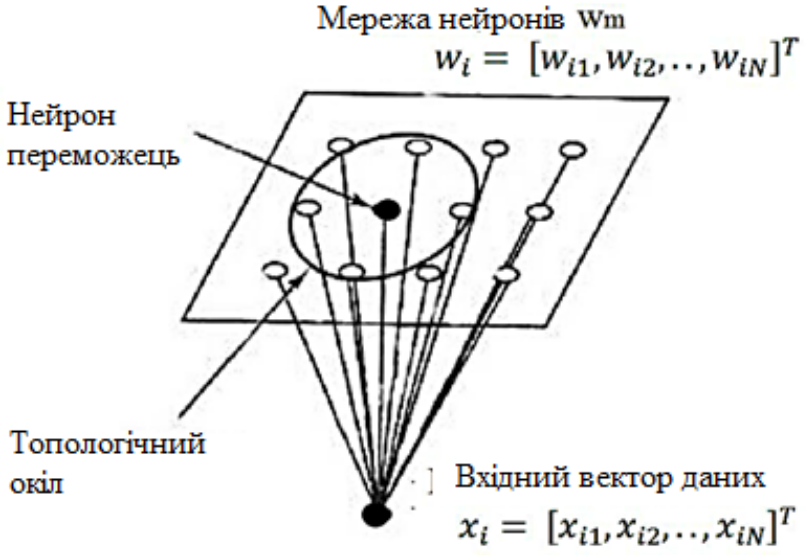


Рисунок 2.1 – Структурна схема самоорганізуючої карти Кохонена

Існує безліч варіантів алгоритмів побудови самоорганізуючих систем. Основні відмінності зачіпають процес ініціалізації карт, який дозволяє

прискорити процес збіжності, алгоритми побудови систем для самоорганізуючої карти можна описати такою послідовністю кроків [31]:

– Ініціалізація карти, тобто початкове завдання векторів ваг для вузлів.

– Підвибірки. Вибираємо вектор X з вхідного простору.

– Пошук максимального подібності. Знаходимо найбільш підходящий нейрон (нейрон-переможець).

– Корекція. Коригуємо ваги нейронів, що входять в топологічну околиця.

– Продовження. Повертаємося до кроку 2 і продовжуємо обчислення до тих пір, доки на карті не припинять відбуватися помітні зміни.

Якість одержуваних карт може відрізнятися від передбачуваних запитів і вимог. Для отримання «хороших» карт існує ряд прийомів, які не потребують будь-яких спеціальних засобів [7].

Для гарного візуального сприйняття карти, в разі великої кількість нейронів, краще використовувати гексагональну форму околів. Це обумовлено тим, що в разі застосування чотирикутних осередків, мають місце яскраво виражені горизонтальні і вертикальні напрямки (рисунок 2.2).

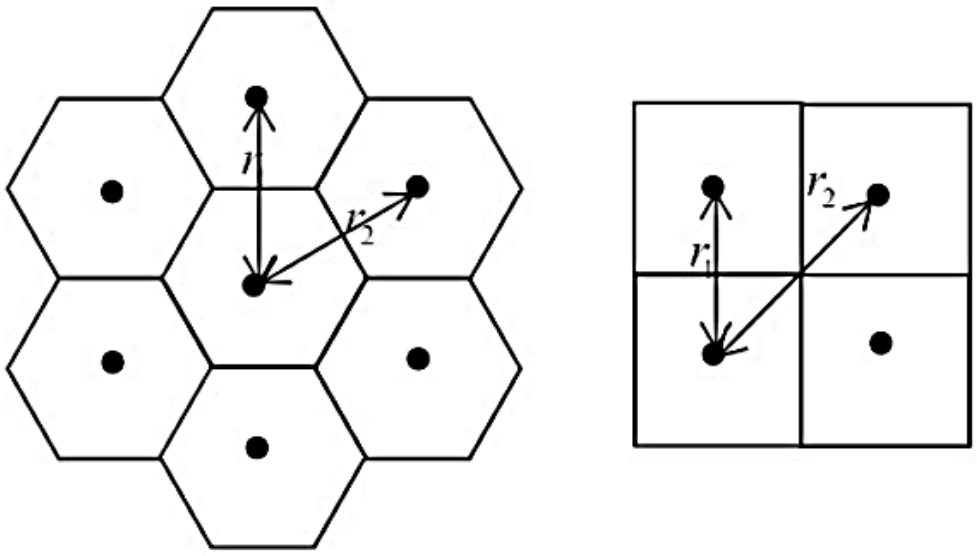


Рисунок 2.2 – Гексагональні і чотирикутні осередки

У разі навчання при малій кількості даних, для отримання хорошої статистичної точності потрібно дуже багато кроків, але частіше кількість зразків даних може бути менше. У цьому випадку дані для навчання використовуються багаторазово. Дані можуть вибиратися різними способами (циклічно, випадково, випадково з деякого базового набору). Але на практиці просте циклічно-впорядковане використання даних нічим не поступається іншим методам [9].

Найчастіше ситуації, такі як аномалії, можуть проявлятися вкрай рідко. У цьому випадку вони можуть бути взагалі не представлені на картах. Для того щоб підвищити вплив таких даних, в процесі навчання потрібно: або у випадковому порядку достатню кількість разів подати їх, або збільшити значення функцій швидкості навчання або функції околів [8].

Якщо висувається вимога про розташування даних в конкретній області карти, для цього можна використати еталонні вектору даних, розміщених в цих місцях. Далі, при навчанні, зберігати для них низьке значення коефіцієнта навчання [9].

2.2 Вибір комп’ютерних технологій побудови моделі кластеризації для виявлення мережевих атак

Мережі Кохонена реалізовані в великій кількості програмних засобів, серед них: Python, R, WEKA, Knime, RapidMiner [19].

Python – відповідно до загального визначення є високорівнева мова програмування загального призначення, який орієнтований на підвищення продуктивності та читання коду. За роки існування, python обзавівся безліччю спеціалізованих бібліотек. Для аналізу даних використовують такі [22]:

- Pandas – відповідає за обробку даних;
- Numphy – працює з матрицями;
- Statsmodels – містить основні статистичні функції і моделі;

- Sklearni Pybrain – спеціалізуються на алгоритмах машинного навчання;
- Matplotlib – відповідає за візуалізацію.

Крім добре документованих бібліотек, python відрізняється гнучкістю і зрозумілим синтаксисом.

R «заточений» під статистичну обробку даних, роботу з графікою і алгоритмами машинного навчання. Велика перевага R – прекрасна візуалізація за допомогою пакета ggplot2 [23].

Weka - це ціла колекція інструментів і алгоритмів для аналізу даних і прогнозування. Серед плюсів інструменту:

- зручний інтерфейс (наприклад, текстовий рядок для введення команд);
- перетворення даних (в тому числі попередня обробка сирих даних);
- підтримка безлічі алгоритмів машинного навчання і можливість їх швидкого застосування;
- зручний висновок результатів роботи алгоритму (легко порівнювати точність різних моделей);
- вибір ознак;
- візуалізація даних;
- можливість проведення експериментів (причому можна запускати відразу декілька алгоритмів на різних завданнях і отримати загальний звіт);
- можливість подання всього процесу рішення задачі в формі графа.

Інструменти Knime і RapidMiner [24] схожі і за формою, і за змістом (хоча перший, на відміну від другого, існує на повністю безкоштовній основі) – тому вирішено об'єднати їх в одну вкладену категорію. Обидва інструменти підтримують безліч стандартних завдань – що стосуються перетворення даних, статистики, машинного навчання та візуалізації. Весь процес аналізу даних представляється у вигляді інтерактивного графа – послідовності операторів, при цьому користувачеві доступні оператори Weka і R [23].

Порівняльна характеристика найпопулярніших інструментів для аналізу даних наведено на таблиці 2.1.

Таблиця 2.1 – Порівняльна характеристика програмних продуктів[22].

Характеристика	Python	R	Weka	Knime/RapidMiner
Час на написання коду	*	*	****	****
Швидкість виводу результатів	**	***	****	***
Витрачання часу на налаштування	*	**	****	***
Різноманітність методів машинного навчання	****	****	***	***
Сума	8	10	15	13

В даній роботі, в зв’язку з необхідністю використання ліцензійних програм і швидкістю виконання кластеризації, пропонується, для застосування самоорганізуючих карт Кохонена і для регресійного аналізу атрибутів тестової вибірки, використовувати програмний пакет WEKA [11].

WEKA (Waikato Environment for Knowledge Analysis) – це відкритий програмний продукт, що розвивається світовим науковим співтовариством, вільно розповсюджуваний під ліцензією GNU GPL. Система дозволяє безпосередньо застосовувати алгоритми до вибірок даних, а також викликати алгоритми з програм на мові Java [11].

WEKA надає безліч реалізацій алгоритмів кластеризації. Алгоритм побудови систем самоорганізуючих карт Кохонена не входить в безліч алгоритмів, що поставляються зі стандартною складанням пакета, але його можна знайти серед плагінів в менеджері пакетів WEKA.

При використанні SOM в WEKA є можливість вибору наступних параметрів, які впливають на роботу алгоритму:

– height і width – розміри решітки, від яких залежить кількість кластерів в розглянутій карті;

– Normalize Attributes – нормалізувати дані (false – працювати з вихідними даними, true – нормалізувати), функція нормалізації перетворює текстові дані до числовим, а все числові дані наводяться до виду [0 ; 1];

– Debug – якщо встановлено значення true, буде виводитися додаткова інформація в консоль;

– Congence Epochs – кількість епох в фазі збіжності;

– Order Epochs – кількість епох в фазі упорядкування.

WEKA надає безліч різних алгоритмів вибору атрибутів, використовуючи які можна скоротити розмірність вихідної тестової вибірки, шляхом відкидання незначущих або малозначущих атрибутів [13]. Це дозволяє прискорити побудову карт в силу зменшення кількості обчислювальних операцій.

Розглянемо лише частину алгоритмів вибору атрибутів, що дали найбільш високий відсоток виявлення мережевих атак.

Information Gain Attribute Evaluator – алгоритм на основі оцінки взаємної інформації за такою формулою:

$$InfoGain\ (class, Attribute) = \mathrm{H}\ (class) - \mathrm{H}\ (class \setminus Attribute), \quad (2.1)$$

де H - ентропія.

Результати розрахунків допомагають побачити силу впливу атрибутів на кінцевий клас (аномальне/нормальна поведінка).

Gain Ratio Attribute Evaluator – є модифікацією Information Gain Attribute Evaluator , відмінність у формулі обчислення:

$$GainR(class, Attribute) = \frac{\text{H}\,(class) - \text{H}\,(class \setminus Attribute)}{\text{H}\,(Attribute)}. \qquad (2.2)$$

Gain Attribute Evaluator нормалізує дані пропорційно їх частоті.

OneR (скор. від One Rule) – цей алгоритм використовується для формування правил класифікації об'єктів. Для значень кожної незалежної змінної знаходяться правила, для кожного з правил обчислюється помилка – це кількість об'єктів з таким же значенням незалежної змінної, але отримані значення не відповідають значенням залежної змінної, що зустрічається найчастіше для обраного значення незалежної змінної. В результаті вибираються атрибути, по яким можна класифікувати обрані об'єкти з найбільшою точністю [13].

Correlation Attribute Evaluator – це метод оцінює цінність атрибута, вимірюючи кореляцію між ним і класом поведінки (аномальне / нормальне).

2.3 Вибір навчальних даних для задачі виявлення мережевих атак

Система виявлення атак може бути побудована і навчена тільки на даних, що охоплюють і моделюють різні атаки і спроби вторгнення. Одним таким загальновідомим і відкритим набором даних є NSL KDD, який був зібраний з ініціативи Управління перспективних дослідницьких проектів Міністерства оборони США (DARPA) [28]. Дані збиралися 7 тижнів і містили 5 мільйонів записів про з'єднання розміром близько 100 байт кожна.

Вектори, що описують з'єднання, імітують чотири категорії атак [10]:

1) DoS – атаки, при яких зловмисник робить багато запитів, перевантажує ресурси або пам'ять комп'ютера.

2) U2R – зловмисник, який має доступ звичайного користувача в системі, може використовувати деяку уразливість для підвищення прав до root – доступ до системи.

3) R2L – якщо зловмисник, має можливість відправляти пакети на комп'ютер мережі, але у нього немає облікового запису на цьому комп'ютері, використовує деяку уразливість, щоб отримати права доступу користувача цієї машини.

4) Probe-атаки – спроба збору інформації про мережу комп'ютерів з метою обходу засобів контролю безпеки.

Кожен вектор містять 41 атрибут. Значення атрибутів наводяться в додатку В.

3 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ВИЯВЛЕННЯ МЕРЕЖЕВИХ АТАК

3.1 Моделювання інформаційної системи

Для аналізу особливості функціонування системи виявлення мережевих атак, необхідно створити її формальне опис у вигляді функціональної моделі. У даній роботі для цих цілей буде використано системне моделювання за методологією IDEF0, з допомогою якої можна побудувати функціональні моделі, що показують структуру і функції проекторів, а також потоки матеріальних об'єктів і інформації, що зв'язують ці функції [16]. Методологія основана на графічному підході до опису (моделювання) системи SADT (System Analysis and Design Technique), для якого характерно:

– представлення блочного моделювання графічно: функція відображається у вигляді блоку, інтерфейс входу / виходу представляє собою дуги, які входять в блок і виходять з нього відповідно;

– опис взаємодії блоків між собою з допомогою інтерфейсних дуг, які виражають "обмеження", що встановлюються, коли і як виконуються функції;

– обмеження кожного рівня декомпозиції кількості блоків (3–6);

– взаємозв'язок діаграм через номери блоків;

– відсутність повторення найменування, унікальність міток;

– синтаксичні правила для блоків і дуг;

– розділення вхідних (обертових) і керуючих даних;

– відсутність впливів організаційних структур на функціональну модель [17].

Результатом використання методології SADT стає функціональна модель, що складається з діаграм, текстових фрагментів і глосарію, пов'язаних між собою

посиланнями. Головними компонентами моделі є діаграми, всі функції і інтерфейси на яких представлені в вигляді блоків і дуг. Точка з'єднання дуги з блоком визначає тип інтерфейсу:

– керуюча інформація входить в блок зверху;
– обробляється інформація відображається з лівого боку блоку;
– результати виконання функції показуються з правого боку блоку;
– механізм (автоматизована система або людина), що виконує операцію, відображається у вигляді дуги, яка входить в блок знизу [17].

Модель IDEF0 являє собою сукупність ієрархічно упорядкованих і взаємопов'язаних діаграм. Так як система виявлення атак є складовою частиною інформаційної системи, то моделювання необхідно почати з неї. Контекстна діаграма, що описує призначення інформаційної системи (ІС) і її взаємодія з зовнішнім середовищем, представлена на рисунку 3.1.

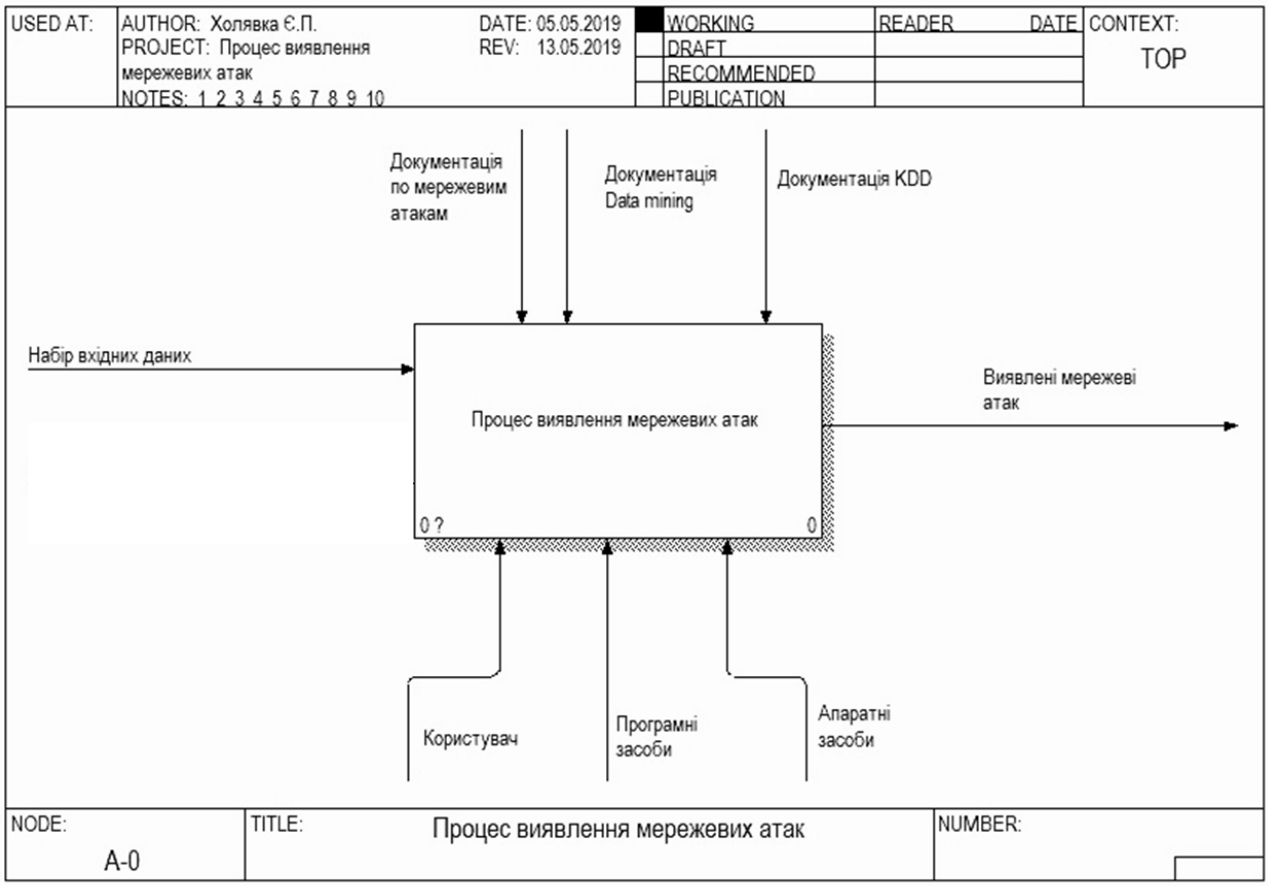


Рисунок 3.1 – Контекстна діаграма

ІС можна розглядати як «чорний ящик», на входи якого поступає набір даних з інформацією про можливі види мережевих атак. Як механізми (ресурсів) для функціонування ІС виступають програмні та апаратні засоби і розробники. Управління здійснюється за допомогою адміністраторів на основі нормативних документів, правил та інструкцій, а також положень політики безпеки. Результатом функціонування ІС є оброблена інформація або інші повідомлення, що виробляються в процесі функціонування системи.

