Шифр «EUGEN»

Наукова робота на конкурс за напрямом:
Комп’ютерна інженерія

на тему:
«Метод виявлення мережевих атак в комп’ютеризованих системах управління»

2019

ЗМІСТ

ВСТУП

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.

Предмет дослідження. Математична модель і інформаційна технологія виявлення атак.

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

Мета. Розробити методу виявлення мережевих в комп'ютеризованих системах управління 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 комп'ютерних система управління.

Впровадження. Результати впроваджено в навчальний процес Сумського державного університету та в сервісний центр «СумиТехСервіс».

Публікації. За матеріалами дослідження опубліковано 3 наукові роботи. Список робіт та копії публікацій додаються.

Апробації. Результати доповідались на конференції – X Міжнародна студентська конференція «Перший крок у науку».

1 МЕРЕЖЕВІ АТАКИ ТА ТЕХНОЛОГІЇ ЇХ ВИЯВЛЕННЯ

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

Класифікація мережевих атак за характером впливу мережеві атаки поділяють на [2]:

1) пасивні;
2) активні.

При пасивному впливі не відбувається прямого впливу на систему, через що виявлення такого виду атак важче, ніж виявлення атак з активним впливом.

При активному впливі на функціонування системи виявляється безпосередній вплив, яке може порушити її функціонування, змінити конфігурацію і т.д.

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

1) атаки розвідки;
2) атаки отримання доступу;
3) атаки відмови в обслуговуванні.

Дані типи атак рідко 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.

Атаки типу «відмова в обслуговуванні» (Denial of Service, DoS) вважаються найпоширенішим видом атак. Вони відрізняються від атак інших типів, 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 доступ 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).

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.

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]:

1) методи виявлення зловживань;
2) 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:

1) сигнатурні методи;
2) мови опису сценаріїв;
3) методи на основі кінцевих автоматів;
4) експертні системи;
5) мережі Петрі;
6) 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.

В основі методів виявлення аномалій лежить модель передбачуваної нормальної і очікуваної поведінки користувачів або додатків, і 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 і методи машинного навчання, а так само поведінкові методи [4].

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

Поведінкові методи включають в себе:

1) вейвлет-аналіз;
2) статистичний аналіз;
3) аналіз ентропії;
4) спектральний аналіз;
5) фрактальний аналіз;
6) кластерний аналіз.

Загальна схема методів виявлення мережевих атак наведена в Додатку А

Перевага методів виявлення мережевих аномалій – можливість виявлення раніше невідомих атак незалежно від того, чи міститься інформація про них в базі чи ні, але мають місце помилкові спрацьовування. Для оптимальної роботи систем виявлення атак потрібне комплексне використання методів виявлення аномалій і зловживань.

2 РОЗРОБКА ПІДХОДУ ДО ПОБУДОВИ МОДЕЛІ ВИЗНАЧЕННЯ НАЯВНИХ МЕРЕЖЕВИХ АТАК

2.1 Аналіз даних необхідних для виявлення мережевих атак

Система виявлення атак може бути побудована і навчена тільки на даних, що охоплюють і моделюють всілякі 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). 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 Б.

2.2 Задача виявлення мережевих атак як задача кластеризації

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 певних класів, при

кластеризації розподіл відбувається одночасно з формуванням класів, тобто класи не були попередньо визначені [8].

Кластеризація допомагає вирішити ряд завдань:

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

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

3) 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.

2.3 Підхід до виявлення мережевих атак методом застосування самоорганізуючих карт Кохонена

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

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

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

$$w_j = [w_{j1}, w_{j2}, \dots w_{jN}]^T, \qquad (1)$$

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:

$$d(x, w_j) = \min_{1 \leq j \leq n} d(x, w_j) \quad (2)$$

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:

$$w_i^{k+1} = w_i^k + \eta_i^k(d, k) \cdot a(k) \cdot \left[x(k) - w_i^k\right], \quad (3)$$

