Міністерство освіти і науки України
Український державний університет науки і технологій

Факультет «Комп'ютерні технології і системи»
(назва факультету)

Кафедра «Електронні обчислювальні машини»
(повна назва кафедри)

Пояснювальна записка
до кваліфікаційної роботи
бакалавра
(ступінь вищої освіти)

на тему: Створення нейронної мережі для визначення мережевих атак категорії U2R

за освітньою програмою Кібербезпека

зі спеціальності: 125 Кібербезпека
(шифр і назва спеціальності)

Виконав: студент групи КБ1811

	(підпис студента)	/ Єгор МЕГЕЛЬБЕЙ / (Ім'я ПРІЗВИЩЕ)
Керівник:	(підпис)	/ доц. Вікторія ПАХОМОВА / (посада, Ім'я ПРІЗВИЩЕ)
Нормоконтролер:	(підпис)	/ ст. викладач Володимир ДЗЮБА / (посада, Ім'я ПРІЗВИЩЕ)

Консультанти:

(назва розділу)	(підпис)	/ (посада, Ім'я ПРІЗВИЩЕ) /
(назва розділу)	(підпис)	/ (посада, Ім'я ПРІЗВИЩЕ) /
(назва розділу)	(підпис)	/ (посада, Ім'я ПРІЗВИЩЕ) /
(назва розділу)	(підпис)	/ (посада, Ім'я ПРІЗВИЩЕ) /

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

Студент _____ (підпис)

Дніпро – 2022 рік

Ministry of Education and Science of Ukraine
Ukrainian State University of Science and Technologies

Faculty «Computer technologies and systems»
(faculty)

Department «Electronic Computers»
(department)

Explanatory Note
to Bachelor's Thesis
bachelor
(higher education degree)

on the topic: Detection of network attacks of the U2R category by means of the Kohonen self- organizing map

according to educational curriculum Cybersecurity

in the Speciality: 125 Cybersecurity
(speciality and its code)

Done by the student of the group: KB1811 / Yehor Mehelbei /
(name, surname)

Scientific Supervisor: / docent Victoria Pakhomova /
(position, name, surname)

Normative controller : / senior lecturer Volodymyr Dzuba /
(position, name, surname)

Supervisors

(Chapter title heading) / / (position, name, surname)

(Chapter title heading) / / (position, name, surname)

(Chapter title heading) / / (position, name, surname)

(Chapter title heading) / / (position, name, surname)

Dnipro – 2022

Міністерство освіти і науки України
Український державний університет науки і технологій

Факультет: Комп'ютерні технології і системи
Кафедра: Електронні обчислювальні машини
Рівень вищої освіти: перший ступінь(бакалавр)
Освітня програма: Кібербезпека
Спеціальність: 125 "Кібербезпека"
(шифр та назва)

ЗАТВЕРДЖУЮ
Завідувач кафедри ЕОМ
_____ Ігор ЖУКОВИЦЬКИЙ
(підпис) (Ім'я ПРІЗВИЩЕ)
Дата _____

ЗАВДАННЯ

на кваліфікаційну роботу перший ступінь (бакалаврський)
(ступінь вищої освіти)

студенту Мегельбею Єгору Олександровичу

1.Тема роботи: Створення нейронної мережі для визначення мережевих атак категорії U2R

Керівник роботи: Пахомова Вікторія Миколаївна, к.т.н., доцент кафедри ЕОМ
(Прізвище, Ім'я, По батькові, науковий ступінь, вчене звання)

затверджені наказом від 07.12.2021 р. № 67 ст

2. Строк подання студентом роботи: 13.06.2022 р.

3. Вихідні дані: дані мережевого трафіку, які розташовані в базі даних NSL-KDD.

4. Зміст пояснювальної записки (перелік питань, які потрібно опрацювати):
4.1 Аналітична частина: огляд і аналіз предмета дослідження.
4.2 Основна частина: постановка задачі; самоорганізуюча карта Кохонена як основний метод рішення задачі; розробка програмної моделі; дослідження параметрів якості визначення атак на програмній моделі; використання розробленої програми в навчальному процесі.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень):

6. Консультанти розділів роботи:

Розділ	Прізвище, ініціали та посада консультанта	Завдання видав: (підпис консультанта, дата)	Завдання прийняв: (підпис студента, дата)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітки
1	Вступ	03.06.22 - 05.06.22	5%
2	Огляд і аналіз предмета дослідження	25.04.22 - 16.04.22	20%
3	Постановка задачі	14.04.22 - 22.04.22	10%
4	Розробка програмної моделі	22.04.22 - 09.05.22	20%
5	Дослідження параметрів якості визначення атак на програмній моделі	07.05.22 - 24.05.22	20%
6	Використання розробленої програми в навчальному процесі	23.05.22 - 02.06.22	20%
7	Висновки	03.06.22 - 05.06.22	5%
8	Подання кваліфікаційної роботи до кафедри	13.06.2022	
9	Захист кваліфікаційної роботи на засіданні Екзаменаційної комісії	23.06.2022	