На рисунку 3.2 зображена функціональна модель інформаційної системи, відповідно до якої робота ІС здійснюється за рахунок виконання відповідних функцій наступними підсистемами:

- нормалізація даних: здійснює обробку даних і перетворення текстових даних до числових;
- кластеризація: забезпечує обробку і представлення інформації;
- обробка даних: представляє собою алгоритми вибору атрибутів для зменшення кількості обчислювальних операцій;
- аналіз результатів: призначена для перевірки коректності роботи компонентів ІС;

На практиці склад підсистеми захисту може різнитися залежно від конкретної реалізації, виду діяльності та відповідних вимог до інформаційної безпеки нормативних документів регуляторів.

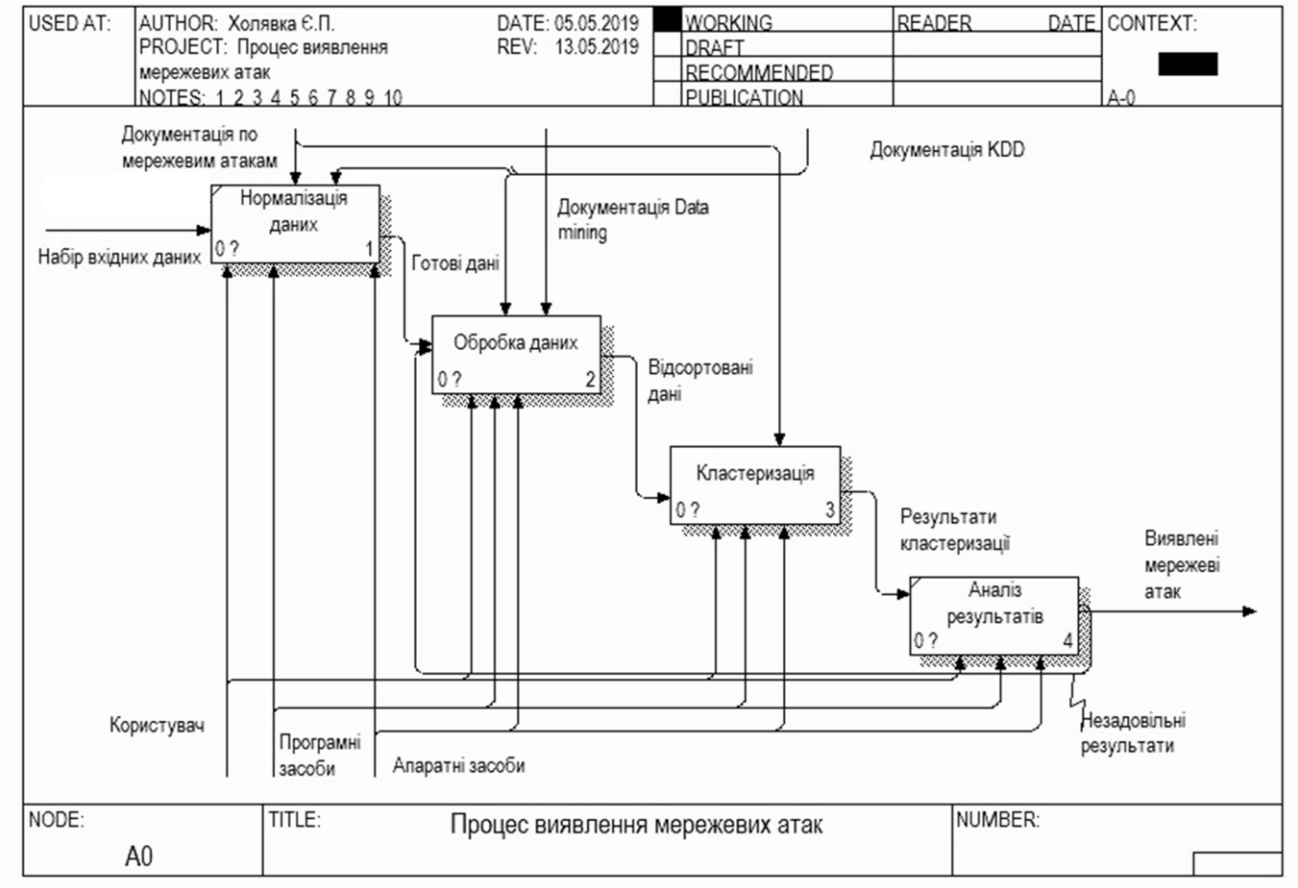


Рисунок 3.2 – Декомпозиція контекстної діаграми

На рисунку 3.3 представлена функціональна модель підсистеми обробки даних, яка містить наступні компоненти:

– ранжування атрибутів: використовуються алгоритми сортування даних в порядку зростання важливості;

– відкидання зайвих атрибутів: використовуються алгоритми вибору атрибутів для зменшення розмірності вхідних даних.

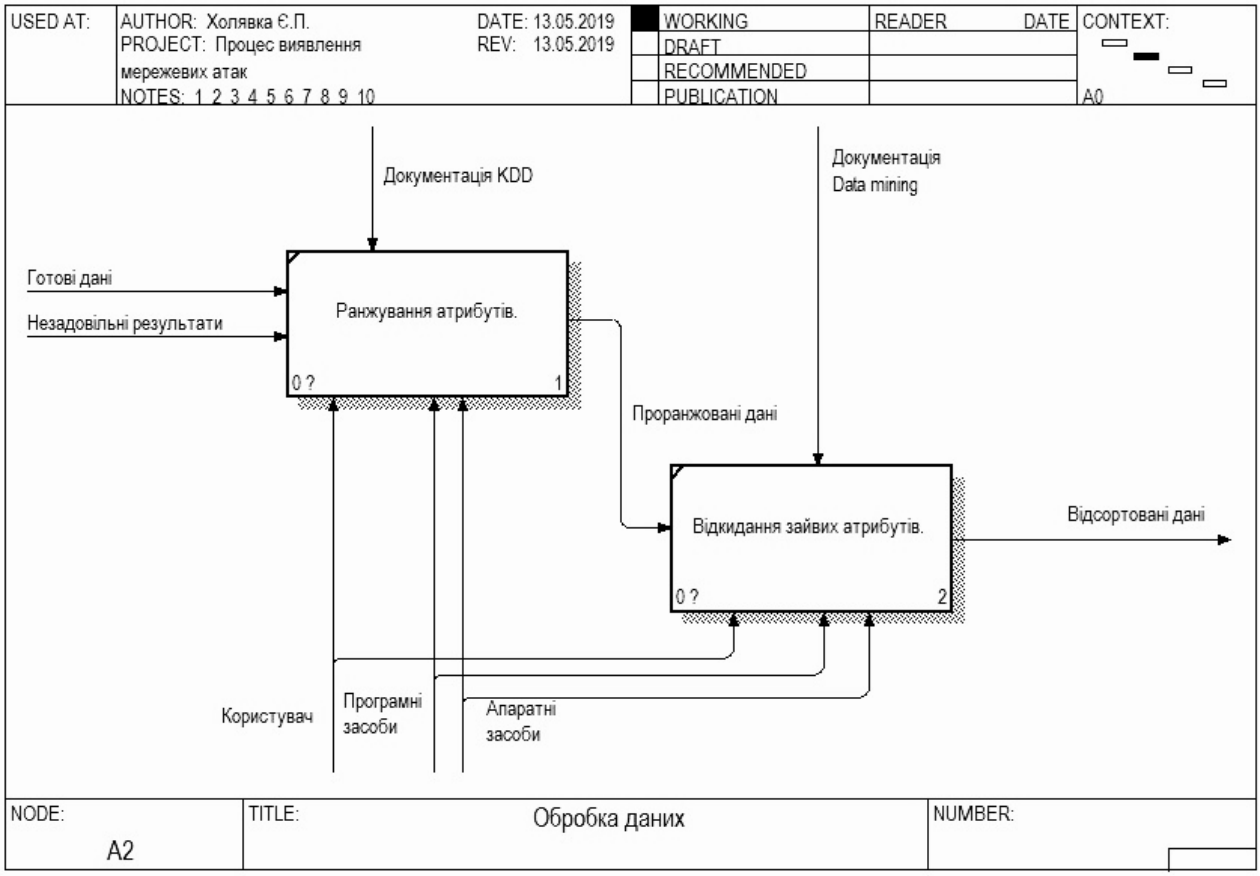


Рисунок 3.3 – Декомпозиція обробки даних

3.2 Моделювання варіантів використання інформаційної системи виявлення мережевих атак

Наступним етапом є створення діаграми варіантів використання (use case diagram). В діаграмах варіантів використання зображуються взаємозв'язок між варіантами використання, які представляють функції системи, і діючими особами, які представляють людей або системи, що отримують або передають інформацію в дану систему. Діаграми варіантів використання відображають багато інформації про систему. Даний тип діаграм описує повну функціональність системи. Користувачі, аналітики, розробники, менеджери проектів, фахівці з контролю якості та всіх, кого цікавить система в цілому, можуть, переглядаючи діаграми варіантів використання, зрозуміти, що повинна робити система [18].

Щоб зрозуміти і правильно спроектувати майбутню систему, потрібно перш за все визначити, що вона буде робити, тобто, необхідно створення проекції, званої функціональної моделлю. Першим кроком при описі функціональності системи є моделювання вимог до неї.

Цілями аналізу і моделювання вимог є:

- досягнення розуміння між розробниками, замовниками і користувачами про те, що має робити ПС;
- обмеження системної функціональності;
- досягнення кращого розуміння розробниками поведінки ПС;
- створення бази для планування до розробки проекту;
- визначення призначеного для користувача інтерфейсу.

Результатом даного етапу є розробка діаграми варіантів використання для інформаційної системи (рисунок 3.4).

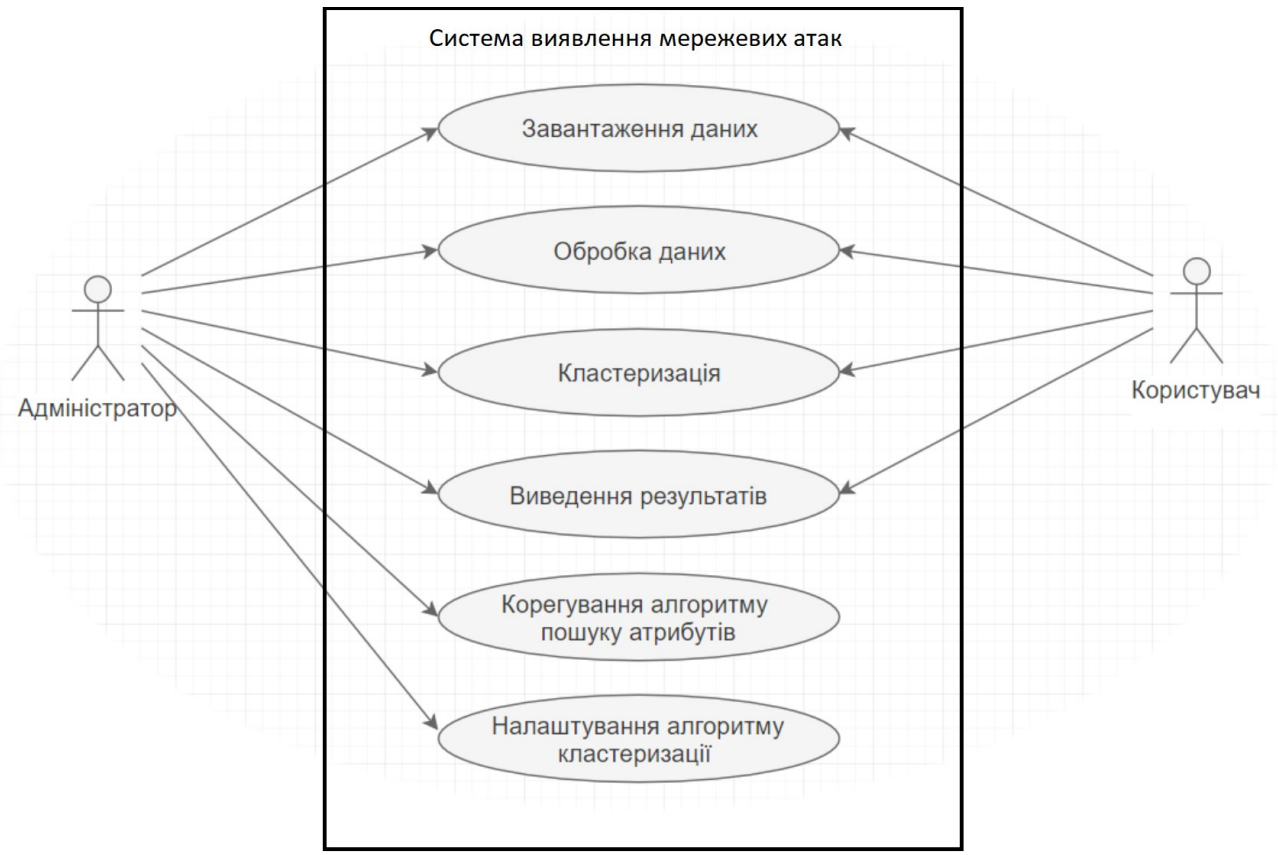


Рисунок 3.4 – Use case diagram

4 РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ВИЯВЛЕННЯ МЕРЕЖЕВИХ АТАК

4.1 Реалізація технологій кластеризації ситуацій

Вихідна вибірка KDD містить 25192 пакетів, з яких 13449 позначені як нормальна поведінка, а 11743 – аномальна.

Для визначення мережевих атак використовувалася карта Кохонена, що складається з двох кластерів (рисунок 4.1). За підсумками роботи алгоритму SOM в один кластер будуть відсортовані дані з імовірно нормальною поведінкою, в другий з імовірно аномальною поведінкою (рисунок 4.1).

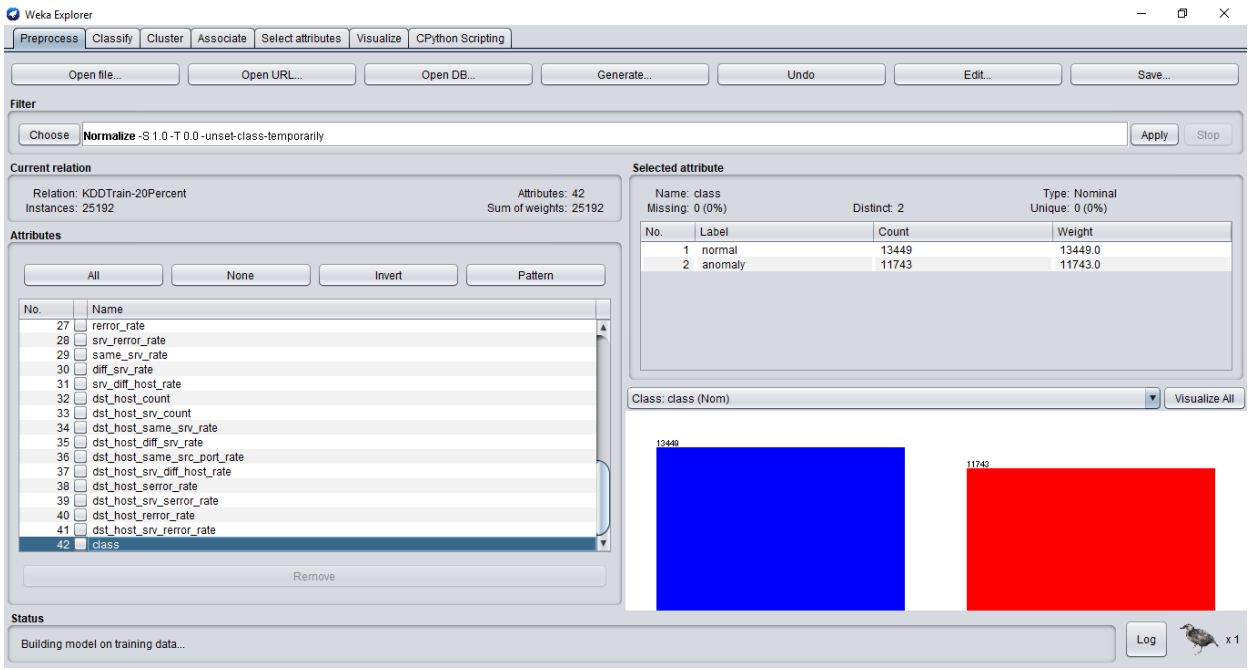


Рисунок 4.1 – Дані для кластеризації

Для вихідної вибірки без нормалізації параметрів (функція normalize Attribute) результати можна назвати незадовільними. Майже всі дані визначилися

як нормальні, аномальні дані були визначені вірно лише в 1,8% випадків. Результат робота алгоритму для даних без нормалізації наводяться в таблиці 4.1.

Таблиця 4.1 – Розподіл даних по кластерам без нормалізації даних

	Кластер 0 anomaly	Кластер 1 normal
Normal	181	13268
anomaly	210	11533

Нормалізація параметрів (рисунок 4.2) дає значний приріст до виявлення аномалій. Результати роботи алгоритму з нормалізацією наведені в таблиці 4.2.