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

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

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

$$\eta_i^k(d, k) = e^{-\frac{d_i}{2\sigma(k)}} \quad (4)$$

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 2.3.1

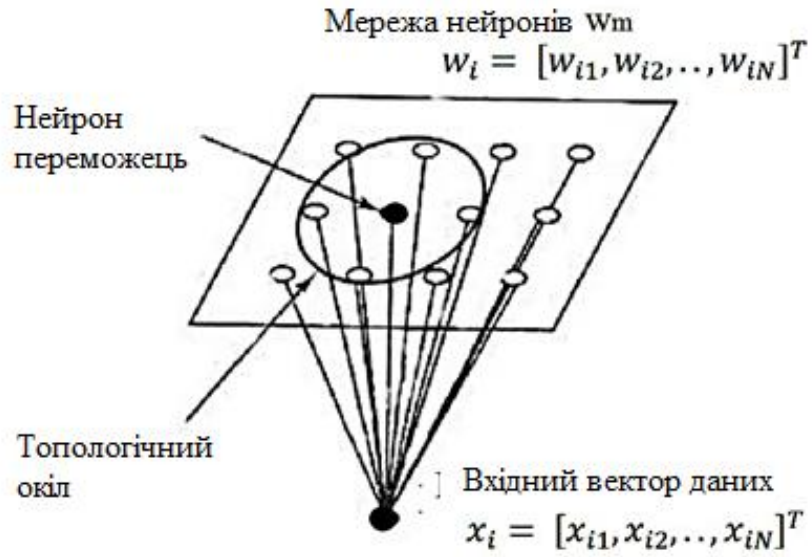


Рисунок 2.3.1 – Структурна схема самоорганізуючої карти Кохонена

Існує безліч варіантів алгоритмів побудови самоорганізуючих систем. Основні відмінності зачіпають процес ініціалізації карт, яка дозволяє прискорити процес збіжності, але коротко все алгоритми побудови систем що самоорганізуючої карти можна описати такою послідовністю кроків:

1. Ініціалізація карти, тобто початкове завдання векторів ваг для вузлів.
2. Підвибірки. Вибираємо вектор X з вхідного простору.
3. Пошук максимального подібності. Знаходимо найбільш підходящий нейрон (нейрон-переможець).
4. Корекція. Коригуємо ваги нейронів, що входять в топологічну околиця.
5. Продовження. Повертаємося до кроку 2 і продовжуємо обчислення до тих пір, поки в карті не перестануть 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.

Для гарного візуального сприйняття карти в разі, коли велика 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 2.3.2).

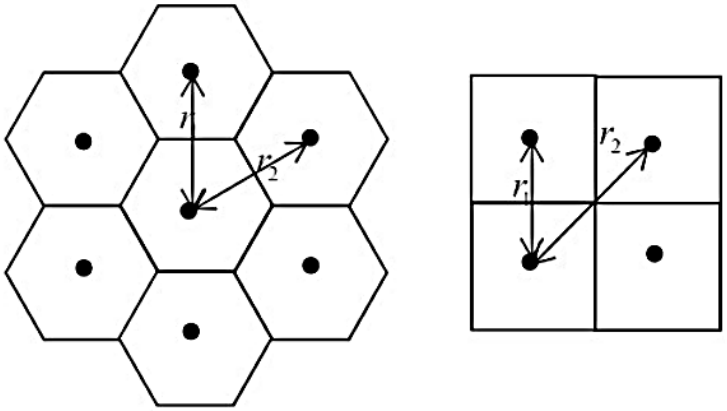


Рисунок 2.3.2 – Гексагональні і чотирикутні осередки

У разі навчання при малій кількості даних для отримання хорошої статистичної точності потрібно дуже багато кроків, але найчастіше 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.

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

3 РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ВИЯВЛЕННЯ АТАК

3.1 Вибір засобів реалізації

Мережі Кохонена реалізовані в великій кількості програмних засобів, серед них: MatLab, Statistika, R, WEKA.

В даній роботі, в зв'язку з необхідністю використання ліцензійних програм, пропонується, для застосування з 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.

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:

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

2) NormalizeAttributes – 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];

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

4) CongenceEpochs – кількість епох в фазі збіжності;

5) OrderEpochs – 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:

$$InfoGain\ (class, Attribute) = H\ (class) - H\ (class \setminus Attribute), \quad (5)$$

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:

$$GainR(class, Attribute) = \frac{\mathrm{H}\ (class)\ -\ \mathrm{H}\ (class \setminus Attribute)}{\mathrm{H}\ (Attribute)}, \quad (6)$$

Gain Attribute Evaluator нормалізує дані пропорційно їх частоті.