Студент _____ (підпис) Єгор МЕГЕЛЬБЕЙ (Ім'я ПРІЗВИЩЕ)

Керівник роботи _____ (підпис) Вікторія ПАХОМОВА (Ім'я ПРІЗВИЩЕ)

6. Консультанти розділів роботи:

Розділ	Прізвище, ініціали та посада консультанта	Завдання видав: (підпис консультанта, дата)	Завдання прийняв: (підпис студента, дата)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Вступ	03.06.22 - 05.06.22	5%
2	Огляд і аналіз предмета дослідження	25.04.22 - 16.04.22	20%
3	Постановка задачі	14.04.22 - 22.04.22	10%
4	Розробка програмної моделі	22.04.22 - 09.05.22	20%
5	Дослідження параметрів якості визначення атак на програмній моделі	07.05.22 - 24.05.22	20%
6	Використання розробленої програми в навчальному процесі	23.05.22 - 02.06.22	20%
7	Висновки	03.06.22 - 05.06.22	5%
8	Подання кваліфікаційної роботи до кафедри	13.06.2022	
9	Захист кваліфікаційної роботи на засіданні Екзаменаційної комісії	23.06.2022	

Студент _____ Єгор МЕГЕЛЬБЕЙ
(підпис) (Ім'я ПРІЗВИЩЕ)

Керівник роботи

_____ Вікторія ПАХОМОВА
(підпис) (Ім'я ПРІЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи бакалавра:
(рівень освіти)
31с., 8 рис., 3 табл., 4 додатки., 30 джерел.

Об'єкт розробки – програмна модель для визначення мережевих атак категорії U2R.

Мета роботи – визначення мережевих атак категорій U2R засобами самоорганізуючої карти Кохонена.

Методи дослідження – самоорганізуюча карта Кохонена, мова Python з використанням стандартних бібліотек, оцінка якості визначення атак.

Створено в Python програмну модель «SOM_U2R», в основу якої покладена самоорганізуюча карта Кохонена. У якості вихідних даних програмної моделі взято 41 параметр мережевого трафіку на основі бази NSL-KDD для визначення наступних класів: Rootkit, Bufferoverflow, Loadmodule. На створеній програмній моделі «SOM_U2R» проведено дослідження помилки за різною кількістю епох (10, 20 та 50) при різних розмірах карти: 5*5; 10*10; 20*20. Визначено, що найменше значення помилки досягається на карті 20*20 при 10 епохах. Крім того, проведено дослідження F-мірки за різною кількістю епох навчання самоорганізуючої карти.

Рекомендовано створену програмну модель «SOM_U2R» здобувачам першого (бакалаврського) ступеня спеціальності «Кібербезпека» при виконанні самостійної роботи з дисципліни «Локальні мережі».

Ключові слова: АТАКА, LOADMODULE, ROOTKIT, BUFFER OVERFLOW, NSL–KDD, SOM, PYTHON, ПОХИБКА, ЯКІСТЬ, F–МІРКА.

ЗМІСТ

ВСТУП

Створення ефективної системи виявлення мережевих атак вимагає застосування якісно нових підходів до обробки інформації, які повинні ґрунтуватися на адаптивних алгоритмах здатних до самонавчання. Найбільш перспективним напрямком у створенні подібних систем виявлення атак на комп'ютерну мережу є застосування нейромережевих технологій, що підтверджує актуальність теми даної роботи.

Метою даної роботи є виявлення мережевих класів категорії U2R засобами самоорганізуючої карти Кохонена. Відповідно до мети поставлені наступні задачі:

1. Виконати огляд нейронних мереж щодо визначення мережевих атак.
2. Створити програмну модель самоорганізуючої карти Кохонена для виявлення мережевих класів категорії U2R.
3. Визначити оптимальні параметри самоорганізуючої карти Кохонена на створеній прогарамній моделі.
4. Провести дослідження параметрів якості виявлення мережевих атак на створеній прогарамній моделі.

На сучасному етапі виявленням мережевих атак з використанням нейронних мереж займаються різні вчені та науковці: Жуковицький І. В., Жульков Е. В., Євсеєв С. П., Мустафаєв А. Г, Пахомова В. М., Шматко О. В. та ін. Питанням мережевої безпеки з використанням нейромережних технологій присвячені також наукові праці авторів держав дальнього зарубіжжя: Amini M., Bhavin S., Gunes K., Hadavandi E., Hotho A., Karpinski M., Kruti C., Landes D., Malcolm I. Milevskyi S., Ompriya K., Ortiz A., Rezaeenour J., Ring M., Scheuring D., Wunderlich S., Zincir-Heywood A. Для виявлення мережевих атак вчені надають перевагу наступним нейронним мережам: багатошаровий персептрон, самоорганізуюча карта, радіально-базисна мережа, нейронечітка мережа.