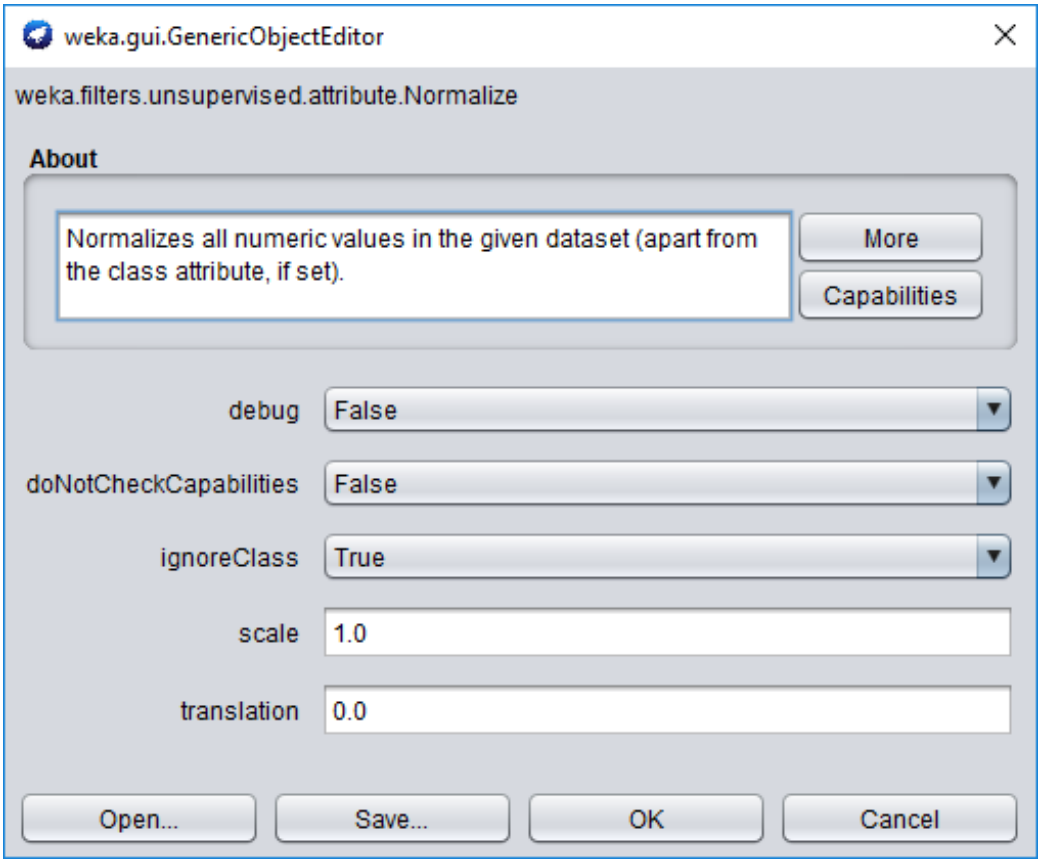


Рисунок 4.2 – Налаштування алгоритму нормалізації даних

Близько 59,5% атак були розпізнані коректно, що значно краще результатів без нормалізації даних. При цьому загальний відсоток правильної ідентифікації даних досягає 80,9%.

Таблиця 4.2 – Розподіл даних по кластерам з нормалізацією даних

	Кластер 0 anomaly	Кластер 1 normal
Normal	40	13409
anomaly	6989	4754

Для збільшення відсотка правильної ідентифікації даних використовувалось зменшення розмірності вихідної вибірки шляхом застосування алгоритмів вибору найбільш значущих параметрів, описаних раніше. Оптимальними параметрами при рівному відсотку загальної ідентифікації, вважається таке поєднання параметрів, при якому буде найбільш низький відсоток помилок першого роду (атака буде розпізнано як нормальна поведінка). Такий підхід з більшою ймовірністю збереже працездатність системи, ніж зменшення відсотка помилок другого роду (помилкові спрацьовування, при яких нормальна поведінка буде розпізнаватися як атака).

Результати ранжування атрибутів даних алгоритмом Gain Ratio Attribute Evaluator (рисунок 4.3) наведені в таблиці 4.3.

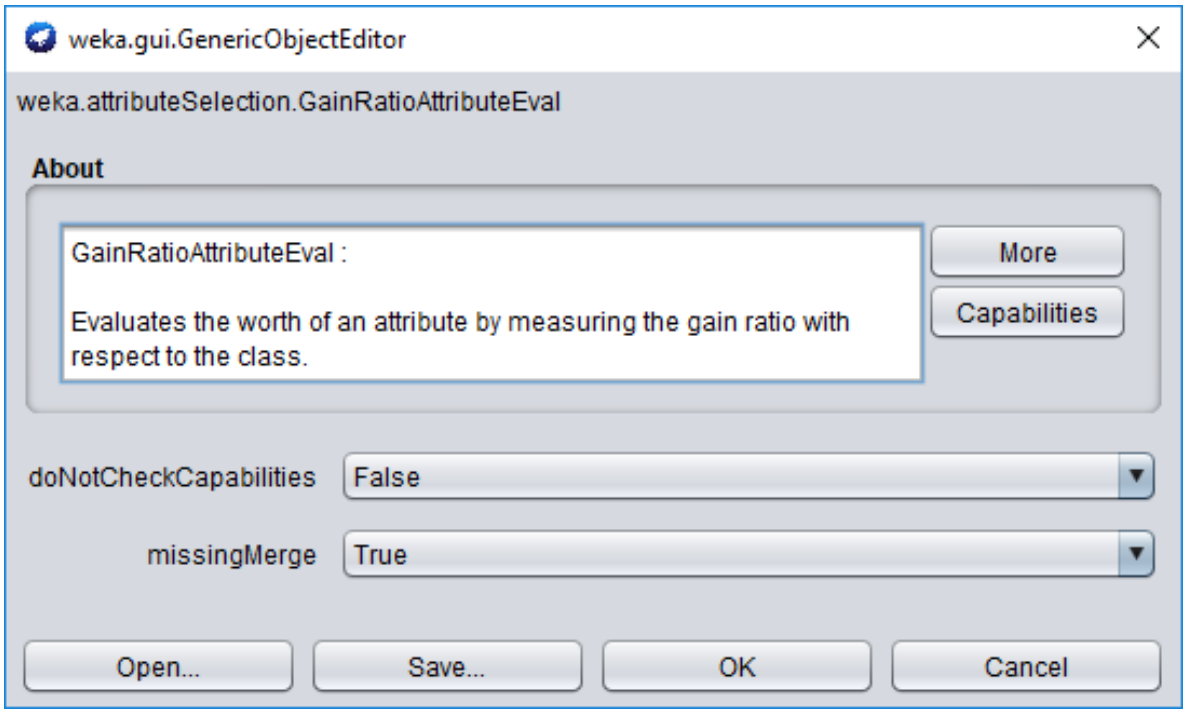


Рисунок 4.3 – Налаштування алгоритму Gain Ratio Attribute Evaluator

Таблиця 4.3 – Ранжування атрибутів алгоритмом Gain Ratio Attribute Evaluator

№ за значимістю	№ у вхідній вибірці	Атрибут
1	12	logged_in
2	26	srv_serror_rate
3	4	Flag
4	25	serror rate
5	39	dst host srv serror rate
6	6	dst_bytes
7	30	diff srv rate
8	38	dst host serror rate
9	5	src_bytes
10	29	same_srv_rate
11	3	Service
12	37	dst host srv diff host rate
13	34	dst host same srv rate
14	33	dst_host_STV_count
15	8	wrong_fragment
16	35	dst_host_diff_srv_rate
17	23	Count
18	31	srv diff host rate
19	41	dst host srv rerror rate
20	32	dst host count
21	28	srv_rerror_rate
22	27	rerror_rate
23	36	dst_host_same_src_port_rate
24	16	num_root
25	15	su_attempted
26	2	protocol_type
27	10	Hot
28	13	num_compromi sed
29	19	num access fi les
30	1	Duration
31	40	dst_host_rerror_rate
32	18	num shells
33	17	num_fi! e_creations
34	24	srv_count
35	14	root shell
36	22	is_guest_login
37	7	Land
38	11	num_failed_logins
39	20	num_outbound_cmds
40	9	Urgent
41	21	is_host_login

На основі результатів ранжирування було побудовано 41 вибірок і проведено тестування алгоритму SOM, наведено в таблиці 4.4.

Таблиця 4.4 – Результати розподілу даних по кластерам в залежності від кількості атрибутів, обраних алгоритмом Gain Ratio Attribute Evaluator

Атрибут	Нормальна поведінка	Помилкове виконання	Аномальна поведінка	Пропуск атаки	% Виявл. Атак	%Помилкове виконання	% Загальний результат
1	13449	0	0	11743	0,00	0	53,39
2	13364	85	6972	4771	59,37	0,63	80,72
3	13364	85	6972	4771	59,37	0,63	80,72
4	13289	160	6998	4745	59,59	1,19	80,53
5	13366	83	6994	4749	59,56	0,62	80,82
6	13366	83	6994	4749	59,56	0,62	80,82
7	13366	83	6994	4749	59,56	0,62	80,82
8	13375	74	6993	4750	59,55	0,55	80,85
9	13375	74	6993	4750	59,55	0,55	80,85
10	13409	40	6991	4752	59,53	0,3	80,98
11	13409	40	6991	4752	59,53	0,3	80,98
12	13405	44	6993	4750	59,55	0,33	80,97
13	13404	45	6993	4750	59,55	0,33	80,97
14	13404	45	6993	4750	59,55	0,33	80,97
15	13404	45	6993	4750	59,55	0,33	80,97
16	13405	44	6992	4751	59,54	0,33	80,97
17	13404	45	6993	4750	59,55	0,33	80,97
18	13409	40	6991	4752	59,53	0,3	80,98
19	13409	40	6991	4752	59,53	0,3	80,98
20	13409	40	6991	4752	59,53	0,3	80,98
21	13409	40	6991	4752	59,53	0,3	80,98
22	13409	40	6991	4752	59,53	0,3	80,98
23	13409	40	6991	4752	59,53	0,3	80,98
24	13409	40	6991	4752	59,53	0,3	80,98
25	13409	40	6991	4752	59,53	0,3	80,98
26	13409	40	6991	4752	59,53	0,3	80,98
27	13409	40	6991	4752	59,53	0,3	80,98
28	13409	40	6991	4752	59,53	0,3	80,98
29	13409	40	6991	4752	59,53	0,3	80,98
30	13409	40	6991	4752	59,53	0,3	80,98
31	13409	40	6991	4752	59,53	0,3	80,98
32	13409	40	6991	4752	59,53	0,3	80,98
33	13409	40	6991	4752	59,53	0,3	80,98
34	13409	40	6991	4752	59,53	0,3	80,98
35	13409	40	6991	4752	59,53	0,3	80,98
36	13409	40	6991	4752	59,53	0,3	80,98
37	13409	40	6991	4752	59,53	0,3	80,98
38	13409	40	6991	4752	59,53	0,3	80,98
39	13409	40	6991	4752	59,53	0,3	80,98
40	13409	40	6991	4752	59,53	0,3	80,98
41	13409	40	6991	4752	59,53	0,3	80,98

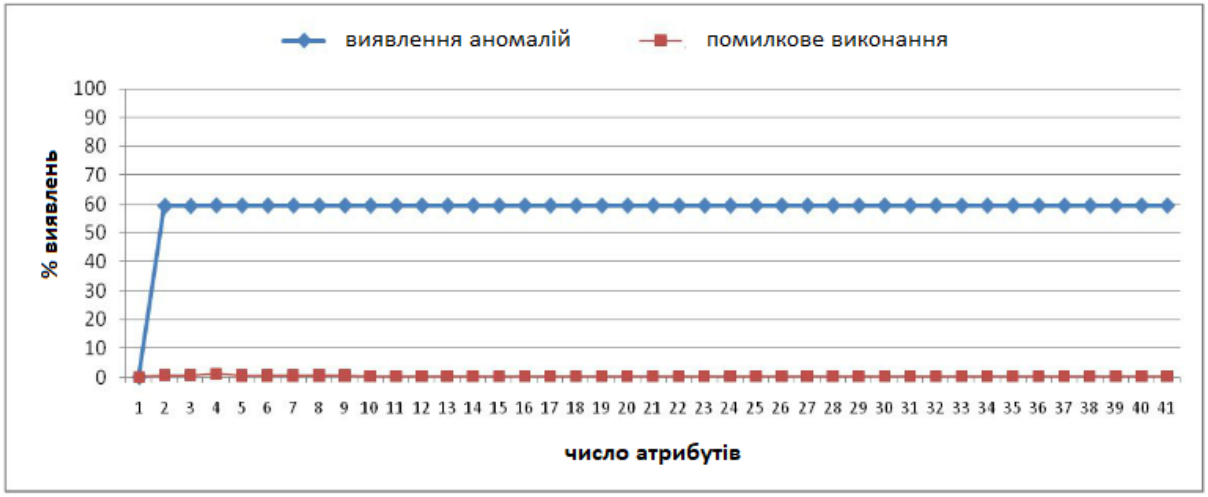


Рисунок 4.4 – Залежності відсотка виявлення мережевих атак і помилкових спрацьовувань від числа параметрів, обраних алгоритмом Gain Ratio Attribute Evaluator

В ході побудови систем самоорганізуючих карт на основі вибірок атрибутів, створених за результатами роботи алгоритму Gain Ratio Attribute Evaluator, отримати значущі покращення в виявленні мережевих атак не вдалося. Кращі дані показала вибірка з чотирьох атрибутів (табл. 4.5).

Таблиця 4.5 – Атрибути, що дали кращі результати, відібрані алгоритмом Gain Ratio Attribute Evaluator

№ у вихідній вибірці	Атрибут
4	Flag
12	logged in
25	serror rate
26	srv_serror_rate

Вищеописана вибірка дала наступні результати:

1) відсоток виявлення аномалій зріс з вихідних 59,53% до 59,6%;

2) відсоток помилкових спрацьовувань в цьому випадку збільшився з 0,3% до 1,2%;

3) загальний відсоток вірного визначення пакетів знизився з 80,98% до 80,53%.

Час роботи алгоритму побудови SOM вдалося значно знизити без втрати в ефективності виявлення мережевих атак (незначне зростання з 59,53% до 59,6%). Графік залежності загальних результатів визначення даних наведені на рисунку 4.5.

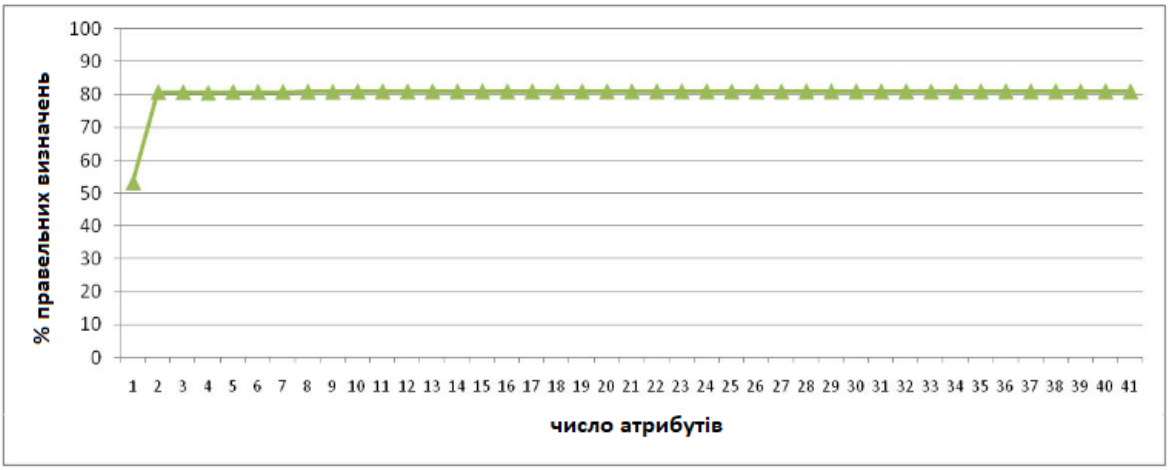


Рисунок 4.5 – Залежності вірного визначення даних алгоритмом SOM від числа параметрів, обраних алгоритмом Gain Ratio Attribute Evaluator

Результати ранжирування атрибутів в корені відрізняються від розглянутого раніше алгоритму Gain Ratio Attribute Evaluator. Цей факт дає підставу припускати, що результати роботи алгоритму SOM будуть відрізнятися. Результати ранжирування алгоритмом Information Gain Attribute Evaluator наведені в таблиці 4.6.

Таблиця 4.6 – Ранжування атрибутів алгоритмом Information Gain Attribute Evaluator

№ за значимістю	№ у вхідній вибірці	Атрибут
1	5	src_bytes
2	3	Service
3	6	dst_bytes
4	4	Nag
5	30	diff srv rate
6	29	same_srv_rate
7	33	dst_host_srv_count
8	34	dst_host_same_srv_rate
9	35	dst host diff srv rate
10	38	dst_host_serror_rate
11	12	logged_in
12	39	dst_host_srv_serror_rate
13	25	serror rate
14	23	Count
15	26	srv_serror_rate
16	37	dst_host_srv_diff_host_rate
17	32	dst host count
18	36	dst_host_samc_src_port_rate
19	31	STV_diff_host_rate
20	24	srv count
21	41	dst host srv rerror rate
22	2	protocol_type
23	27	rerror_rate
24	40	dst_host_rerror_rate
25	28	srv rerror rate
26	1	Duration
27	10	Hot
28	8	wrong_fragment
29	13	num_compromised
30	16	num root
31	19	num_access_files
32	22	is_guest_login
33	17	num file creations
34	15	su_attemptcd
35	14	root_shell
36	18	num_shells
37	7	Land
38	I	num_failed_logins
39	9	Urgent
40	20	num_outbound_cmds
41	21	is_host_login

На основі результатів ранжирування було побудовано 41 вибірок і проведено тестування алгоритму SOM. У деяких випадках вдалося отримати значний приріст до виявлення мережевих атак, а в інших процент виявлення

мережевих атак і помилкових спрацьовувань був таким же, як і у вихідній вибірці. Результати наводяться в таблиці 4.7.