OneR (скор. від One Rule) – простий алгоритм формування правил для класифікації об'єктів. Для значення кожної незалежної змінної будуються правила, для даного правила обчислюється помилка – кількість об'єктів з тим же значенням незалежної змінної, але які не відповідають тим значенням залежної змінної, яка найчастіше зустрічається для даного значення незалежної змінної. В результаті вибирається змінна, по якій можна з найбільшою точністю класифікувати об'єкти.

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

3.2 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. За підсумками роботи алгоритму SOM в один кластер будуть визначені дані імовірно нормального поведінки, в другій імовірно аномального поведінки.

Для вихідної вибірки без нормалізації параметрів (функція normalizeAttribute) 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 3.2.1.

Таблиця 3.2.1 – Розподіл даних по кластерам без нормалізації даних

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

Нормалізація параметрів дає значний приріст до виявлення аномалій. Результати роботи алгоритму з нормалізацією наведені в таблиці 3.2.2.

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 3.2.2 – 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 даними алгоритмом наведені в таблиці 3.2.3.

Таблиця 3.2.3 – Ранжування атрибутів алгоритмом Gain Ratio Attribute Evaluator

№ за значимістю	№ у вихідній 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 40 вибірок і проведено тестування алгоритму SOM (таблиця 3.2.4).

Таблиця 3.2.4 – 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

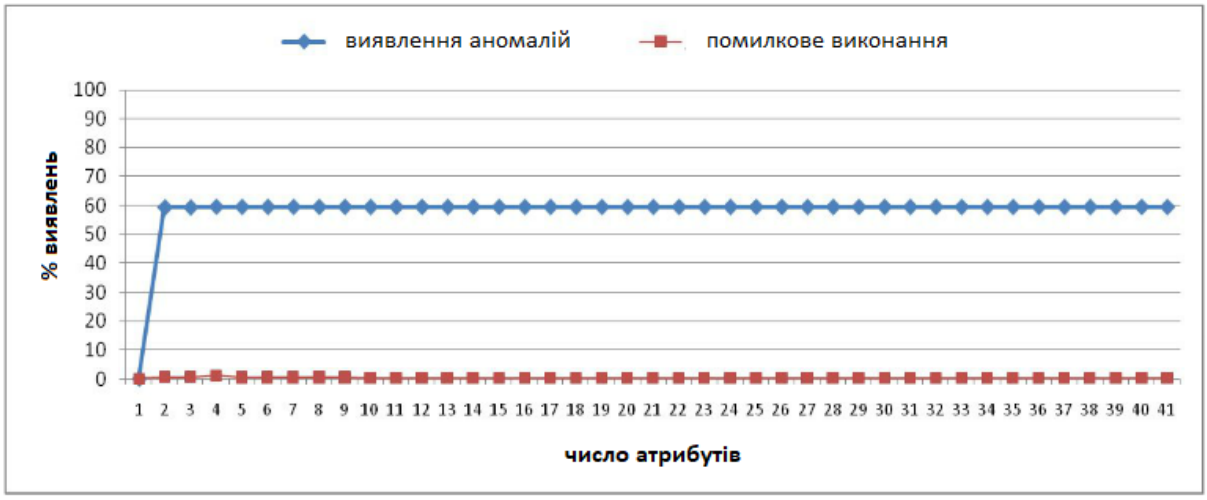


Рисунок 3.2.1 - Графіки залежності відсотка виявлення мережевих атак і помилкових спрацьовувань від числа параметрів, обраних алгоритмом Gain Ratio Attribute Evaluator

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. 3.2.5).

Таблиця 3.2.5 – 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 (70 секунд проти 479 секунд вихідної вибірки з 41 атрибутом) без втрати в ефективності виявлення мережевих атак (незначне зростання з 59,53% до 59,6%). Графік залежності загальних результатів визначення даних наведені на рисунку 3.2.2.