Дана робота складається із вступу, двох розділів та висновків. У першому розділі виконаний огляд нейромереж та підходів щодо виявлення мережевих

атак. У другому розділі сформульована постановка задачі виявлення мережевих класів категорії U2R: buffer overflow, loadmodule, rootkit. з використанням бази даних NSL-KDD. Крім того, в цьому розділі складена в Python програмна модель «SOM_U2R» з використанням апарату самоорганізуючої карта Кохонена з використанням чотирьох нейронів на шарі Кохонена та 41-го параметрів мережевого трафіку. Також проведені дослідження оптимальних параметрів нейронної мережі на створеній програмній моделі «SOM_U2R» та дослідження параметрів якості виявлення атак: помилки першого і другого роду; коректності визначення мережевих атак; помилкових спрацьовувань; достовірності; точності та повноти.

Результати виконання кваліфікаційної роботи подавалися до міжнародного конкурсу студентських наукових проектів «Black Sea Sciense 2022», який проводився в Одеському національному технологічному університеті за підтримкою Міністерства освіти і науки України під егідою Black Sea Universities Network та ISEKI-Food Association за напрямком «Інформаційні технології, автоматизація та робототехніка».

Крім того апробація отриманих результатів відбулася на Всеукраїнській науково-технічній конференції студентів та молодих учених, що проходила в Український державний університет науки і технологій (УДУНТ) 2022 р.

1 ОГЛЯД І АНАЛІЗ ПРЕДМЕТА ДОСЛІДЖЕНЬ

1.1 Нейронні мережі щодо визначення мережевих атак

На сучасному етапі виявленням мережевих атак з використанням нейронних мереж займаються різні вчені та науковці. Так Kruti C., Bhavin S. та Ompriya K. у своїй науковій роботі [1] використовували систему виявлення проникнень що базується на зростаючою Ієрархічною самоорганізуючою картою для виявлення атак типу U2R та L2R. В іншій роботі [2] Карпінські М., Шматко А., Євсеів С., Жанкарчук Д., Мілевський С. використовували багатошаровий персептрон для виявлення атак класів R2L, U2R, Probe, DOS. У своїй роботі [3] Жуковицький І. В., Пахомова В. М., Остапець Д. О., Циганок О. І. використовували нейронну мережу багатошарового персептрона для виявлення мережевих атак класів R2L, U2R та DoS. У роботі [4] в якій Wang M., Lu Y., Qin J. використовували нейронні мережі такі як багатошаровий персептрон, радіальну базисну функцію нейронної мережі, навчаючийся вектор квантування та самоорганізуючу карту для виявлення DDoS атак. Vinícius de Miranda Rios та його команда використовували два підходи для виявлення мережевих атак. Перший підхід [5] використовує багатошаровий персептрон як основу для алгоритмів що використовуються для виявлення DDoS атак: k-найближчих сусідів, метод опорних векторів. У іншому підході використовують комбінацію з трьох різних методів: евклідова відстань, багатошаровий персептрон та нечітка логіка. У роботі [6], що проводилось дослідження виявлення мережевих атак з використанням самоорганізуючої карти. Для навчання і перевірки роботи нейронної мережі використовувалась база даних NSL-KDD.

На сучасному етапі найбільш перспективним напрямком у створенні систем виявлення атак є застосування нейронних мереж: багатошарового персептрону (Multi Layer Perceptron, MLP) [2, 7, 8]; радіально-базисної мережі (Radial Basis Function Network, RBF) [9]; самоорганізуючої карти Кохонена (Self Organizing Maps, SOM) [1, 6, 11]; нейронечіткої мережі (Adaptive-Network-Based Fuzzy Inference System, ANFIS) [9] та на основі поєднання методів обчислювального інтелекту [9, 10].

З одного боку, нейронні мережі з різними топологіями можуть виявляти різні атаки, але помилкові спрацьовування також відбуваються не завжди на одних і тих самих мережевих пакетах при аналізі за допомогою різних типів нейронних мереж. Крім того, кожний тип нейронної мережі має свої переваги та недоліки, які необхідно враховувати або проводити додаткові дослідження. Наприклад, хоча мережа RBF і навчається швидше ніж мережа MLP, але необхідно визначити кількість радіальних елементів, розташовування їх центрів і значення відхилення, модель RBF потребує декілька більшої кількості елементів, тобто буде працювати повільніше і потребує більше пам'яти, ніж модель MLP. Обробка великого обсягу мережного трафіку, який постійно змінюється, на основі MLP з використанням машинного навчання призводить до великої кількості помилкових спрацьовувань і пропусків атак, що також потребує проведення додаткових досліджень для визначення оптимальних параметрів нейронної мережі.