Таблиця 4.7 – Результати розподілу даних по кластерам в залежності від кількості атрибутів, обраних алгоритмом Information Gain Attribute Evaluator

Атрибут	Нормальна поведінка	Помилкове виконання	Аномальна поведінка	Пропуск атаки	% Виявл. Атак	% Помилкове виконання	% Загальний результат
1	13449	0	1	11742	0,01	0,00	53,39
2	13449	0	1	11742	0,01	0,00	53,39
3	13447	2	5	11738	0,04	0,01	53,40
4	13447	2	5	11738	0,04	0,01	53,40
5	12999	450	617	11126	5,25	3,35	54,05
6	12848	601	8889	2854	75,70	4,47	86,29
7	12855	594	8929	2814	76,04	4,42	86,47
8	11351	2098	9944	1799	84,68	15,60	84,53
9	11346	2103	9940	1803	84,65	15,64	84,50
10	12005	1444	9814	1929	83,57	10,74	86,61
11	12005	1444	9814	1929	83,57	10,74	86,61
12	12005	1444	9814	1929	83,57	10,74	86,61
13	12906	543	9028	2715	76,88	4,04	87,07
14	13407	42	6998	4745	59,59	0,31	81,00
15	13409	40	7000	4743	59,61	0,30	81,01
16	13405	44	6992	4751	59,54	0,33	80,97
17	13405	44	6992	4751	59,54	0,33	80,97
18	13410	39	6991	4752	59,53	0,29	80,98
19	13409	40	6990	4753	59,52	0,30	80,97
20	13409	40	6990	4753	59,52	0,30	80,97
21	13409	40	6989	4754	59,52	0,30	80,97
22	13409	40	6989	4754	59,52	0,30	80,97
23	13409	40	6989	4754	59,52	0,30	80,97
24	13409	40	6989	4754	59,52	0,30	80,97
25	13409	40	6989	4754	59,52	0,30	80,97
26	13409	40	6989	4754	59,52	0,30	80,97
27	13409	40	6989	4754	59,52	0,30	80,97
28	13409	40	6989	4754	59,52	0,30	80,97
29	13409	40	6989	4754	59,52	0,30	80,97
30	13409	40	6989	4754	59,52	0,30	80,97
31	13409	40	6989	4754	59,52	0,30	80,97
32	13409	40	6989	4754	59,52	0,30	80,97
33	13409	40	6989	4754	59,52	0,30	80,97
34	13409	40	6989	4754	59,52	0,30	80,97
35	13409	40	6989	4754	59,52	0,30	80,97
36	13409	40	6989	4754	59,52	0,30	80,97
37	13409	40	6989	4754	59,52	0,30	80,97
38	13409	40	6989	4754	59,52	0,30	80,97
39	13409	40	6989	4754	59,52	0,30	80,97
40	13409	40	6989	4754	59,52	0,30	80,97
41	13409	40	6989	4754	59,52	0,30	80,97

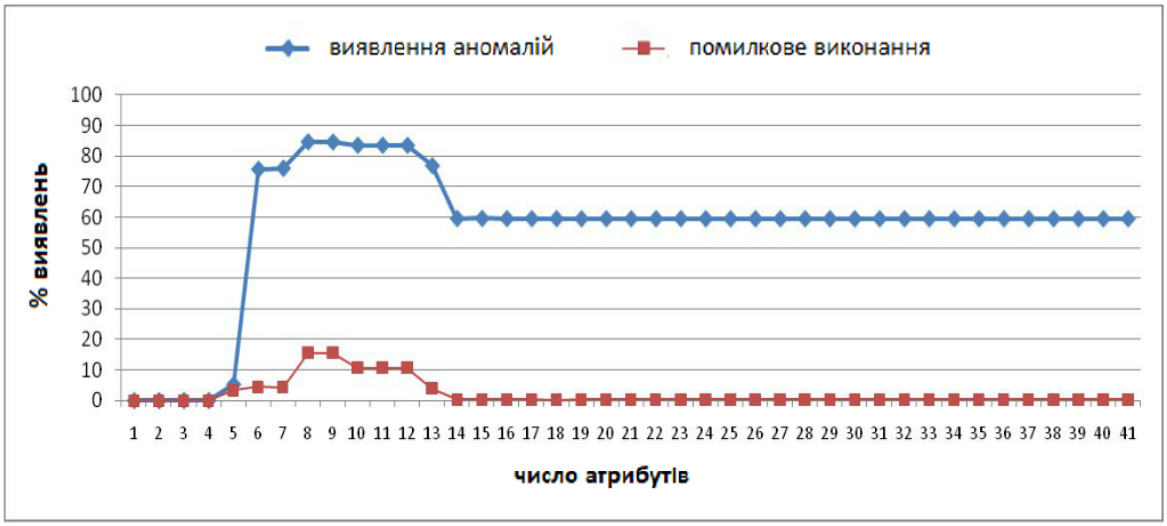


Рисунок 4.6 – Залежності відсотка виявлення мережевих атак і помилкових спрацьовувань від числа параметрів, обраних алгоритмом Information Gain Attribute Evaluator

Варто відзначити, що при збільшеному відсотку виявлення мережевих атак, вдалося значно скоротити час роботи алгоритму побудови SOM.

Кращі результати виявлення мережевих атак показала вибірка з восьми атрибутів (таблиця 4.8).

Таблиця 4.8 – Атрибути, що дали кращі результати, відібрані алгоритмом Information Gain Attribute Evaluatora

№ у вхідній вибірці	Атрибут
5	sre bytes
3	Service
6	dst bytes
4	Hag
30	diff srv rate
29	same srv rate
33	dst host srv count
34	dst host same srv rate

Дана вибірка дала наступні результати:

1) відсоток виявлення аномалій зріс з вихідних 59,53% до 84,68%;

2) відсоток помилкових спрацьовувань в цьому випадку збільшився з 0,3% до 15,6%;

3) загальний відсоток вірного визначення пакетів збільшився з 80,98% до 84,53%.

Загальні результати визначення даних для вибірок на основі алгоритму Information Gain Attribute Evaluator наведені на рисунку 4.7.

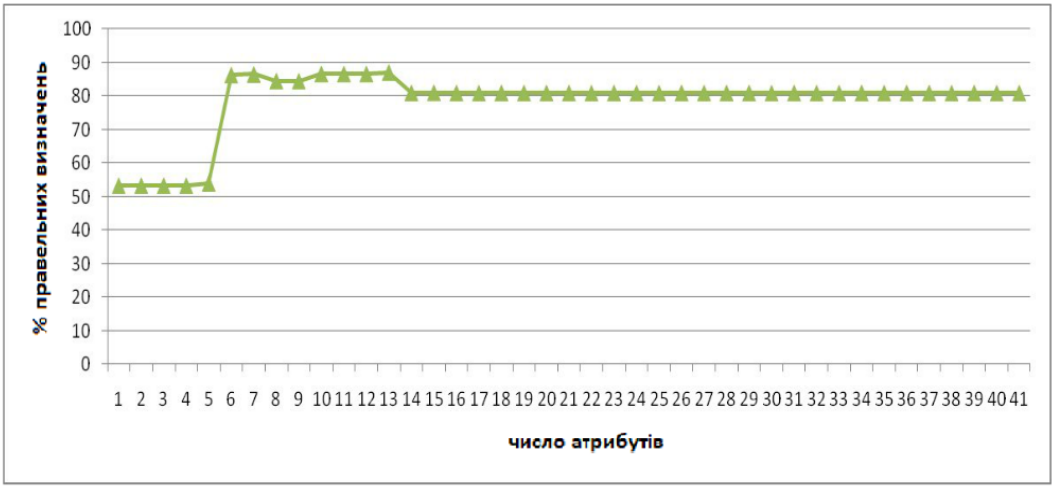


Рисунок 4.7 – Залежності вірного визначення даних алгоритмом SOM від числа параметрів, обраних алгоритмом Information Gain Attribute Evaluator

Проведення ранжування алгоритмом Correlation Attribute Evaluator.

На основі ранжируваних даних (таблиця 4.9) було побудовано 40 вибірок і проведено тестування алгоритму SOM. У деяких випадках вдалося отримати значний приріст до виявлення мережевих атак.

Таблиця 4.9 – Ранжування алгоритмом Correlation Attribute Evaluator

№ за значимістю	№ у вхідній вибірці	Атрибут
1	29	same srv rate
2	33	dst host srv count
3	34	dst host same srv rate
4	12	Loggedjn
5	39	dst host srv serror rate
6	4	Flag
7	38	dst host serror rate
8	25	serror rate
9	26	srv serror rate
10	23	Count
11	32	dst host count
12	3	Service
13	41	dst host srv rerror rate
14	27	rerror rate
15	40	dst host rerror rate
16	28	srv rerror rate
17	35	dst host diff srv rate
18	30	diff srv rate
19	31	srv diff host rate
20	8	wrong_fragment
21	36	dst host same src port rate
22	2	protocoI_type
23	37	dst host srv diff host rate
24	1	Duration
25	22	is_guest login
26	19	num access files
27	15	su_attempted
28	16	num root
29	13	num_compromised
30	14	root shell
31	17	num file creations
32	18	num shells
33	10	Hot
34	6	dst bytes
35	9	Urgent
36	5	src_bytes
37	24	srv count
38	7	Land
39	11	num_fai ledjogins
40	9	Urgent
41	21	is_host_login

В ході побудови системо самоорганізуючих карт на основі вибірок атрибутів (таблиця 4.10), створених за результатами роботи алгоритму Correlation Attribute Evaluator,

Таблиця 4.10 – Результати розподілу даних по кластерам в залежності від кількості атрибутів, обраних алгоритмом Correlation Attribute Evaluator

Атрибут	Нормальна поведінка	Помилкове виконання	Аномальна поведінка	Пропуск атаки	% Виявл. Атак	% Помилкове виконання	% Загальний результат
1	12850	599	8879	2864	75,61	4,45	86,25
2	12910	539	8926	2817	76,01	4,01	86,68
3	12105	1344	9751	1992	83,04	9,99	86,76
4	12105	1344	9751	1992	83,04	9,99	86,76
5	11343	2106	9949	1794	84,72	15,66	84,52
6	11343	2106	9949	1794	84,72	15,66	84,52
7	12995	454	9008	2735	76,71	3,38	87,34
8	13407	42	6998	4745	59,59	0,31	81,00
9	13404	45	6993	4750	59,55	0,33	80,97
10	13405	44	6992	4751	59,54	0,33	80,97
11	13410	39	6992	4751	59,54	0,29	80,99
12	13410	39	6992	4751	59,54	0,29	80,99
13	13410	39	6991	4752	59,53	0,29	80,98
14	13409	40	6991	4752	59,53	0,30	80,98
15	13410	39	6991	4752	59,53	0,29	80,98
16	13410	39	6991	4752	59,53	0,29	80,98
17	13410	39	6991	4752	59,53	0,29	80,98
18	13410	39	6991	4752	59,53	0,29	80,98
19	13410	39	6991	4752	59,53	0,29	80,98
20	13410	39	6991	4752	59,53	0,29	80,98
21	13409	40	6989	4754	59,52	0,30	80,97
22	13409	40	6989	4754	59,52	0,30	80,97
23	13409	40	6989	4754	59,52	0,30	80,97
24	13409	40	6989	4754	59,52	0,30	80,97
25	13409	40	6989	4754	59,52	0,30	80,97
26	13409	40	6989	4754	59,52	0,30	80,97
27	13409	40	6989	4754	59,52	0,30	80,97
28	13409	40	6989	4754	59,52	0,30	80,97
29	13409	40	6989	4754	59,52	0,30	80,97
30	13409	40	6989	4754	59,52	0,30	80,97
31	13409	40	6989	4754	59,52	0,30	80,97
32	13409	40	6989	4754	59,52	0,30	80,97
33	13409	40	6989	4754	59,52	0,30	80,97
34	13409	40	6989	4754	59,52	0,30	80,97
35	13409	40	6989	4754	59,52	0,30	80,97
36	13409	40	6989	4754	59,52	0,30	80,97

Продовження таблиці 4.10

37	13409	40	6989	4754	59,52	0,30	80,97
38	13409	40	6989	4754	59,52	0,30	80,97
39	13409	40	6989	4754	59,52	0,30	80,97
40	13409	40	6989	4754	59,52	0,30	80,97
41	13409	40	6989	4754	59,52	0,30	80,97

Вдалося отримати поліпшення в виявленні мережевих атак в декількох випадках (рис. 4.8). Кращі результати показала вибірка з п'яти атрибутів (таблиця 4.11).

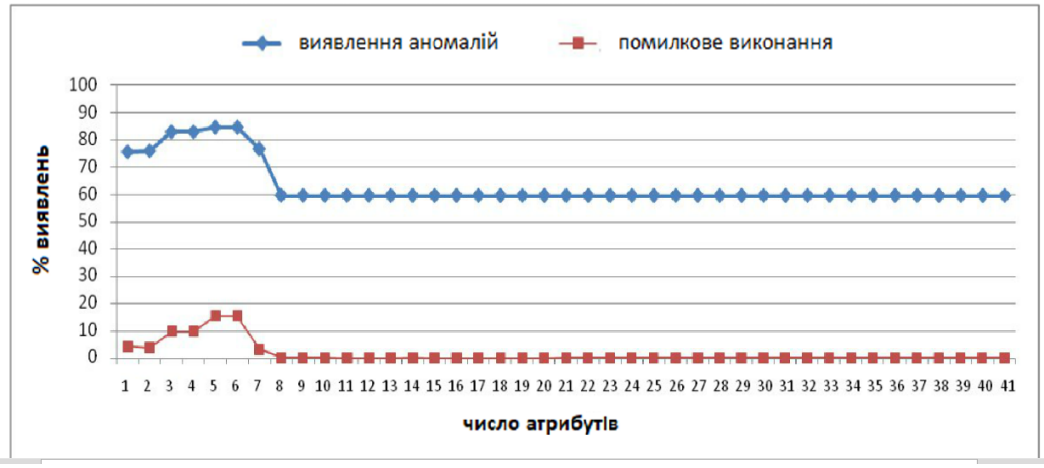


Рисунок 4.8 – Залежності відсотка виявлення мережевих атак і помилкових спрацьовувань про т числа параметрів, обраних алгоритмом Correlation Attribute Evaluator

Таблиця 4.11 – Атрибути, що дали кращі результати, відібрані алгоритмом Correlation Attribute Evaluator

№ у вхідній вибірці	Атрибут
29	same_srv_rate
33	dst host srv count
34	dst host same srv rate
12	logged_in
39	dst host srv serror rate

Дана вибірка дала наступні результати:

1) відсоток виявлення аномалій зріс з вихідних 59,53% до 84,72;

2) відсоток помилкових спрацьовувань в цьому випадку збільшився з 0,3% до 15,7%;

3) загальний відсоток вірного визначення пакетів збільшився з 80,98% до 84,52%.

В заключенні було проведення ранжування алгоритмом OneR Attribute Evaluator. Ранжування алгоритмом OneR Attribute Evaluator (таблиця 4.12) допомогло отримати найкращі результати в порівнянні з усіма вищеописаними алгоритмами.

Таблиця 4.12 – Ранжування алгоритмом OneR Attribute Evaluator

№ за значимістю	№ у вхідній вибірці	Атрибут
1	5	src_bytes
2	3	Service
3	6	dst_bytes
4	4	Flag
5	29	same srv rate
6	30	diff srv rate
7	34	dst host same srv rate
8	33	dst host srv count
9	35	dst host diff srv rate
10	12	loggedjn
11	23	Count
12	25	serror rate
13	38	dst host serror rate
14	39	dst host srv serror rate
15	26	srv serror rate
16	32	dst host count
17	36	dst_host_same_src_port_rate
18	37	dst host srv diff host rate
19	24	srv_count
20	31	srv diff host rate
21	41	dst host srv rerror rate
22	40	dst host rerror rate
23	27	rerror rate
24	28	srv rerror rate
25	2	protocoI_type
26	8	wrong_fragment
27	10	Hot
28	13	num_compromised

Продовження таблиці 4.12

29	1	Duration
30	14	root shell
31	20	num outbound cmds
32	22	is_guest_login
33	18	num_shells
34	19	num access files
35	21	is_host_Iogin
36	9	Urgent
37	15	su_attempted
38	16	num root
39	7	Land
40	17	num file creations
41	11	num_fai ledjogins

На основі ранжируваних даних було побудовано 40 вибірок і проведено тестування алгоритму SOM (таблиця 4.13).