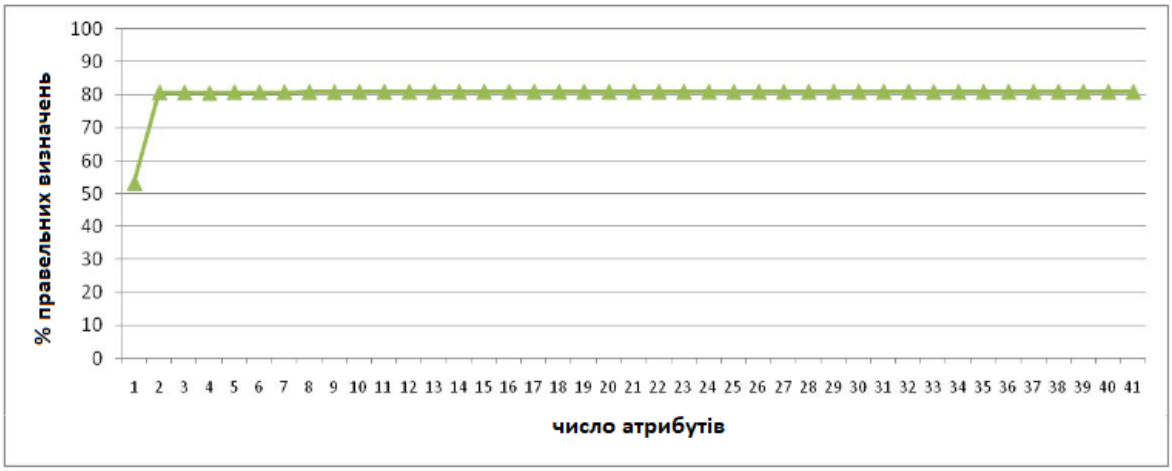


Рисунок 3.2.2 – Графік залежності вірного визначення даних алгоритмом SOM від числа параметрів, обраних алгоритмом Gain Ratio Attribute Evaluator

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 3.2.6.

Таблиця 3.2.6 – Ранжування атрибутів алгоритмом Information Gain Attribute Evaluator

№ за значимістю	№ у вихідній 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 40 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 3.2.7

Таблиця 3.2.7 – 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

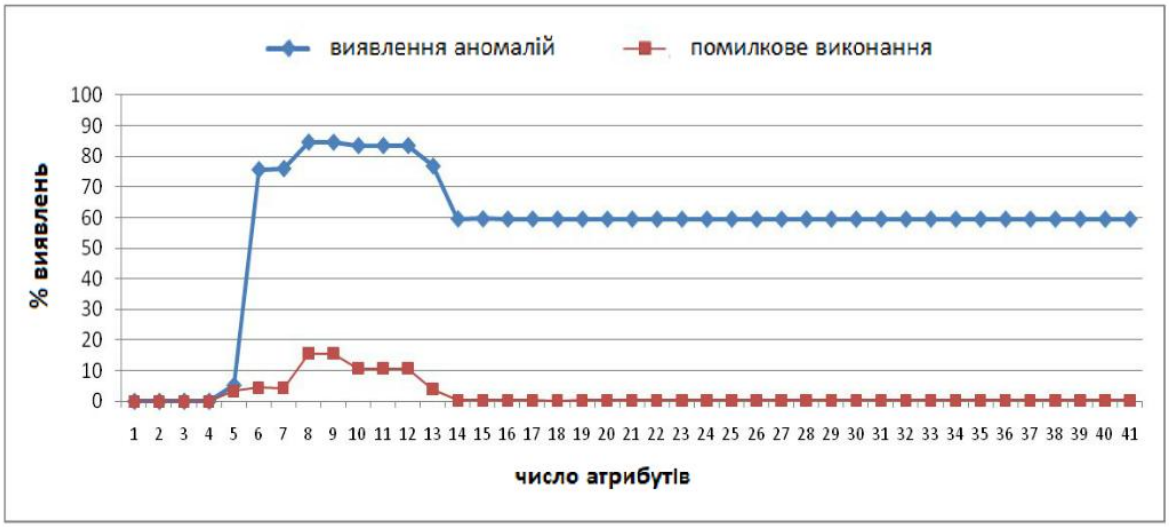


Рисунок 3.2.3 - Графіки залежності відсотка виявлення мережевих атак і помилкових спрацьовувань від числа параметрів, обраних алгоритмом Information Gain Attribute Evaluator

Варто відзначити, що при збільшеному відсотку виявлення мережевих атак, час роботи алгоритму побудови SOM вдалося значно знизити (83 секунди проти 479 секунд з вихідної вибіркою).