З іншого боку, робляться спроби використання нейронних мереж на різних рівнях. Так, наприклад, в [3] розглянута структура гіпотетичного комплексу, яка представлена на рисунок 1.1, та складається із п'яти нейронних мереж (НМ) типу багатошаровий персептрон: НМ1 – для визначення категорії класу атаки (DoS, R2L, U2R, Probe) або факту того, що атаки не було; НМ2…НМ5 – для виявлення мережевого класу атаки, якщо така мала місце бути (кожна з цих чотирьох нейронних мереж відповідає одної категорії атак і вміє визначати класи, що належать тільки цій категорії). Комплекс включає модуль виявлення мережевих атак, що отримує дані про з'єднання від мережевих датчиків і видає результат до модулю реагування. Сигнал від НМ1, який виявляє категорію атаки, через ключ «ключ» вмикає одну з нейронних мереж НМ2…НМ5, яка визначить мережевий клас відповідно до категорії.

У [12] виконаний огляд існуючих наборів даних, найбільш поширеним із яких є база даних NSL-KDD, що створена з ініціативи Управління перспективних дослідницьких проектів Міністерства оборони США (DARPA) на основі бази даних KDD'99 [13]. Треба визначити, що на сьогодні існує цілий ряд наукових

робіт різних вчених та науковців по визначенню мережевих атак категорій DoS та Probe, але достатньо мало робіт, присвячених дослідженню мережевих класів щодо категорій R2L та U2R. Згідно з [1] існуючі системи виявлення вторгнень на основі самоорганізуючих карт мають труднощі, які пов'язані з великим часом обчислень і низькою частотою виявлення атак U2R і R2L.

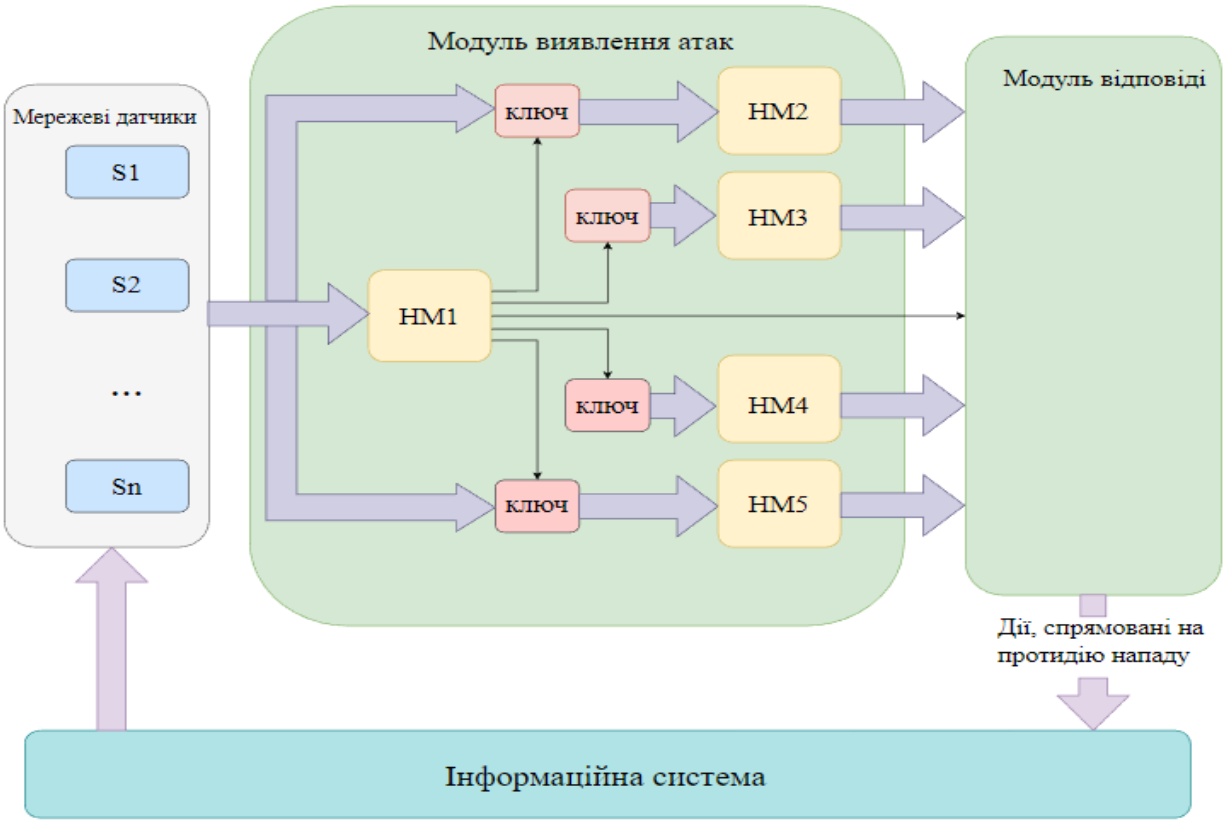

Рисунок 1.1 – Структура гіпотетичного комплексу [3]