Таблиця 4.13 – Результати розподілу даних по кластерам в залежності від кількості атрибутів, обраних алгоритмом OneR Attribute Evaluator

Атрибут	Нормальна поведінка	Помилкове виконання	Аномальна поведінка	Пропуск атаки	% Виявл. Атак	%Помилкове виконання	% Загальний результат
1	13449	0	1	11742	0,01	0,00	53,39
2	13449	0	1	11742	0,01	0,00	53,39
3	13447	2	5	11738	0,04	0,01	53,40
4	13447	2	5	11738	0,04	0,01	53,40
5	12850	599	8879	2864	75,61	4,45	86,25
6	12848	601	8889	2854	75,70	4,47	86,29
7	12898	551	8941	2802	76,14	4,10	86,69
8	11351	2098	9944	1799	84,68	15,60	84,53
9	11346	2103	9940	1803	84,65	15,64	84,50
10	11346	2103	9940	1803	84,65	15,64	84,50
11	11388	2061	9973	1770	84,93	15,32	84,79
12	12404	1045	9739	2004	82,93	7,77	87,90
13	12404	1045	9739	2004	82,93	7,77	87,90
14	13009	440	9030	2713	76,90	3,27	87,48
15	13409	40	7000	4743	59,61	0,30	81,01
16	13405	44	6992	4751	59,54	0,33	80,97
17	12404	1045	9739	2004	82,93	7,77	87,90
18	13409	40	6990	4753	59,52	0,30	80,97
19	13409	40	6990	4753	59,52	0,30	80,97
20	13409	40	6990	4753	59,52	0,30	80,97
21	13409	40	6989	4754	59,52	0,30	80,97
22	13409	40	6989	4754	59,52	0,30	80,97
23	13409	40	6989	4754	59,52	0,30	80,97

Продовження таблиці 4.13

24	13409	40	6989	4754	59,52	0,30	80,97
25	13409	40	6989	4754	59,52	0,30	80,97
26	13409	40	6989	4754	59,52	0,30	80,97
27	13409	40	6989	4754	59,52	0,30	80,97
28	13409	40	6989	4754	59,52	0,30	80,97
29	13409	40	6989	4754	59,52	0,30	80,97
30	13409	40	6989	4754	59,52	0,30	80,97
31	13409	40	6989	4754	59,52	0,30	80,97
32	13409	40	6989	4754	59,52	0,30	80,97
33	13409	40	6989	4754	59,52	0,30	80,97
34	13409	40	6989	4754	59,52	0,30	80,97
35	13409	40	6989	4754	59,52	0,30	80,97
36	13409	40	6989	4754	59,52	0,30	80,97
37	13409	40	6989	4754	59,52	0,30	80,97
38	13409	40	6989	4754	59,52	0,30	80,97
39	13409	40	6989	4754	59,52	0,30	80,97
40	13409	40	6989	4754	59,52	0,30	80,97
41	13409	40	6989	4754	59,52	0,30	80,97

Зріс відсоток як виявлення мережевих атак, так і загального результату визначення даних (рис. 4.9).

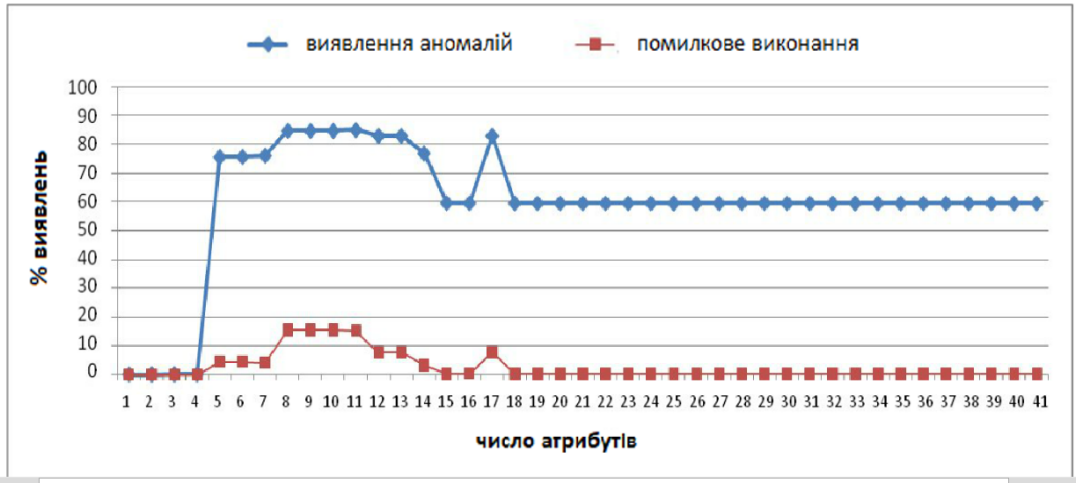


Рисунок 4.9 – Залежності відсотка виявлення мережевих атак і помилкових спрацьовувань від числа параметрів, обраних алгоритмом OneR Attribute Evaluator

Кращі результати показала вибірка з одинадцяти атрибутів (таблиця 4.14).

Таблиця 4.14 – Атрибути, що дали кращі результати, відібрані алгоритмом OneR Attribute Evaluator

№ у вхідній вибірці	Атрибут
5	src_bytes
3	Service
6	dst_bytes
4	Flag
29	same srv rale
30	diff srv rate
34	dst host same srv rate
33	dst host srv count
35	dst host diff srv rate
12	logged_in
23	Count

Дана вибірка дала наступні результати:

1) відсоток виявлення аномалій зріс з вихідних 59,53% до 84,93%;

2) відсоток помилкових спрацьовувань в цьому випадку збільшився з 0,3% до 15,32%;

3) загальний відсоток вірного визначення пакетів збільшився з 80,98% до 84,93%.

4.2 Аналіз результатів

Для оцінювання результатів роботи необхідно порівняти результати виявлення мережевих атак, які отримано використовуючи обрану вибірку, з результатами аналогічних досліджень.

Авторами статті [14] за результатами дослідження запропоновано три вибірки атрибутів з 12, 13 і 17 атрибутів. Вибірка з 12 атрибутів була отримана при ранжируванні методом незалежних компонент, з 13 атрибутів була отримана при вибірці методом головних компонент і їх 17 атрибутів - за допомогою лінійного дискримінантного аналізу.

Результати виявлення мережевих атак на основі атрибутів зі статті [14].

Використовуючи запропоновані вибірки, були отримані наступні результати:

Таблиця 4.15 – Результати виявлення мережевих атак (на основі атрибутів з прикладу[14])

Алгоритм	Нормальна поведінка	Помилкове виконання	Аномальна поведінка	Пропуск атаки	% Виявл. атак	% Помилкове виконання	% Загальний результат
12	12547	1924	9819	1924	83,62	13,30	85,32
13	13363	86	6961	4782	59,28	0,64	80,68
17	13337	112	6999	4744	59,60	0,83	80,72

Результати виявлення, отримані в ході тестування запропонованих вибірок атрибутів, виявилися нижче, ніж отримані з використанням алгоритму OneR в ході роботи.

В статті [15] дані вибірки атрибутів не тільки для класифікації трафіку на «нормальний» або «аномальний», а й вибірки атрибутів для кращого виявлення конкретних атак.

Результати виявлення мережевих атак, отримані в ході тестування запропонованих вибірок атрибутів, виявилися нижче, ніж отримані з використанням алгоритму OneR в ході роботи:

Таблиця 4.16 – Результати виявлення мережевих атак на основі атрибутів статті [15]

Алгоритм	Нормальна поведінка	Помилкове виконання	Аномальна поведінка	Пропуск Атаки	% Виявл. атак	% Помилкове виконання	% Загальний результат
Best First	13124	325	8855	2888	75,41	2,42	87,25
Greedy Stepwise	13411	38	6991	4752	59,53	0,28	80,99
PSO Search	13124	325	8855	2888	75,41	2,42	87,25
Tabu Search	13124	325	8855	2888	75,41	2,42	87,25
OneR	12404	1045	9739	2004	59,53	0,28	80,99

Результати тестування алгоритму SOM на вибірці KDD для DOS атак з атрибутами, запропонованими в статті, виявилися нижчими, ніж отримані з використанням алгоритму OneR в ході роботи (таблиця 4.17).

Таблиця 4.17 – Результати виявлення DOS-атаки алгоритмом SOM

Алгоритм	Нормальна поведінка	Помилкове виконання	Аномальна поведінка	Пропуск атаки	% Виявл. атак	% Помилкове виконання	% Загальний результат
Best First	4243	34	1262	427	74,719	0,8	92,27
Greedy	4264	13	1262	427	74,72	0,30	92,62
PSO Search	4265	12	1262	427	74,72	0,28	92,64
Tabu Search	4243	34	1262	427	74,72	0,79	92,27
OneR	4140	137	1511	178	89,46	3,20	94,72

Аналогічно у випадку з probe-атаки результати тестування алгоритму SOM на вибірці KDD для probe-атак з атрибутами, запропонованими в статті, виявилися нижчими, ніж отримані з використанням алгоритму OneR в ході роботи (таблиця 4.18).

Таблиця 4.18 – Результати виявлення PROBE-атаки алгоритмом SOM

Алгоритм	Нормальна поведінка	Помилкове виконання	Аномальна поведінка	Пропуск атаки	% Виявл. атак	% Помилкове виконання	% Загальний результат
Best Frist	4362	447	3474	2290	60,27	9,30	74,11
Greedy	4492	317	3108	2656	53,92	6,59	71,88
PSO Search	4492	317	3108	2656	53,92	6,59	71,88
Tabu Search	4492	317	3108	2656	53,92	6,59	71,88
OneR	4362	447	3474	2290	60,27	9,30	74,11

В результаті аналізу даних. зібраних під час роботи з вибірками даних на основі ранжирування атрибутів, було встановлено, що можна значно прискорити роботу алгоритму SOM без зниження якості виявлення мережевих атак. Крім

цього, можна підвищити відсоток виявлення аномалій шляхом вибору найбільш значущих параметрів.

В ході експериментів вдалося підвищити відсоток виявлення мережевих атак з вихідних 59.53% до 84.93% за допомогою алгоритму вибору атрибутів OneR Attribute Evaluator. У порівнянні з рекомендованими вибірками атрибутів в подібних наукових роботах вдалося отримати більший відсоток як виявлення атак у вигляді аномалій, так і конкретних видів атак (DOS, probe-атаки).

ВИСНОВКИ

Мережеві атаки стають все більшою загрозою для економіки України. Виділяють активні та пасивні, за цілями впливу можуть бути атаки: атаки розвідки, атаки отримання доступу, атаки відмови в обслуговуванні. Вектори що описують з’єднання повинні імітувати 4 види атак: DoS – атаки перевантажує ресурси або пам'ять комп'ютера, U2R – може використовувати деяку уразливість для підвищення прав до системи, R2L – використовує деяку уразливість, щоб отримати права доступу користувача, probe-атаки – спроба обходу засобів контролю безпеки.

В загальному випадку для виявлення кібератак необхідність сформувати вектор, що містить 41 атрибут. Задача виявлення мережевих атак може бути сформована як задача кластеризації.

Зручним інструментом вирішення завдання задачі кластеризації проблемних ситуацій наявності кібератак, може бути програмне забезпечення WEKA. Досліджено алгоритми формування навчальної вибірки. Приведені комп’ютерні експерименти, дозволяють визначити ефективність алгоритмів визначення множини необхідних атрибутів та виконання ранжування у порядку зростання ефективності:

1. OneR;
2. Correlation;
3. Information Gain;
4. Gain Ratio.

Оптимальним алгоритмом можна вважати алгоритм OneR (відсоток виявлення аномалій 84.93%, відсоток вірного визначення даних 84.78%). За більшістю показників алгоритм конкурентоспроможним для вирішення задачі кластеризації.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Платонов, В. В. Програмно-апаратні засоби захисту інформації. - М .: Видавничий центр «Академія», 2013. - 336 с.

2. Боршевніков А. Е. Мережеві атаки. Види. Способи боротьби. – Уфа. 2011.- С. 8-13.

3. Гамаюнов Д.Ю. Виявлення комп'ютерних атак на основі аналізу поведінки об'єктів: автореф. дис. ... канд. фіз.-мат. наук: 05.13.11. - М., 2007. - 89 с

4. А. А. Браницький, І. В. Котенко, Аналіз і класифікація методів виявлення мережевих атак, Тр. СПІІРАН, 2016, випуск 45, 207-244

5. Системи і методи виявлення вторгнень: сучасний стан і напрями вдосконалення [Електронний ресурс]. URL: http://citforum.ni/security/intemet/ids_overview/#3

6. Мережеві аномалії [Електронний ресурс]. URL: http://nag.ru/articles/revievvs/15588/setevyie-anomalii.html

7. Хайкін, Р. Нейронні мережі: повний курс, 2-е видання. Пер. з англ. / Р.Хайкін. - М .: Видавничий дім «Вільямс», 2006. - 1104 с.

8. Горбаченко В.І. мережі і карти Кохонена. URL: http: //gorbachenko.self- organization.ru/articles/Self-organizing_map.pdf

9. Самоорганізуючі карти. Пер. з англ. / Т. Кохонен - М .: БИНОМ. Лабораторія знань 2012. - 655 с .

10. NSL-KDD dataset [Електронний ресурс]. URL: http://wwvv.unb.ca/cic/research/datasets/nsl.html

11. Weka: Data Mining Software in Java [Електронний ресурс]. URL: http://www.cs.waikato.ac.nz/ml/weka

12. WEKA Classification Algorithms [Електронний ресурс]. URL: http://wekaclassalgos.sourceforge.net/

13. WEKA: attribute selection [Електронний ресурс]. URL: http://weka.sourceforge.net/doc.dev/weka/attributeSelection/package-summary.html/

14. Venkatachalam. V. Performance comparison of intrusion detection system classifiers using various feature re daction techniques // SELVAN S Erode Sengunthar Engineering College. 31 March 2015. – P. 19

15. Madbouly A. Relevant Feature Selection Model Using Data Mining for Intrusion Detection System / Madbouly, A. // Gody, A. // Barakat, T International Journal of Engineering Trends and Technology (IJETT) - Volume 9 Number 10 - Mar 2014. – P. 12

16. Керівний документ. Методологія функціонального моделювання IDEF0. Розроблено Науково-дослідним Центром CALS - технологій «Прикладна Логістика». Держстандарт. - М .: ІПК Видавництво стандартів, 2000. - 75 с.

17. Вендров А.М. CASE-технології. Сучасні методи і засоби проектування інформаційних систем.. - М .: Фінанси і статистика, 1998. - 98 с.

18. Забезпечення комплексної безпеки підприємств: проблеми та рішення: збірник тез доповідей IV Міжнародну науково-практичної конференції. Рязан. держ. радіотехн. ун-т; Рязань, 2016 - 144 с.

19. Басалаєва, Ю.С. Вибір інструментів Data Mining для аналізу результатів дистанційної освіти / Ю.С. Басалаєва // Сучасні матеріали, техніка та технологія. - 2015. - № 2. - С. 22 - 25.

20. V.J. Nirmal and D.I.G. Amalarethinam, “Parallel Implementation of Big Data Pre-Processing Algorithms for Sentiment Analysis of Social Networking Data,” International Journal of Fuzzy Mathematical Archive, Vol. 6, No. 2, pp.149-159, 2015.

21. K. M. Ting, T. Washio, J. R. Wells, F. T. Liu, S. Aryal, “DEMass: a new density estimator for big data,” Knowledge & Information Systems, Vol. 35, pp. 493–524, 2013.

22. 5 інструментів Data Mining: порівняльний аналіз. [Електронний ресурс]. URL: http://datareview.info/article/5-instrumentov-data-mining-sravnitelnyiy-analiz/

23. Гремякіна О. А. Вибір платформи інтелектуального аналізу даних для застосування в академічних цілях // Молодий вчений. - 2015. - №22. - С. 26-29.

24. RapidMiner Studio Manual by RapidMiner, 2014, URL: https://rapidminer.com/wp-content/uploads/2014/10/RapidMiner-v6-user-manual.pdf

25. Організація комп'ютерних мереж: конспект лекцій/ Л.М. Олешенко ; КПІ ім. Ігоря Сікорського. - Електронні текстові дані. - Київ : КПІім. Ігоря Сікорського, 2018. - 225 с.

26. Baddar S.A.-H., Merlo A., Migliardi M. Anomaly Detection in Computer Networks: A State-of-the-Art Review // Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications. 2014. vol. 5. no. 4. pp. 29–64.

27. Hall M., Witten I., Frank E. Data Mining: Practical Machine Learning Tools and Tech-niques. Morgan Kaufmann Publishers, 2011. 629 p.

28. K. Alrawashdeh and C. Purdy, ``Toward an online anomaly intrusion detection system based on deep learning," in Proc. 15th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA), Anaheim, CA, USA, Feb. 2017, pp. 195-200.

29. A. A. Diro and N. Chilamkurti, ``Distributed attack detection scheme using deep learning approach for Internet of Things," Future Gener. Comput. Syst., vol. 82, pp. 761-768, May 2018.

30. Garg, T.; Khurana, S.S. Comparison of Classification Techniques for Intrusion Detection Dataset Using WEKA. In Proceedings of the International Conference on Recent Advances and Innovations in Engineering, Jaipur, India, 9–11 May 20Bharath, R.R., Thanigaivel, K., Alfahath, A., Prasanth, T.: Feature extraction based dynamic recommendation for analogous users. Int. J. Comput. Sci. Inf. Technol. 5(2), 1358–1362 (2014)

31. Холявка Є. П.; Метод виявлення мережевих атак в комп’ютеризованих системах управління: наукова робота, Хмельницький національний університет. – Хмельницький, 2019, [Електронний ресурс]. URL: http://konkurs.khnu.km.ua/wp-content/uploads/sites/25/2019/04/DP3_Eugen.pdf

ДОДАТОК А. ТЕХНІЧНЕ ЗАВДАННЯ

ТЕХНІЧНЕ ЗАВДАННЯ
на розробку інформаційної системи «Інформаційна технологія виявлення мережевих атак в критичних інформаційних системах»

Суми 2019

Призначення й мета створення інформаційної системи

Призначення інформаційної системи

Математична модель і інформаційна технологія виявлення атак.

Мета створення інформаційної системи

Розробити інформаційну технологію виявлення мережевих атак в критичних інформаційних системах та обгрунтувати можливість використання її на практиці.

Цільова аудиторія

Метод може бути застосовано в системах підтримки прийняття рішень з питань кібербезпеки автоматизованих систем і забезпечити виявлення мережевих атак в критичних системах.

Вимоги до інформаційної системи

Вимоги до інформаційної системи в цілому

2.1.1 Вимоги до структури й функціонування інформаційної системи

На відміну від існуючих статистичних методів та методів орієнтованих на використання технології навчання з вчителем, запропонована технологія орієнтована на використанні самоорганізуючих карт Кохонена.