Кращі результати виявлення мережевих атак показала вибірка з шести атрибутів.

Кращі результати виявлення мережевих атак показала вибірка з шести атрибутів (таблиця 3.2.8).

Таблиця 3.2.8 – Атрибути, що дали кращі результати, відібрані алгоритмом Information Gain Attribute Evaluator

№ у вихідній 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 Correlation Attribute Evaluator наведені на рисунку 3.2.4.

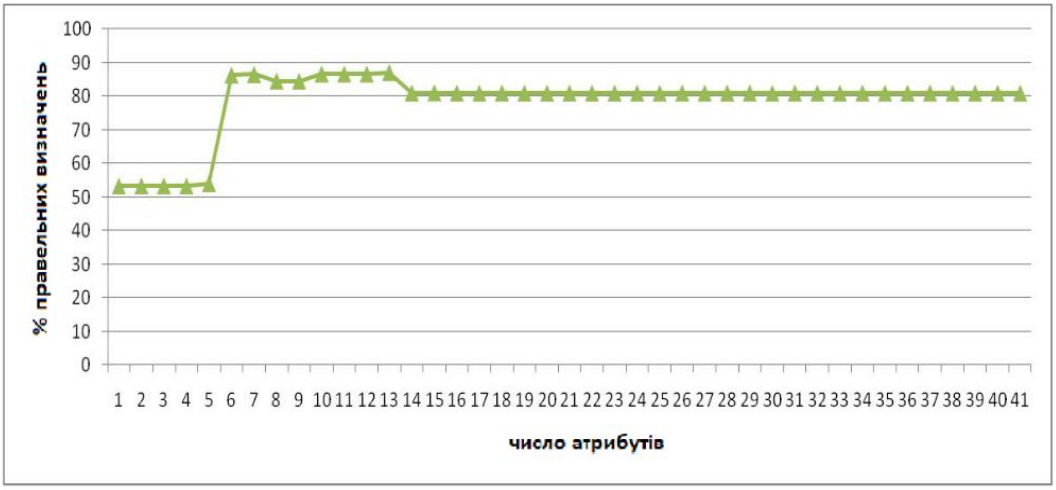
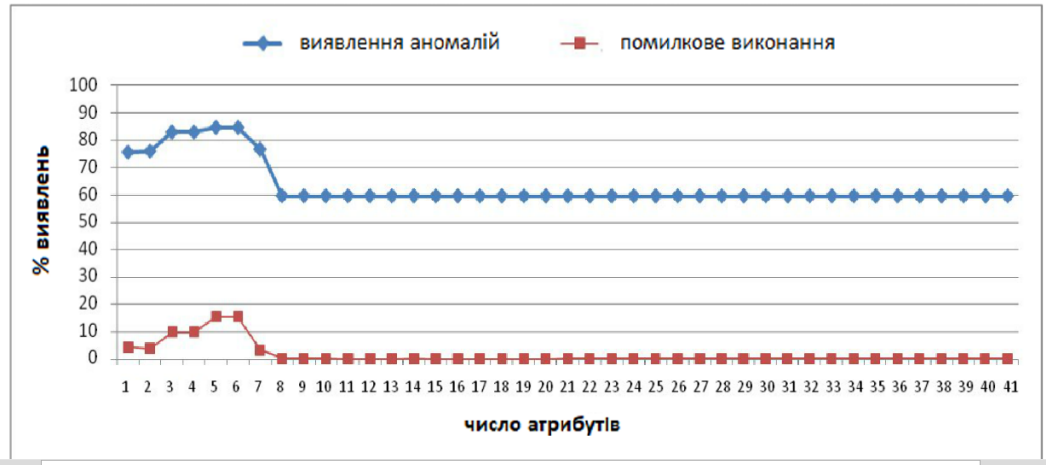


Рисунок 3.2.4 – Графік залежності вірного визначення даних алгоритмом SOM від числа параметрів, обраних алгоритмом Information Gain Attr ibute Evaluator

Було проведення ранжування алгоритмом Correlation Attribute Evaluator.