У [14] проведено дослідження двох підходів щодо виявлення мережевих атак з використанням одної нейронної мережі та комплексу нейронних мереж на основі розрахунку показників якості визначення атак, серед яких вагоме значення мають помилки першого та другого роду. Помилка першого роду – це кількість невірно виявлених атак (FP, False Positive). Помилка другого роду – це кількість пропусків атак (FN, False Negative). Так, наприклад, в [15] розглянутий новий підхід до побудови багаторівневої мережної системи виявлення вторгнень,

який полягає в тому, що групи однотипних параметрів міжмережної взаємодії подаються на входи окремих модулів першого рівня, кожен з яких представляє собою ієрархічну структуру декількох нейронних мереж різного типу і виконує виявлення аномалій по заданій групі параметрів. Результати роботи модулів першого рівня подаються на вхід вирішувача другого рівня, що приймає остаточне рішення про наявність атаки та її класифікації. За даним підходом ймовірність визначення відомих атак склала 91 %, виявлення вторгнень, інформації про яких не було при навчанні, склала 86 %. Однак, розроблений прототип має відносно значну ймовірність помилки II-го роду 18 %, аналіз та виправлення причин цих помилок є перспективним для подальшого дослідження.

1.2 Засоби програмної реалізації

Реалізувати нейронну мережу можна з використанням нейропакета, або з використанням універсальних мов програмування. У своїй роботі [3] по виявленню мережевих атак Жуковицький І. В., Пахомова В. М., Остапець Д. О., Циганок О. І. використовували для створення програмної моделі мову програмування Python та фреймворк Tensorflow 1.2 для виявлення мережевих атак класів: DoS, U2R, R2L, Probe. Пахомова В.М. та Коннов М.С. у своїй роботі [14] по дослідженню підходів до виявлення мережевих атак використовували нейронні мережі багатошаровий персептрон та самоорганізуючу карту що створені у програмному пакеті MatLAB. У своїх дослідах [4] з виявлення DDoS атак з використанням багатошарового персептрону Wang M. та його команда реалізовували нейронну мережу у програмному пакеті MatLAB.

1.3 Основні висновки

1. Огляд наукових джерел щодо виявлення мережевих атак показав можливість використання наступних нейронних мереж: багатошарового персептрону, самоорганізуючої карти, радіально-базисної мережі, нейронечіткої мережі. Для подальшого виконання кваліфікаційної роботи обрано самоорганізуючу карту Кохонена, тому що вона застосовується до багатьох

різних областей комп'ютерної безпеки по знаходженню потенційних загроз та використовується для вирішення або допомоги іншим методам у вирішенні проблеми безпеки.

2. Огляд наукових джерел щодо виявлення мережевих атак продемонстрував можливість реалізації нейронних мереж за допомогою нейропакетів: Deductor Studio, GENESIS, NEURON, Brian, NEST, XNBC, Neural Planner, Partek Discovery, ThinksPro, RapidMiner, BrainMaker Process Advisor та інші; та мов програмування: Python, Lisp, C++, Java, R, Prolog. Для подальшого виконання кваліфікаційної роботи обрана мова Python, тому що вона має багато бібліотек для реалізації самоорганізуючої карти Кохонена.

2 СТВОРЕННЯ НЕЙРОННОЇ МЕРЕЖІ ДЛЯ ВИЗНАЧЕННЯ АТАК КАТЕГОРІЇ U2R

2.1 Постановка задачі

Стрімкий розвиток комп’ютерних мереж та інформаційних технологій викликає ряд проблем, пов’язаних з безпекою мережевих ресурсів, які потребують ефективних підходів. Використання нейромережевої технології є найбільш раціональним, оскільки нейронні мережі мають наступні переваги: вирішення задач при невідомих закономірностях; стійкість до шумів вхідних даних; адаптування до змін у навколишній середі; потенційна надвисока швидкодія. У даній роботі необхідно виявляти мережеві класи атаки категорії U2R. Мережеві атаки категорії U2R – атаки, що є експлуатацією системи, під час якої хакер починає роботу в системі зі звичайним обліковим записом користувача і намагається зловживати вразливими місцями в системі, щоб отримати привілеї суперкористувача. Цей тип атаки розділяється на такі класи: Buffer_overflow, Loadmodule, Perl, Rootkit [16].