2.1.2 Вимоги до персоналу

Для експлуатації системи виявлення аномалій, від персоналу не вимагається спеціальних технічних навичок, або програмних продуктів, за винятком загальних навичок роботи з персональним комп'ютером і знання технологій.

2.1.3 Вимоги до збереження інформації

Резервне копіювання потрібно робити регулярно, повинно бути не менше двох типів даних на різних носіях.

2.1.4 Вимоги до розмежування доступу

Дана технологія є загальнодоступною.

Користувачі не розділяються на групи, кожен користувач має рівні права доступу.

ДОДАТОК Б. ПЛАНУВАННЯ РОБІТ

Планування змісту робіт

Структурна декомпозиція робіт – орієнтоване на конкретні завдання "дерево" робіт. Декомпозиція служить графічним відображенням усього процесу кінцевих цілей проекту, для розподілу великої кількості інформації за рівнями управління. Дана структура являє собою систему розбиття проекту на компоненти: комплекси, роботи, групи робіт(рис Б1.1.).

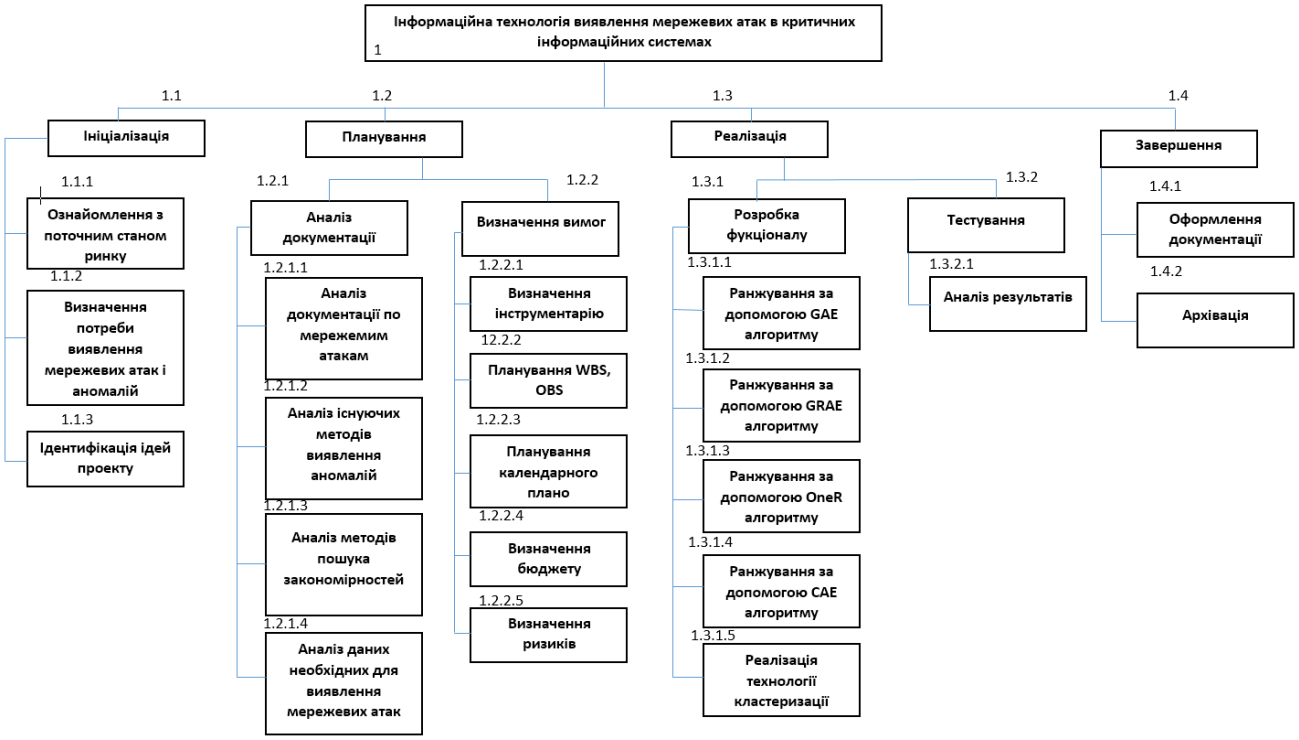


Рисунок Б1.1 – WBS структура системи

Планування структури виконавця для впровадження готового проекту

Декомпозиція організаційної структури (Organisational Breakdown Structure - OBS) – структурна декомпозиція організації проекту, призначена для співвіднесення пакетів робіт з організаційними одиницями. OBS є графічної діаграмою організаційної структури проекту.

Організаційна структура проекту - це інструмент планування взаємодії за проектом і планування повноважень команди управління проектом, який встановлює зв'язки між етапом, певним в структурі робіт проекту, і організаційною структурою компанії.

OBS структура представлена на рисунку Б1.2.

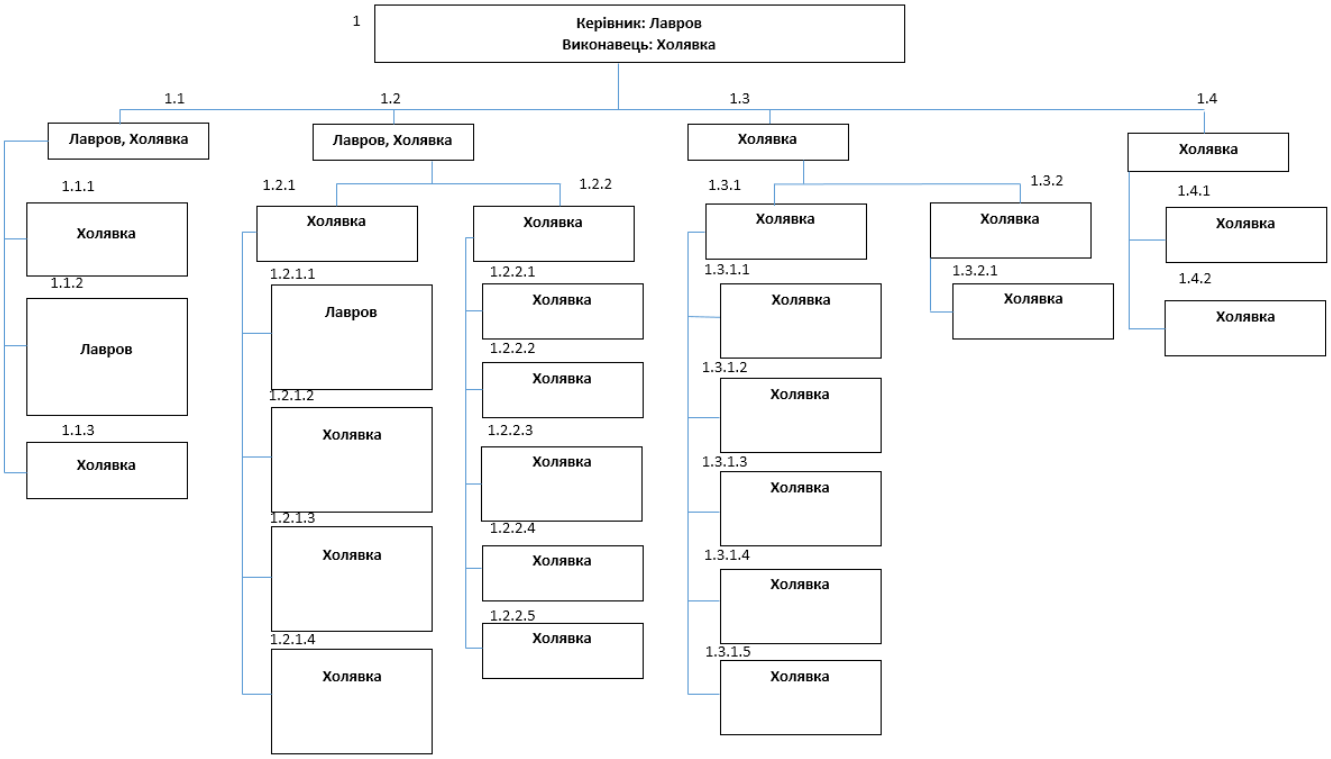


Рисунок Б1.2 – OBS структура

Побудова матриці відповідальності

При складанні матриці відповідальності проекту використовують, наприклад, методику RACI. Дана методика RACI – це зручний і наочний засіб планування відповідальності членів команди розробників при виконанні кожного етапу завдань проекту.

Таблиця Б1.1 – Матриця відповідальності

WBS\OBS	Холявка	Лавров
Ознайомлення з поточним станом ринку		
Визначення потреби виявлення мережевих атак і аномалій		
Ідентифікація ідей проекту		
Аналіз документації по мережевим атакам		
Аналіз існуючих методів виявлення аномалій		
Аналіз методів пошуку закономірностей		
Аналіз даних необхідних для виявлення мережевих атак		
Визначення інструментарію		
Планування WBS, OBS		
Планування календарного плану		
Визначення бюджету		
Визначення ризиків		
Ранжування за допомогою GAE алгоритму		
Ранжування за допомогою GRAE алгоритму		
Ранжування за допомогою OneR алгоритму		
Ранжування за допомогою CAE алгоритму		

Продовження таблиці Б1.1

Аналіз результатів		
Оформлення документації		
Архівація		

Розробка PDM мережі

Головне призначення PDM-систем - управління інформацією та полегшення доступу до даних про виріб протягом усього його життєвого циклу. Позитивний ефект досягається завдяки можливості об'єднати всі дані про виріб в єдину логічну систему. В результаті такого об'єднання все, хто бере участь в розробці вироби, отримують розподілений авторизований доступ до проектної інформації та управління процесами проектування. Найбільш поширені завдання, які можна вирішити за допомогою PDM-систем, такі:

- створення архіву креслень і іншої технічної документації;
- автоматизація внесення змін в конфігурацію виробу.

Методи оцінювання та аналізу програми (PERT) – це метод мережевого аналізу, який орієнтований на події і використовується для оцінювання тривалості проекту при високому ступені невизначеності оцінками тривалості окремих робіт. Методологія PERT застосовує методи критичного шляху для точного оцінювання середнього значення тривалості, це дозволяє приблизно оцінити можливий час виконання робіт, це рекомендується для аналізу проектів з істотними ризиками (рис. Б1.3.).

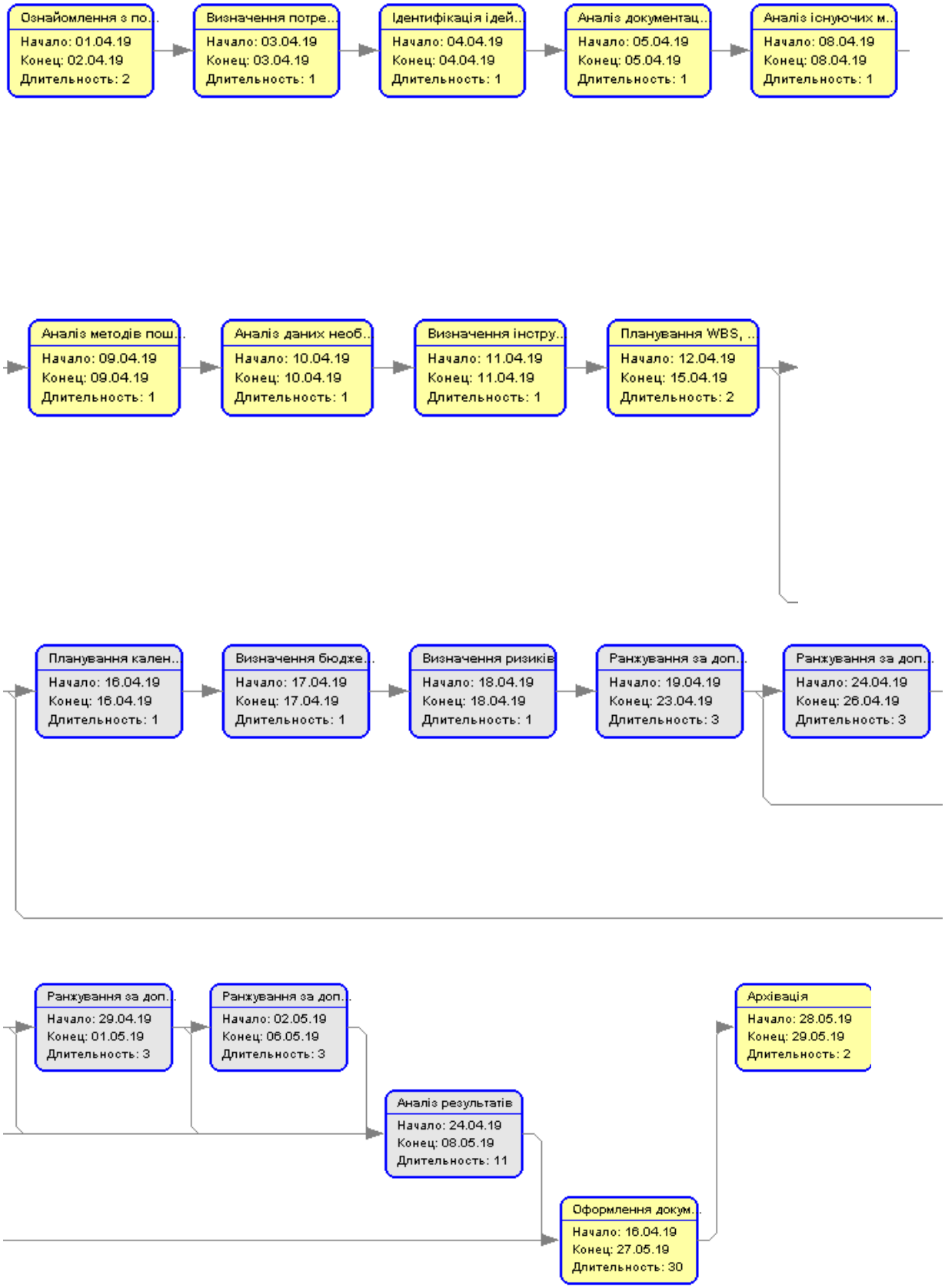


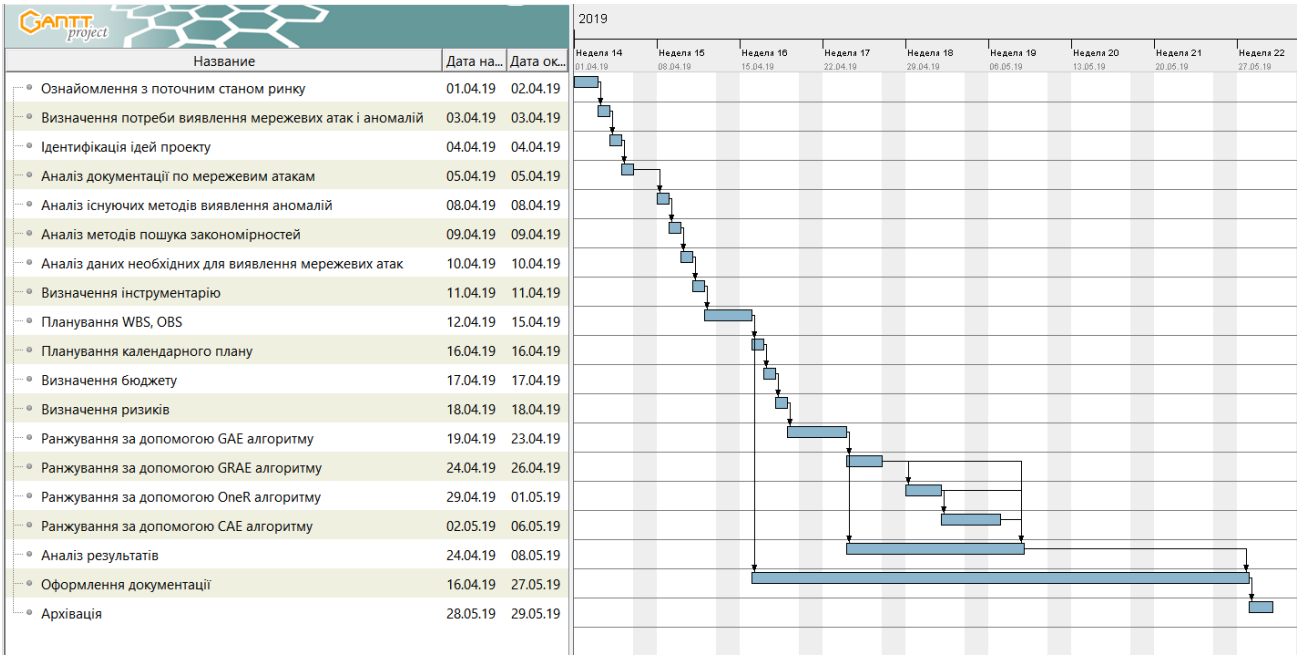
Рисунок Б1.3 – PERT діаграма

Побудова календарного графіку виконання IT-проекту

Діаграма Ганта - це спеціальний інструмент для управління та планування завданнями, який розробив американський інженер Генрі Гант (Henry Gantt). Це інструмент у вигляді горизонтальних смуг, розташованих між двома осями: списком завдань по вертикалі і датами по горизонталі.

Діаграма відображає не тільки завдання, але і послідовність їх виконання. Це дозволяє контролювати всі етапи і виконувати все вчасно.

Стане в нагоді діаграма Ганта і для презентації етапів створення проекту. Особа, яка замовила побачить обсяги та терміни робіт і зрозуміє деякі нюанси поставленого завдання (рис Б1.4).