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

В ході побудови системо самоорганізуючих карт на основі вибірок атрибутів (Додаток Г), створених за результатами роботи алгоритму Correlation Attribute Evaluator, вдалося отримати поліпшення в виявленні мережевих атак в декількох випадках (рис. 3.2.5). Кращі результати показала вибірка з п'яти атрибутів (таблиця 3.2.9).



Риунок 3.2.5 – Графіки залежності відсотка виявлення мережевих атак і помилкових спрацьовувань про т числа параметрів, обраних алгоритмом Correlation Attribute Evaluator

Таблиця 3.2.9 – Атрибути, що дали кращі результати, відібрані алгоритмом Correlation Attribute Evaluator

№ у вихідній вибірці	Атрибут
29	same_srv_rate
33	dst host srv count
34	dst host same srv rate
12	loggedjn
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 (Додаток Ґ) допомогло отримати найкращі результати в порівнянні з усіма вищеописаними алгоритмами.

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

Зріс відсоток як виявлення мережевих атак, так і загального результату визначення даних (рис. 3.2.6).

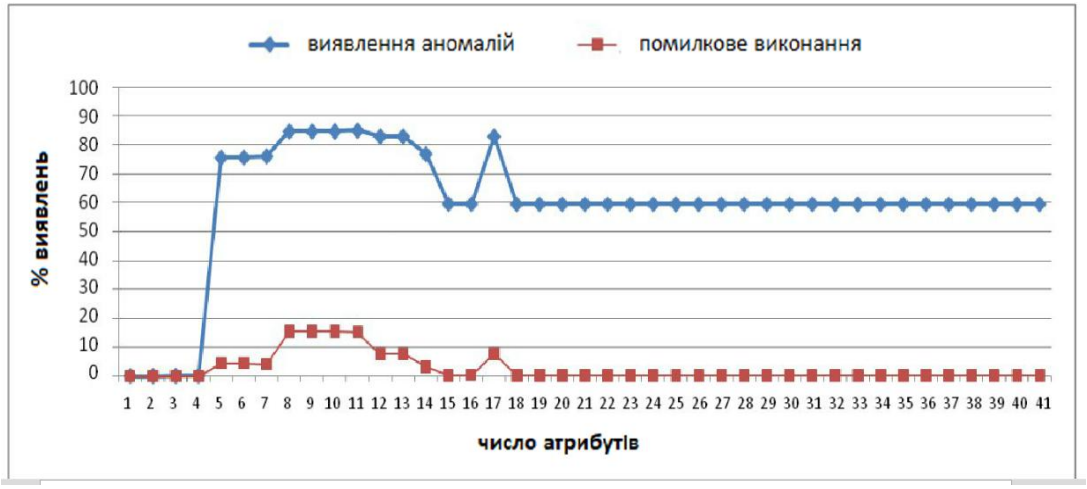


Рисунок 3.2.6 – Графіки залежності відсотка виявлення мережевих атак і помилкових спрацьовувань від числа параметрів, обраних алгоритмом OneR Attribute Evaluator

Кращі результати показала вибірка з одинадцяти атрибутів (таблиця 3.2.10).

Таблиця 3.2.10 – Атрибути, що дали кращі результати, відібрані алгоритмом OneR Attribute Evaluator

№ у вихідній 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%.

3.3 Аналіз результатів

Для того щоб оцінити результати роботи, можна порівняти результати виявлення мережевих атак, які можна отримати використовую отриману 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 линів ного дискримінантного аналізу.

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

Використовуючи запропоновані вибірки, були отримані наступні результати:

Таблиця 3.3.1 – 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 [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 3.3.2 – 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 3.3.3).

Таблиця 3.3.3 – 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 3.3.4).

Таблиця 3.3.4 – 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

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 сформовано як задача кластеризації.

Зручним інструментом вирішення завдання задачі кластеризації проблемних ситуацій наявності кібератак може бути програмне забезпечення WEKA. Досліджено методи виявлення кібератак, найменший відсоток помилок задовольняє алгоритм OneR, який забезпечує наступні показники: відсоток виявлення аномалій 84.93%, відсоток вірного визначення даних 84.78%.