Buffer_overflow – вид атаки, коли зловмисники використовують проблеми переповнення буфера, перезаписуючи пам’ять програми. Це змінює шлях

виконання програми, викликаючи відповідь, яка ушкоджує файли або розкриває особисту інформацію [17].

Loadmodule – це атаки користувача на Root проти систем SunOS 4.1, які використовують віконну систему xnews для двох динамічно завантажуваних драйверів ядра в файл поточної системи та створюють спеціальні пристрої в каталозі /dev, щоб використовувати їх модулі. Через помилку в тому, як програма Loadmodule очищає середовище, неавторизовані користувачі можуть отримати root-доступ на локальній машині [18].

Perl – це вид атаки, яка використовує помилку в деяких реалізаціях Perl.Suidperl; це версія Perl, яка підтримує збережені сценарії set-user-ID і set-group-ID. У ранніх версіях Suidperl інтерпретатор належним чином не відмовляється від своїх привілегій root при зміні його ефективних ідентифікаторів користувачів і груп. У системі, яка має Suidperl, або Sperl, програма встановлена та підтримує збережений ідентифікатор set-user-ID та збережений ідентифікатор групи set-group, будь-хто з доступом до облікового запису в системі може отримати root-доступ [19].

Rootkit – це програма (або набір програм), яка встановлює та виконує код у системі без згоди або відома кінцевого користувача. Він використовує прихованість, щоб підтримувати постійну і непомітну присутність на машині. Руткіти не заражають комп’ютери шкідливим програмним забезпеченням, а намагаються створити невиявлене середовище для виконання шкідливого коду. Руткіти встановлюються в системах за допомогою соціальної інженерії, під час запуску шкідливого програмного забезпечення або просто шляхом перегляду шкідливого веб-сайту. Після встановлення зловмисник може виконувати практично будь-які функції в системі, включаючи віддалений доступ, підслуховування, а також приховування процесів, файлів, ключів реєстру та каналів зв’язку [20].

У базі NSL-KDD [13] представлено у достатній кількості параметри для мережевих класів Buffer_overflow, Rootkit, Loadmodule. Оскільки в базі недостатньо параметрів для класу Perl, він не буде використовуватись.

2.2 Самоорганізуюча карта Кохонена як основний метод вирішення задач

Нейронна мережа Кохонена призначена для вирішення задач кластерного аналізу та об’єднання множин діагностичних ознак у групи, формування класів дефектів ОК, визначення його стану тощо [21]. Він поділяється на багато видів за способами налаштування ваг. У даній роботі використовувалась самоорганізуюча карта Кохонена, яка заснована на навчанні без вчителя. Структура самоорганізуючої карти Кохонена представлена на рис. 2.1.

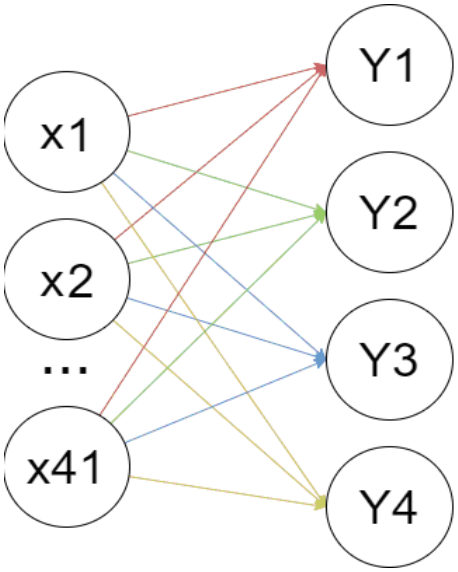

Рисунок 2.1 – Структура самоорганізуючої карти Кохонена

У якості початкових параметрів взято [22]:

X1 (duration) – тривалість з’єднання (у секундах);

X2 (protocol_type) – тип протоколу транспортного рівня;

X3 (service) – сервис прикладного рівня;

X4 (flag) – статус з’єднання;

X5 (src_bytes) – кількість байтів від джерела до призначення;

X6 (dst_bytes) – кількість байтів відповіді клієнту;

X7 (land): 1 – якщо з’єднання від/до того самого хоста/порта;

X8 (wrong_fragment) – кількість «хибних» фрагментів;

X9 (urgent) – кількість термінових пакетів;

X10 (hot) – кількість «гарячих» індикаторів;

X11 (num_failed_logins) – кількість невдалих спроб реєстрації;

X12 (logged_in): 1 – якщо успішний вхід в систему; 0 – якщо неуспішний;

X13 (num_compromised) – кількість «компроментуючих» умов;