Название	Дата на...	Дата ок...
Ознайомлення з поточним станом ринку	01.04.19	02.04.19
Визначення потреби виявлення мережевих атак і аномалій	03.04.19	03.04.19
Ідентифікація ідей проекту	04.04.19	04.04.19
Аналіз документації по мережевим атакам	05.04.19	05.04.19
Аналіз існуючих методів виявлення аномалій	08.04.19	08.04.19
Аналіз методів пошука закономірностей	09.04.19	09.04.19
Аналіз даних необхідних для виявлення мережевих атак	10.04.19	10.04.19
Визначення інструментарію	11.04.19	11.04.19
Планування WBS, OBS	12.04.19	15.04.19
Планування календарного плану	16.04.19	16.04.19
Визначення бюджету	17.04.19	17.04.19
Визначення ризиків	18.04.19	18.04.19
Ранжування за допомогою GAE алгоритму	19.04.19	23.04.19
Ранжування за допомогою GRAE алгоритму	24.04.19	26.04.19
Ранжування за допомогою OneR алгоритму	29.04.19	01.05.19
Ранжування за допомогою CAE алгоритму	02.05.19	06.05.19
Аналіз результатів	24.04.19	08.05.19
Оформлення документації	16.04.19	27.05.19
Архівація	28.05.19	29.05.19

Рисунок Б1.4 – Діаграма Ганта

ДОДАТОК В. ОПИС ДАНИХ KDD

Таблиця В1 – Список файлів, що містяться в наборі NSL KDD

№	Ім'я файлу	Опис
1	KDDTrain + .ARFF	Повний NSL-KDD набір даних для навчання в ARFF форматі
2	KDDTrain + .TXT	Повний NSL-KDD набір даних для навчання, що включає типи атак, в форматі TXT
3	KDDTrain + _20Percent.ARFF	Вибірка 20% даних з KDDTrain + .ARFF
4	KDDTrain + _20Percent.TXT	Вибірка 20% даних з KDDTrain + .TXT
5	KDDTest + .ARFF	Повний NSL-KDD набір даних для тестування в ARFF форматі
6	KDDTest + .TXT	Повний NSL-KDD набір даних для тестування, що включає типи атак, в форматі TXT
7	KDDTest-21.ARFF	Вибірка з KDDTest + .ARFF яка містить неправильно класифіковані під час дослідження дані
8	KDDTest-21.TXT	Вибірка з KDDTest + .TXT яка містить неправильно класифіковані під час дослідження дані

Таблиця В2 – Опис атрибутів, що використовуються для виявлення атак

№	Назва параметра	Опис	Приклад
Основні параметри кожного мережевого з'єднання			
1	Duration	Час тривалості підключення	0
2	protocol_type	Протокол, який використовується при підключенні	TCP
3	Service	Мережева служба, яка використовується підключенням	ftp_data
4	src_bytes	Кількість відправлених байт за одне з'єднання	491
5	dst_bytes	Кількість прийнятих байт за одне з'єднання	0
6	Flag	Статус з'єднання - нормальне або з Помилкою	SF
7	Land	Якщо ip-адреси хоста джерела і призначення рівні, і аналогічна ситуація з портами, то параметр приймає значення 1, інакше 0.	0
8	wrong_fragment	Загальна кількість невірних фрагментів за цепідключення	0
9	Urgent	Кількість urgent-пакетів в цьому Підключенні	0
Параметри, пов'язані з контентом кожного мережевого з'єднання			
10	Hot	Кількість hot-індикаторів, наприклад таких як: вхід в системні директорії, створення програм, виконання програм.	0
11	num_failed_logins	Кількість невдалих спроб входу	0
12	logged_in	Логін статус. 1 - якщо успішно увійшли в систему, інакше 0	0
13	num_compromised	Число скомпрометованих станів	0
14	root_shell	1, якщо root-права отримані, інакше 0	0
15	su_attempted	1, якщо su root-права отримані, інакше 0	0

Продовження таблиці В2

16	num_root	Число root-доступів	0
17	num_file_creations	Число операцій по створенню файлів під час з'єднання	0
18	num_shells	Число викликів shell-оболонки	0
19	num_access_files	Число операцій з отримання контролю доступу до файлів	0
20	num_outbound_cmds	Число вихідних команд в FTP-сесії	0
21	is_hot_login	1, якщо логін належить hot-листу тобто якщо є root або адміністратором, інакше 0	0
22	is_guest_login	1, якщо логін є гостьовим, інакше 0	0
Параметри, пов'язані з тимчасовими характеристиками кожного мережевого з'єднання			
23	Count	Кількість підключень до одного і того ж хосту призначення за останні дві секунди	2
24	serror_rate	Відсоток з'єднань з хостом з count з SYN- помилками	0
25	rerror_rate	Відсоток з'єднань з хостом з count з REJ- помилками	0
26	same_srv_rate	Відсоток з'єднань з хостом з count використовують одні і ті ж служби	1
27	diff_srv_rate	Відсоток з'єднань з хостом використовуючи різні служби	0
28	srv_count	Число з'єднань з однієї і тієї ж службою за останні дві секунди.	2
29	srv_serror_rate	Відсоток з'єднань з SYN-помилками при з'єднанні по службі з srv_count	0
30	srv_rerror_rate	Відсоток з'єднань з REJ-помилками при з'єднанні по службі з srv_count	0

Продовження таблиці В2

31	srv_diff_host_rate	Відсоток з'єднань з різними хостами при з'єднанні по службі з srv_count	0
Параметри, пов'язані з характеристиками хоста кожного мережевого з'єднання			
32	dst_host_count	Число з'єднань з тим же самим ip-адресою хоста призначення	150
33	dst_host_srv_count	Число з'єднань з тим же самим номером порту	25
34	dst_host_same_srv_rate	Відсоток з'єднань по тій же самій службі під час з'єднання з ip з dst_host_count	0.17
35	dst_host_diff_srv_rate	Відсоток з'єднань по різних служб у час з'єднання по ip з dst_host_count	0.03
36	dst_host_same_src_port_ rate	Відсоток з'єднань до того ж самому хосту приймача під час зв'язок через порт з dst_host_srv_count	0.17
37	dst_host_srv_diff_host_rate	Відсоток з'єднань з різними хостами приймачами під час зв'язок через порт з dst_host_srv_count	0
38	dst_host_serror_rate	Відсоток з'єднань з хостом з dst_host_count з SYN-помилками	0
39	dst_host_srv_serror_rate	Відсоток з'єднань з SYN-помилками при з'єднанні по службі з dst_host_srv_count	0
40	dst_host_rerror_rate	Відсоток з'єднань з хостом з dst_host_count з REJ -помилки	0.05
41	dst_host_srv_rerror_rate	Відсоток з'єднань з REJ -помилки при з'єднанні по службі з dst_host_srv_count	0

ДОДАТОК Г. ПРИКЛАД ДАНИХ NSL KDD

```
@relation 'KDDTrain-20Percent'
@attribute 'duration' real
@attribute 'protocol_type' {'tcp','udp', 'icmp'}
@attribute 'service' {'aol', 'auth', 'bgp', 'courier',
'csnet_ns', 'ctf', 'daytime', 'discard', 'domain', 'domain_u',
'echo', 'eco_i', 'ecr_i', 'efs', 'exec', 'finger', 'ftp',
'ftp_data', 'gopher', 'harvest', 'hostnames', 'http', 'http_
2784', 'http_443', 'http_8001', 'imap4', 'IRC', 'iso_tsap',
'klogin', 'kshell', 'ldap', 'link', 'login', 'mtp', 'name',
'netbios_dgm', 'netbios_ns', 'netbios_ssn', 'netstat', 'nnsp',
'nntp', 'ntp_u', 'other', 'pm_dump', 'pop_2', 'pop_3',
'printer', 'private', 'red_i', 'remote_job', 'rje', 'shell',
'smtp', 'sql_net', 'ssh', 'sunrpc', 'supdup', 'systat',
'telnet', 'tftp_u', 'tim_i', 'time', 'urh_i', 'urp_i', 'uucp',
'uucp_path', 'vmnet', 'whois', 'X11', 'Z39_50'}
@attribute 'flag' { 'OTH', 'REJ', 'RSTO', 'RSTOS0', 'RSTR',
'S0', 'S1', 'S2', 'S3', 'SF', 'SH' }
@attribute 'src_bytes' real
@attribute 'dst_bytes' real
@attribute 'land' {'0', '1'}
@attribute 'wrong_fragment' real
@attribute 'urgent' real
@attribute 'hot' real
@attribute 'num_failed_logins' real
@attribute 'logged_in' {'0', '1'}
@attribute 'num_compromised' real
@attribute 'root_shell' real
@attribute 'su_attempted' real
@attribute 'num_root' real
@attribute 'num_file_creations' real
@attribute 'num_shells' real
@attribute 'num_access_files' real
@attribute 'num_outbound_cmds' real
@attribute 'is_host_login' {'0', '1'}
@attribute 'is_guest_login' {'0', '1'}
@attribute 'count' real
@attribute 'srv_count' real
@attribute 'serror_rate' real
@attribute 'srv_serror_rate' real
@attribute 'rerror_rate' real
@attribute 'srv_rerror_rate' real
@attribute 'same_srv_rate' real
@attribute 'diff_srv_rate' real
@attribute 'srv_diff_host_rate' real
@attribute 'dst_host_count' real
@attribute 'dst_host_srv_count' real
@attribute 'dst_host_same_srv_rate' real
@attribute 'dst_host_diff_srv_rate' real
@attribute 'dst_host_same_src_port_rate' real
@attribute 'dst_host_srv_diff_host_rate' real
@attribute 'dst_host_serror_rate' real
@attribute 'dst_host_srv_serror_rate' real
@attribute 'dst_host_rerror_rate' real
@attribute 'dst_host_srv_rerror_rate' real
@attribute 'class' {'normal', 'anomaly'}
```

```
@data
0,tcp,ftp_data,SF,491,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.
00,0.00,0.00,0.00,1.00,0.00,0.00,150,25,0.17,0.03,0.17,0.00,0.
00,0.00,0.05,0.00,normal
0,udp,other,SF,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,13,1,0.00
,0.00,0.00,0.00,0.08,0.15,0.00,255,1,0.00,0.60,0.88,0.00,0.00,
0.00,0.00,0.00,normal
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,123,6,1.0
0,1.00,0.00,0.00,0.05,0.07,0.00,255,26,0.10,0.05,0.00,0.00,1.0
0,1.00,0.00,0.00,anomaly
0,tcp,http,SF,232,8153,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,5,5,0.2
0,0.20,0.00,0.00,1.00,0.00,0.00,30,255,1.00,0.00,0.03,0.04,0.0
3,0.01,0.00,0.01,normal
0,tcp,http,SF,199,420,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,30,32,0.
00,0.00,0.00,0.00,1.00,0.00,0.09,255,255,1.00,0.00,0.00,0.00,0
.00,0.00,0.00,0.00,normal
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,121,19,0
.00,0.00,1.00,1.00,0.16,0.06,0.00,255,19,0.07,0.07,0.00,0.00,0
.00,0.00,1.00,1.00,anomaly
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,166,9,1.0
0,1.00,0.00,0.00,0.05,0.06,0.00,255,9,0.04,0.05,0.00,0.00,1.00
,1.00,0.00,0.00,anomaly
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,117,16,1.
00,1.00,0.00,0.00,0.14,0.06,0.00,255,15,0.06,0.07,0.00,0.00,1.
00,1.00,0.00,0.00,anomaly
0,tcp,remote_job,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,270,23
,1.00,1.00,0.00,0.00,0.09,0.05,0.00,255,23,0.09,0.05,0.00,0.00
,1.00,1.00,0.00,0.00,anomaly
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,133,8,1.0
0,1.00,0.00,0.00,0.06,0.06,0.00,255,13,0.05,0.06,0.00,0.00,1.0
0,1.00,0.00,0.00,anomaly
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,205,12,0
.00,0.00,1.00,1.00,0.06,0.06,0.00,255,12,0.05,0.07,0.00,0.00,0
.00,0.00,1.00,1.00,anomaly
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,199,3,1.0
0,1.00,0.00,0.00,0.02,0.06,0.00,255,13,0.05,0.07,0.00,0.00,1.0
0,1.00,0.00,0.00,anomaly
0,tcp,http,SF,287,2251,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,3,7,0.0
0,0.00,0.00,0.00,1.00,0.00,0.43,8,219,1.00,0.00,0.12,0.03,0.00
,0.00,0.00,0.00,normal
0,tcp,ftp_data,SF,334,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,2,2,0.
00,0.00,0.00,0.00,1.00,0.00,0.00,2,20,1.00,0.00,1.00,0.20,0.00
,0.00,0.00,0.00,anomaly
0,tcp,name,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,233,1,1.00,1
.00,0.00,0.00,0.00,0.06,0.00,255,1,0.00,0.07,0.00,0.00,1.00,1.
00,0.00,0.00,anomaly
0,tcp,netbios_ns,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,96,16,
1.00,1.00,0.00,0.00,0.17,0.05,0.00,255,2,0.01,0.06,0.00,0.00,1
.00,1.00,0.00,0.00,anomaly
0,tcp,http,SF,300,13788,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,9,0.
00,0.11,0.00,0.00,1.00,0.00,0.22,91,255,1.00,0.00,0.01,0.02,0.
00,0.00,0.00,0.00,normal
0,icmp,eco_i,SF,18,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,
0.00,0.00,0.00,1.00,0.00,0.00,1,16,1.00,0.00,1.00,1.00,0.00,0.
00,0.00,0.00,anomaly
```

```
0,tcp,http,SF,343,1178,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,9,10,0.
00,0.00,0.00,0.00,1.00,0.00,0.20,157,255,1.00,0.00,0.01,0.04,0
.00,0.00,0.00,0.00,normal
0,tcp,mtp,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,223,23,1.00,1
.00,0.00,0.00,0.10,0.05,0.00,255,23,0.09,0.05,0.00,0.00,1.00,1
.00,0.00,0.00,anomaly
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,280,17,1.
00,1.00,0.00,0.00,0.06,0.05,0.00,238,17,0.07,0.06,0.00,0.00,0.
99,1.00,0.00,0.00,anomaly
0,tcp,http,SF,253,11905,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,10,0
.00,0.00,0.00,0.00,1.00,0.00,0.20,87,255,1.00,0.00,0.01,0.02,0
.00,0.00,0.00,0.00,normal
5607,udp,other,SF,147,105,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,
0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,1,0.00,0.85,1.00,0.00,0
.00,0.00,0.00,0.00,normal
0,tcp,mtp,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,248,2,1.00,1.
00,0.00,0.00,0.01,0.06,0.00,255,2,0.01,0.06,0.00,0.00,1.00,1.0
0,0.00,0.00,anomaly
507,tcp,telnet,SF,437,14421,0,0,0,0,0,1,3,0,0,0,0,0,1,0,0,0,1,
1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,25,0.10,0.05,0.00,0.0
0,0.53,0.00,0.02,0.16,normal
0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,279,7,1.0
0,1.00,0.00,0.00,0.03,0.06,0.00,255,13,0.05,0.07,0.00,0.00,1.0
0,1.00,0.00,0.00,anomaly
0,tcp,http,SF,227,6588,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,5,22,0.
00,0.00,0.00,0.00,1.00,0.00,0.18,43,255,1.00,0.00,0.02,0.14,0.
00,0.00,0.56,0.57,normal
0,tcp,http,SF,215,10499,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,14,14,
0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,255,1.00,0.00,0.00,0.00
,0.00,0.00,0.00,0.00,normal
0,tcp,http,SF,241,1400,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,33,33,0
.00,0.00,0.03,0.03,1.00,0.00,0.00,255,255,1.00,0.00,0.00,0.00,
0.00,0.00,0.00,0.00,normal
0,icmp,eco_i,SF,8,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,12,0.00,
0.00,0.00,0.00,1.00,0.00,1.00,1,53,1.00,0.00,1.00,0.51,0.00,0.
00,0.00,0.00,anomaly
0,tcp,finger,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,57,16,1.00
,1.00,0.00,0.00,0.28,0.07,0.00,255,59,0.23,0.04,0.01,0.00,1.00
,1.00,0.00,0.00,anomaly
0,tcp,http,SF,303,555,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,9,9,0.00
,0.00,0.00,0.00,1.00,0.00,0.00,9,255,1.00,0.00,0.11,0.01,0.00,
0.00,0.00,0.00,normal
0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,1,0.00
,0.00,1.00,1.00,0.50,1.00,0.00,255,1,0.00,0.31,0.28,0.00,0.00,
0.00,0.29,1.00,anomaly
0,udp,domain_u,SF,45,45,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,181,18
1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,250,0.98,0.01,0.00,0.
00,0.00,0.00,0.00,0.00,normal
1,udp,private,SF,105,147,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0
.00,0.00,0.00,0.00,1.00,0.00,0.00,41,5,0.12,0.05,0.05,0.00,0.0
0,0.00,0.00,0.00,normal
0,udp,domain_u,SF,43,43,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,122,20
2,0.00,0.00,0.00,0.00,1.00,0.00,0.01,255,255,1.00,0.00,0.01,0.
00,0.00,0.00,0.00,0.00,normal
```