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: Літо, 2011 року.- 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. Самоорганізуються 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/ WEKA: attribute selection [Електронний ресурс].
13. WEKA: attribute selection [Электронный 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 / VENKATACHALAM, V. // 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. Lavrov E., Pasko N., Krybidub A. & Tolbatov A. Mathematical models for the distribution of functions between the operators of the computer-integrated flexible manufacturing systems. Proceedings of the XIIIth International Scientific Conference TCSET'2016, Publishing House of Lviv Polytechnic, Lviv–Slavsko, Ukraine February 23 – 26. – 2016, – P. 72-77.

17. Lavrov E. Planning of Group Activity of Man-Operators in Information Systems / E. Lavrov, N. Pasko // International Scientific Conference "UNITECH II". Proceedings. (18-19 November 2011, Gabrovo), Bulgaria. – Gabrovo: University Publishing House "V.APRILOV", 2011, – Volume 1. – P. 371–376.

18. Lyubchak, V., Lavrov, E. & Pasko, N., Ergonomic support of man-machine interaction. Approach to designing of operators' group activities. International Journal of Bio-Medical Soft Computing and Human Sciences, Japan, Tokyo, 2011. – 17, №2. – P. 53–58.

ДОДА

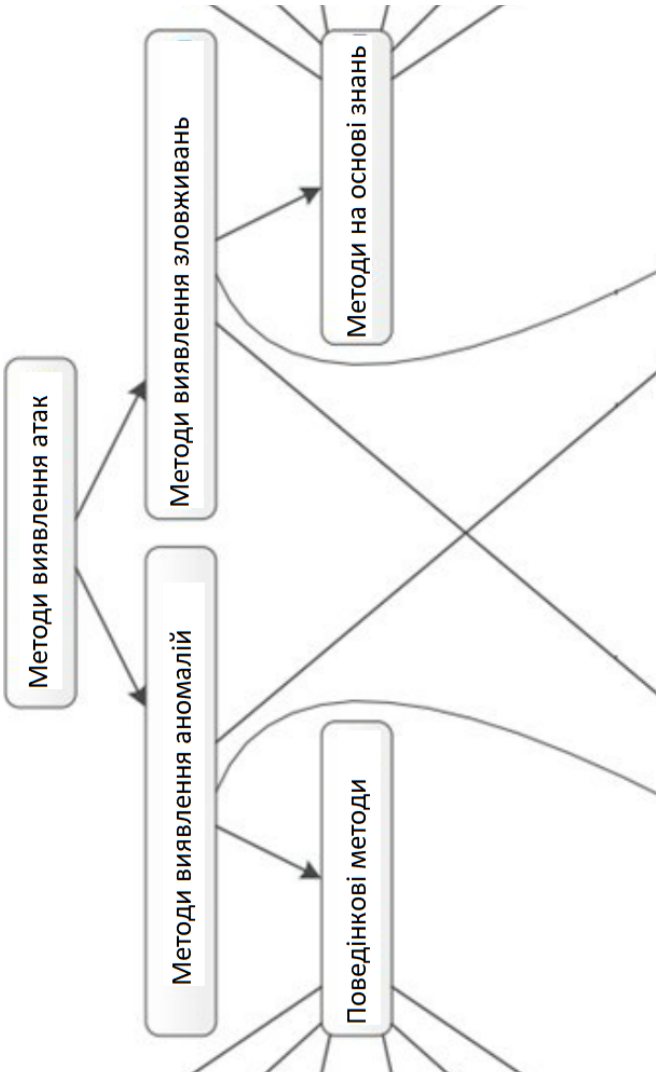
ТОК А

Класифіка

ція методів

виявлення

мережевих атак



ДОДАТОК Б

Опис атрибутів, що використовуються для виявлення атак

№	Назва атрибуту	Визначення
1	duration	Час з'єднання
2	proiocol_type	Тип протоколу
3	service	Мережева служба, використана підключенням
4	flag	Статус з'єднання (нормальне \ з помилкою)
5	src_bytes	Кількість біт даних, від джерела на вузол призначення
6	Lst_bytes	Кількість біт, що приймаються даних від вузла призначення
7	land	Якщо ip (або порти) джерела і приймача рівні - 1, інакше 0
8	wrong_fragment	Кількість невірних фрагментів за сеанс
9	urgent	Кількість термінових пакетів
10	hot	Кількість «гарячих» індикаторів
11	num_failed_logins	Кількість невдалих спроб входу
12	loggcd_in	Статус входження: 1 - успішне, інакше 0
13	num_compromised	Число скомпрометованих станів
14	root_shell	Якщо root -права отримані 1, інакше 0