X14 (root_shell): 1 – якщо root shell отриманий; 0 – інакше;

X15 (su_attempted): 1 – якщо виконувалась «su root»; 0 – інакше;

X16 (num_root) – кількість «root» доступів;

X17 (num_file_creations) – кількість операцій створення файлів;

X18 (num_shells) – кількість запитів на надання оболонки;

X19 (num_access_files) – кількість операцій на доступ до контролю файлів;

X20 (num_outbound_cmds) кількість вихідних команд для FTP сесії;

X21 (is_hot_login): 1 – якщо логін належав до «гарячого» списку, 0 – інакше;

X22 (is_guest_login): 1 – якщо «гостьовий» вхід, 0 – інакше;

X23 (count) – кількість з'єднань на хост в поточній сесії за останні 2 с;

X24 (srv_count) – кількість з'єднань на такий самий сервіс за останні 2 с;

X25 (serror_rate) – відсоток з'єднань з хостом з count з SYN-помилками;

X26 (srv_serror_rate) – відсоток з'єднань з SYN-помилками при з'єднанні по службі з srv_count;

X27 (rerror_rate) – відсоток з'єднань з REJ-помилками;

X28 (srv_rerror_rate) – відсоток з'єднань з REJ-помилками;

X29 (same_srv_rate) – відсоток з'єднань з однаковим сервісом;

X30 (diff_srv_rate) – відсоток з'єднань з різними сервісами;

X31 (srv_diff_host_rate) – відсоток з'єднань з різними хостами;

X32 (dst_host_count) – кількість з`єднань до локального хоста, встановлених віддаленою стороною;

X33 (dst_host_srv_count) – кількість з`єднань до локального хоста;

X34 (dst_host_same_srv_rate) – відсоток з'єднань з однаковим сервісом;

X35 (dst_host_diff_srv_rate) – відсоток з'єднань з різними службами за час з'єднань по ip з dst_host_srv_count;

X36 (dst_host_same_src_port_rate) – відсоток з'єднань до того ж самого хосту-приймачу за час з'єднань з dst_host_srv_count;

X37 (dst_host_srv_diff_host_rate) ;

X38 (dst_host_serror_rate) – відсоток з'єднань з хостом з dst_host_count з SYN-помилками;

X39 (dst_host_srv_serror_rate) – відсоток з'єданань з SYN-помилкою;

X40 (dst_host_rerror_rate) – відсоток з'єданань з REJ-помилкою;

X41 (dst_host_srv_rerror_rate) – відсоток з'єданань з REJ-помилкою.

У якості результуючих даних:

Y1 (rootkit) – відбулась атака класу Rootkit (0);

Y2 (loadmodule) – відбулась атака класу Loadmodule (1);

Y3 (buffer overflow) – відбулась атака класу Buffer overflow (2);

Y4 (normal) – атаки не було (3).

Алгоритм функціонування самоорганізуючої карти Кохонена можна представити наступним чином [23]:

Крок 1. Вектори ваги вузла розташовувати в довільному порядку на карті.

Крок 2. Випадковим чином обирати вхідний вектор.

Крок 3. Обійти кожен вузол на карті.

Крок 4. Для знаходження схожості між вхідним вектором і ваговим вектором вузла карти, необхідно використати евклідову відстань.

Крок 5. Запам'ятати вузол, який має найменшу відстань, як найкращим вузлом відповідності (Best Matching Unit, BMU).

Крок 6. Оновити вектори ваг вузлів біля BMU включно, шляхом наближення до вхідного вектора за наступною формулою:

$$W_v^{new} = W_v^{old} + \theta(u,v,s) * a(s) * (D(t) - W_v^{old}), \quad (2.1)$$

де W_v^{new} – новий вектор ваги вузла;

W_v^{old} – попередній вектор ваги вузла;

$\theta(u,v,s)$ – функція сусідства;

$a(s)$ – коефіцієнт навчання;

$D(t)$ – вектор цільових вхідних даних;

s – поточна ітерація;

u – індекс найкращого вузла відповідності на карті;

v – індекс вузла на карті.

Крок 7. Збільшити s та повторювати поки $s < \lambda$, де λ – ліміт ітерацій.

2.3 Розробка програмної моделі «SOM_U2R»

2.3.1 Вихідні дані та обмеження

На вхід програми «SOM_U2R» подається файл з розширенням csv, що містить у собі базу даних мережевих атак категорії U2R. База даних містить в собі назву атаки та 41 параметр, що відповідає цій атаці. База даних що використовувалась для навчання наведена на рис. 2.2, а всі елементи бази даних представлені у Додатку А.

Рисунок 2.2 – Фрагмент навчальної вибірки

База даних що використовувалась для тестування програмної моделі «SOM_U2R» наведена на рис. 2.3, а всі елементи бази даних представлені у Додатку Б.