ДОДАТОК Ґ. АЛГОРИТМИ ВИБОРУ АТРИБУТІВ, ЩО ПОКАЗАЛИ НАЙБІЛЬШИЙ ВІДСОТОК ВИЯВЛЕННЯ МЕРЕЖЕВИХ АТАК

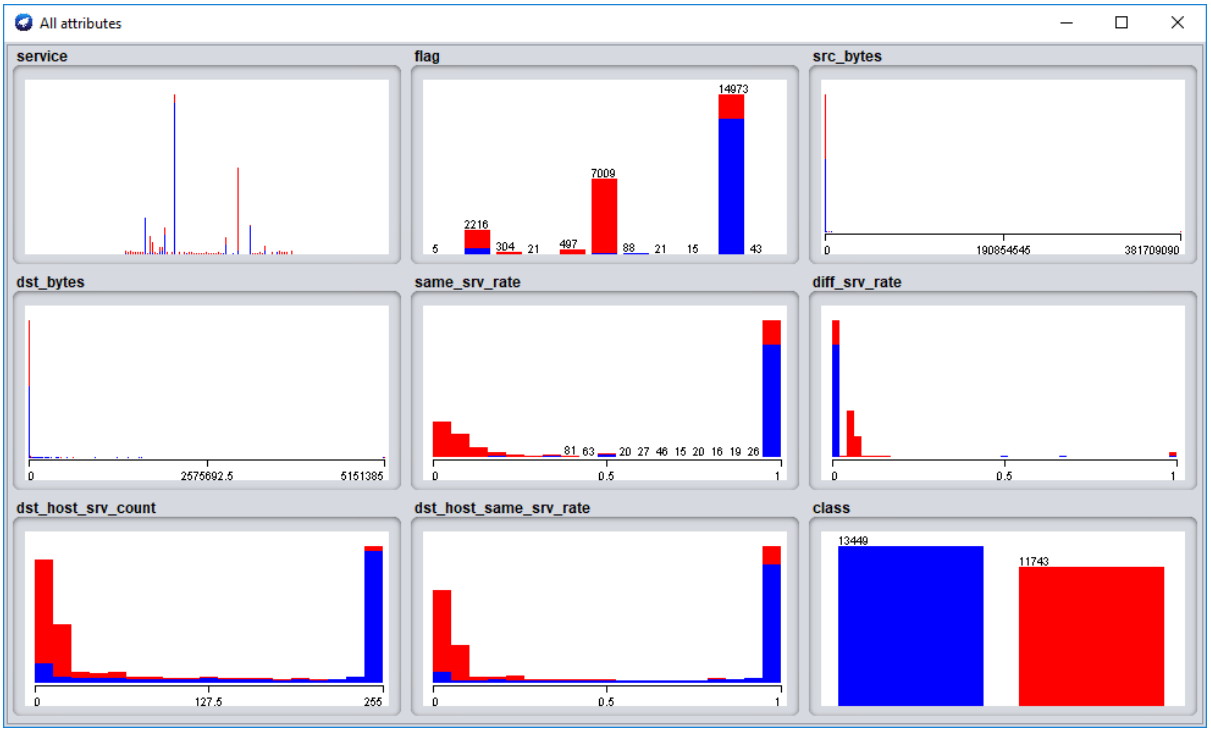


Рисунок Ґ1 – Візуалізація результатів розподілених алгоритмом Information Gain Attribute Evaluator

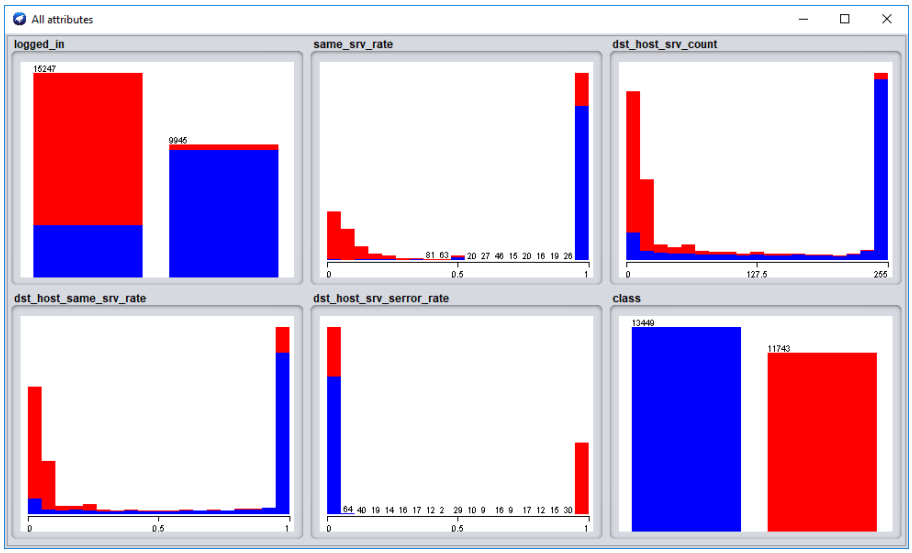


Рисунок Ґ2 – Візуалізація результатів розподілених алгоритмом Correlation Attribute Evaluator

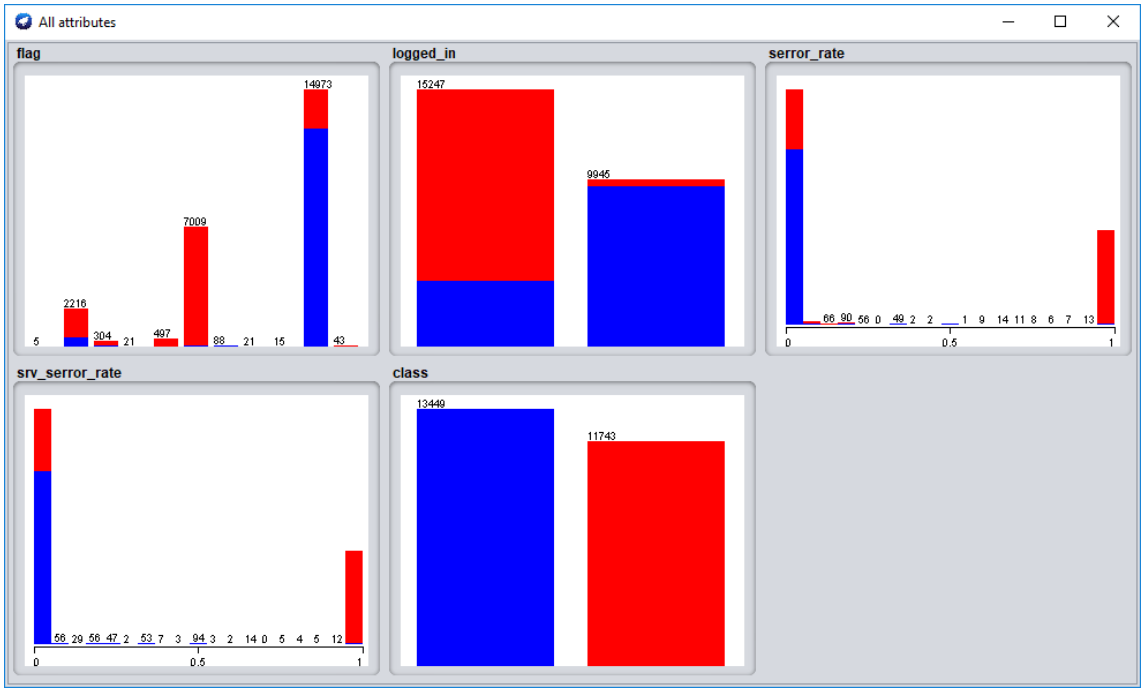


Рисунок Ґ3 – Візуалізація результатів розподілених алгоритмом Gain Ratio Attribute Evaluator

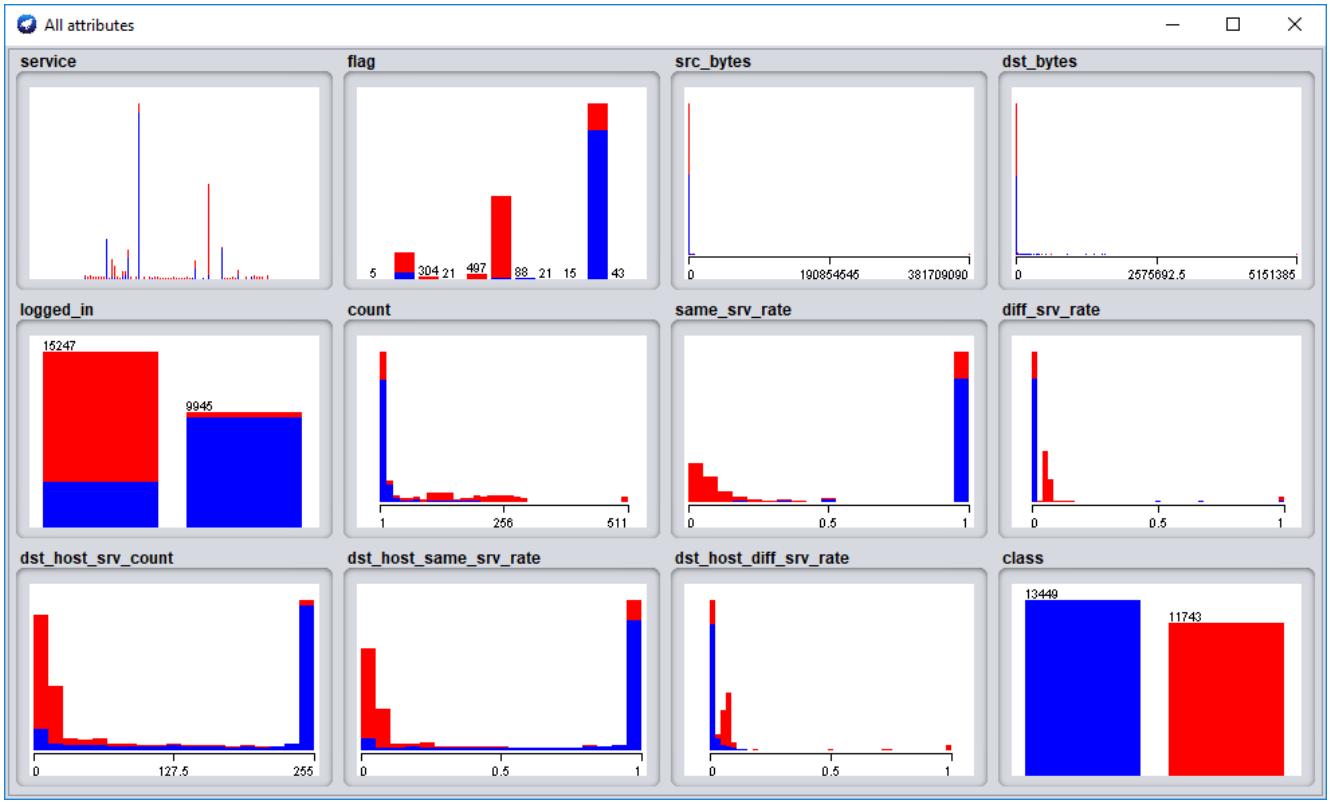


Рисунок Ґ4 – Візуалізація результатів розподілених алгоритмом OneR Attribute Evaluator

ДОДАТОК Д. КОПІЇ ПУБЛІКАЦІЙ

Копії публікацій за темою:
«Інформаційна технологія виявлення мережевих атак в критичних інформаційних системах»


МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАТИКА, МАТЕМАТИКА, АВТОМАТИКА
ІМА - 2019
МАТЕРІАЛИ
та програма
НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ
(Суми, 23-26 квітня 2019 року)
Суми,
Сумський державний університет
2019

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

ІНФОРМАТИКА, МАТЕМАТИКА, АВТОМАТИКА

ІМА :: 2019

МАТЕРІАЛИ
та програма

НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

(Суми, 23–26 квітня 2019 року)

Суми
Сумський державний університет
2019

Інформаційна технологія виявлення мережевих атак в критичних інформаційних системах

Холявка Є.П., *студент*; Лавров Е.А., *професор*
Сумський державний університет, м. Суми, Україна

Актуальність.

Однією з найбільших загроз економіці і державності України є загрози, пов'язані з кібербезпекою. В останні роки збільшується кількість вірусів, що розповсюджуються в мережах, та мережевих атак. Незважаючи на велику кількість наукових робіт, задача виявлення мережевих атак в критичних інформаційних системах, де збитки можуть бути загрозливими, вирішена не до кінця.

Постановка задачі.

Розробити універсальну технологію для виявлення мережевих атак в критичних інформаційних системах управління та обґрунтувати можливість використання її на практиці.

Результати.

Розглянуто активні та пасивні за цілями впливу атаки:

– атаки розвідки;

– атаки отримання доступу;

– атаки відмови в обслуговуванні.

Вектори що описують з'єднання повинні імітувати 4 види атак: DoS – атаки (відмова в обслуговуванні), U2R – використовують уразливість для підвищення прав (root) доступу до системи, R2L – використовують уразливість, щоб отримати неавторизований доступ до системи, probe-атаки – спроба обходу засобів контролю безпеки. Система виявлення атак побудована і навчена на даних, що охоплюють і моделюють атаки і спроби вторгнення. Одним таким загальновідомих і відкритих наборів даних є NSL KDD, який був зібраний з ініціативи Управління перспективних дослідницьких проектів Міністерства оборони США (DARPA).

Обґрунтовано доцільність використання методу кластеризації, основаної на картах Кохонена. В якості моделюючого середовища використовувався ППП WEKA.Для вихідної вибірки, без нормалізації,

результати виявились незадовільними. Нормалізація параметрів (функція NormalizeAttributes – перетворює текстові дані в числових, а все числові дані наводяться до виду [0 ; 1]) дає значний приріст до виявлення аномалій, близько 59,5% атак розпізнано коректно. Для збільшення відсотка правильної ідентифікації даних, запропоновано прийом зменшення розмірності вибірки, шляхом застосування алгоритмів вибору найбільш значущих параметрів.

Досліджено 4 алгоритми формування навчальної вибірки:

– GainRatioAttributeEvaluator;

– InformationGainAttributeEvaluator;

– CorrelationAttributeEvaluator;

– OneRule.

Приведені комп'ютерні експерименти дозволили визначити ефективність алгоритмів визначення множини необхідних атрибутів та виконання ранжування.

Характеристика ефективності алгоритмів наведені в табл.1:

Таблиця 1 – Характеристика ефективності алгоритмів

Алгоритм	% виявлення аномалій	% помилкового спрацювання	% загальний результат
GainRatio	59,6	1,2	50,53
InformationGain	84,68	15,6	84,53
Correlation	84,72	15,7	84,52
OneR	84,93	15,32	84,93

В доповіді наведено технологію формування початкових даних і фрагменти результатів комп'ютерного моделювання атак.

Висновки.

Виявлення атак доцільно здійснювати за допомогою мереж Кохонена.

Формування вибірки для вирішення задачі доцільно проводити алгоритмом OneR (відсоток виявлення DOS-атак 94.72%, відсоток виявлення PROBE-атак 74.11%).

ДОДАТОК Е. ДИПЛОМИ ТА ГРАМОТИ

Дипломи та грамоти, отримані у всеукраїнських конкурсах студентських наукових робіт

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ХНУ

ГРАМОТА

нагороджується

Холявка Євген

студент
Сумського державного університету

Науковий керівник - професор Лавров Є.А.

за 3 місце

у II турі Всеукраїнського конкурсу
студентських наукових робіт з
Комп'ютерної інженерії

Ректор Скиба М.Є.

м. Хмельницький
2019 рік

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

ДИПЛОМ

нагороджується

Холявка Є.П.

(прізвище, ім'я, по батькові)

який (яка) зайняв (ла) IV місце
у II турі Всеукраїнського конкурсу студентських робіт
2018–2019 рр.

з Інженерія програмного забезпечення

(назва дисципліни)

Голова оргкомітету конкурсу
Проректор з наукової роботи В.С. Мартинюк

м. Київ – 2019

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

ДИПЛОМ

нагороджується

Холявка Євген Петрович

студент Сумського державного університету

ПЕРЕМОЖЕЦЬ

другого туру Всеукраїнського конкурсу студентських наукових робіт з галузей знань і спеціальностей у 2018/2019 навчальному році за спеціальністю «Кібербезпека»

Голова галузевої конкурсної комісії,
проректор з науково-педагогічної роботи О.М. Новіков

05 квітня 2019 р.

ДОДАТОК Ж. АКТИ ВПРОВАДЖЕНЬ

Акти впроваджень за темою:
«Інформаційна технологія виявлення мережевих атак в критичних інформаційних системах»

Затверджую
Перший проректор
Сумського державного
університету
_____ Карпуша В.Д.
«___» _____ 2019 р.

Акт
Впровадження в навчальний процес
СУМСЬКОГО ДЕРЖАВНОГО УНІВЕРСИТЕТУ
результатів наукової роботи
студентки групи ІТ-51 Сумського державного університету
Холявка Євген Петрович
на тему
«Інформаційна технологія виявлення мережевих атак в критичних інформаційних системах»

Складений 5 січня 2019 р. комісією у складі:

Голова комісії:

Доцент кафедри комп'ютерних наук, зав. секції «Інформаційні технології проектування», кандидат технічних наук ,доцент ***Шендрик В.В.***

Члени комісії:

4. Професор кафедри *комп'ютерних наук, доктор технічних наук, професор* ***Лавров Є.А.***
5. **Доцент кафедри** *комп'ютерних наук, кандидат технічних наук, доцент* ***Чибіряк Я.І.***
6. Старший викладач кафедри *комп'ютерних наук, кандидат технічних наук,* **Кузнєцов Е.Г.**

В період з 3 січня 2019 р. по 5 січня 2019 р. комісія провела роботу з визначення впровадження результатів Холявка Є.П. в навчальний процес кафедри комп'ютерних наук.

Результати роботи комісії

1. На кафедру комп'ютерних наук передано комплекс програм «Інформаційна технологіявиявлення мережевих атак в критичних інформаційних системах».
2. Матеріали використні в дисциплінах:
   - **«Системи підтримки прийняття рішень»** для слухачів магістратури, що навчаються за спеціальністю «Інформатика», при розробці теми **«Прийняття рішень в умовах ризику»** (лабораторна робота – 2 год.).
   - **«Організація людино-машинної взаємодії»** для слухачів магістратури, що навчаються за спеціальністю «Інформаційні технології проектування», при розробці теми **«Ергономіка автоматизованих виробництв»** (лабораторна робота – 2 год.).

Голова комісії _____
Члени комісії _____
_____
_____

Довідка про
впровадження в сервісний центр
СумиТехСервіс
результатів наукової роботи

студента групи ІТ-51 Сумського державного університету
Холявки Євгена Петровича
на тему
«Інформаційна технологія виявлення мережевих атак в критичних інформаційних системах»

Передано комплекс програмного забезпечення «Інформаційна технологія виявлення мережевих атак в критичних інформаційних системах».

Матеріали використовуються для виявлення мережевих атак на підприємстві.

Директор підприємства _____ Полунєв Дмитро Михайлович Ідентифікаційний номер 3134603271 Україна м.Суми