15	su_attemptcd	Якщо su root -права отримані 1, інакше 0
16	num_root	Число root- доступів
17	num_fi le_creations	Число операцій по створенню файлів під час з'єднання
18	num_shells	Число викликів shell -оболонки
19	num_access_fi les	Число операцій з отримання доступу до файлів
20	num_outbound_cmds	Число вихідних команд в FTP -ссссіі
21	is_host_login	1- логін належить hot-лист тобто якщо є root або адміністратором, 0
22	is_guest_login	Якщо не пройдено ідентифікацію 1, інакше 0
23	count	Кількість підключень до цільового хосту за останні дві секунди
24	srv_count	Число з'єднань зі службою за останні дві секунди
25	serror_rate	Відсоток з'єднань з хостом з count з SYN- помилками
26	srv_serror_rate	Відсоток з'єднань з SYN -помилки при з'єднанні по службі з srvcount
27	rerror_rate	Відсоток з'єднань з хостом з count з REJ- помилками
28	srv_rerror_rate	Відсоток з'єднань з REJ -помилки при з'єднанні по службі з srv_count
29	same_srv_ratc	Відсоток з'єднань з хостом з count використовують одні і ті ж служби
30	diff_srv_rate	Відсоток підключень до різних сервісів
31	srv_diff_host_rate	Відсоток підключень до різних хостам
32	dst_host_count	Кількість з'єднань до локального хосту, встановлених віддаленою стороною
33	dst_host_srv_count	Число з'єднань з тим же самим номером порту
34	d.st_host_same_srv _rate	Процентне число з'єднань до локального хосту, встановлених віддаленою стороною і використовують одну і ту ж службу
35	dst_host_di ff_srv_rate	Відсоток з'єднань по різних служб під час з'єднання з ip з dst_host_count
36	dst_host_same_src_ port_rate	Відсоток з'єднань до того ж самому хосту приймача під час з'єднання але порту з dst_host_srv_count
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
38	d.st_host_serror_rate	Відсоток з'єднань з хостом з dst_host_count з SYN -помилки
39	dst_host_srv_serror_rate	Відсоток з'єднань з SYN -помилки при з'єднанні по службі з dst_host_srv_coun
40	dsi_host_rerror_rate	Відсоток з'єднань з SYN -помилки при з'єднанні по службі з dsl_host_srv_coun
41	dst_host_srv_rerror_rate	Відсоток з'єднань з REJ -помилки при з'єднанні по службі з dsl_host_srv_count

ДОДАТОК В

Результати ранжування атрибутів алгоритмом Correlation Attribute Evaluator

№ за значимістю	№ у вихідній 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

ДОДАТОК Г

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
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

ДОДАТОК Ґ

Результати ранжування атрибутів алгоритмом OneR Attribute Evaluator

№ за значимістю	№ в результат ної 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
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

ДОДАТОК Д

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
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

АНОТАЦІЯ

Шифр наукової роботи: «EUGEN».

Тема роботи: Метод виявлення мережевих атак в комп'ютеризованих системах управління.

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.

Мета. Розробити метод виявлення мережевих в комп'ютеризованих системах управління та обґрунтувати можливість використання її на практиці.

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

Предмет дослідження. Математична модель і інформаційна технологія виявлення атак.

Робота складається з трьох розділів, висновку, списку літератури, шести додатків, включає 30 сторінок, 10 рисунків, 18 літературних джерел.

Публікації. За матеріалами дослідження опубліковано 3 наукові роботи**.** Список робіт та копії публікацій додаються.

Апробації. Результати доповідались на науковій конференції – X Міжнародна студентська конференція «Перший крок у науку».

Впровадження. Результати впроваджено в навчальний процес Сумського державного університету та в сервісний центр «СумиТехСервіс».

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