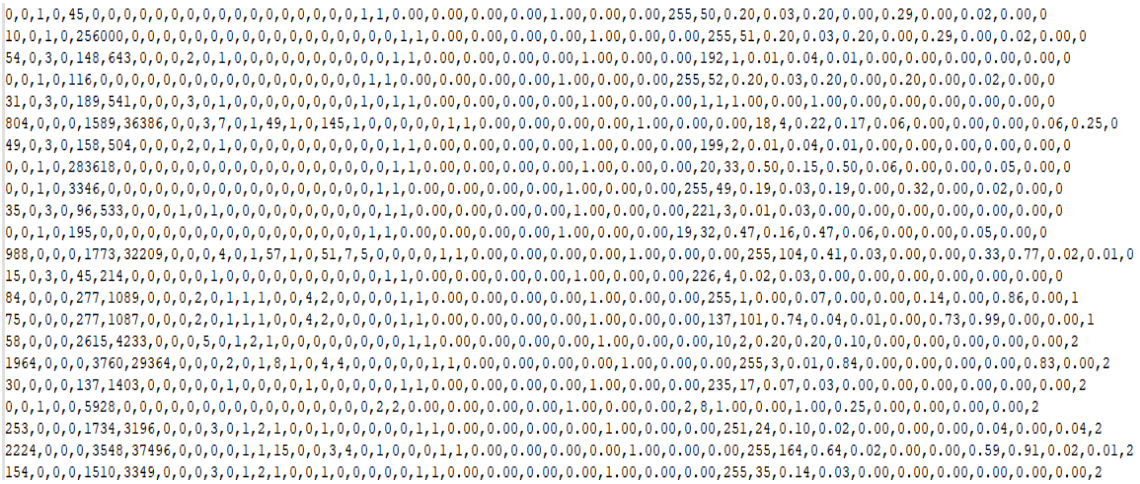

Рисунок 2.3 – Фрагмент вибірки для тестування

На програмну модель «SOM_U2R» накладалися декілька обмежень. Програмна модель очікує на вхід чітко 41 параметр, інакше вона не буде коректно працювати. Файл з вибіркою може бути тільки в csv форматі.

2.3.2 Структура програмної моделі

Для виявлення мережевих атак категорії U2R складена програмна модель «SOM_U2R», в основу якої покладений алгоритм самоорганізуючої карти Кохонена. Структура моделі «SOM_U2R» показана на рис. 2.4.

На вхід «SOM_U2R» подавалися 41 параметр мережевого трафіку (див. п.2.2), результатом виконання є двовимірна карта з розподіленням на неї мережевими класами атак.

Спочатку підключалися бібліотеки, що потрібні для реалізації потрібних функцій, а саме: Numpy [24], Matplotlib [25], MiniSom [26].

Numpy – це бібліотека для мови Python, що додає підтримку великих багатовимірних масивів і матриць, разом із великою бібліотекою високорівневих (і дуже швидких) математичних функцій для операцій із цими масивами [27].

Matplotlib – це бібліотека для побудови графіків і візуалізації даних.Особливістю Matplotlib є те, що за його допомогою можна виводити формули у вигляді TeX, однак існують проблеми з відображенням кириличних букв [27].

MiniSom – це мінімалістична і заснована на Numpy реалізація самоорганізуючих карт [26].

Рисунок 2.4 – Структура створеної програмної моделі «SOM_U2R»

Зчитування даних вибірки з файлу csv та занесення цих даних у масив змінних для подальшого використання. Далі відбувалася нормалізація отриманих даних. Після цього задавалися: розміри карти, кількість вхідних параметрів, ваги

(випадковим чином) та ін. Наступним кроком є навчання нейромережі, для цього у функцію завантажувалися дані з файлу csv та задавалися ітерації. Результати виводилися на екран у вигляді двовимірної карти та кольорових маркерів, що відповідали за окремі мережеві класи атак.

2.3.3 Апробація програмної моделі

У процесі навчання нейронна мережа змінює та налаштовує свої параметри, а також навчається давати відповіді, які нам вже відомі. За рахунок здатності нейронної мережі до узагальнення, нею можуть бути отримані нові результати, якщо подати на вхід вектор, який не зустрічався раніше. Задача навчання полягає в тому, щоб підібрати такі значення параметрів нейронної мережі, за яких помилка буде мінімальною для навчальної множини.

На вхід нейронної мережі подана навчальна вибірка з 55 прикладів, фрагмент якої наведено на рисунку 2.5. Тестування нейронної мережі проводилось протягом 10 епох. Результат роботи програмної моделі «SOM_U2R» з навчання та тестування самоорганізуючої карти Кохонена представлений на рисунку 2.6.

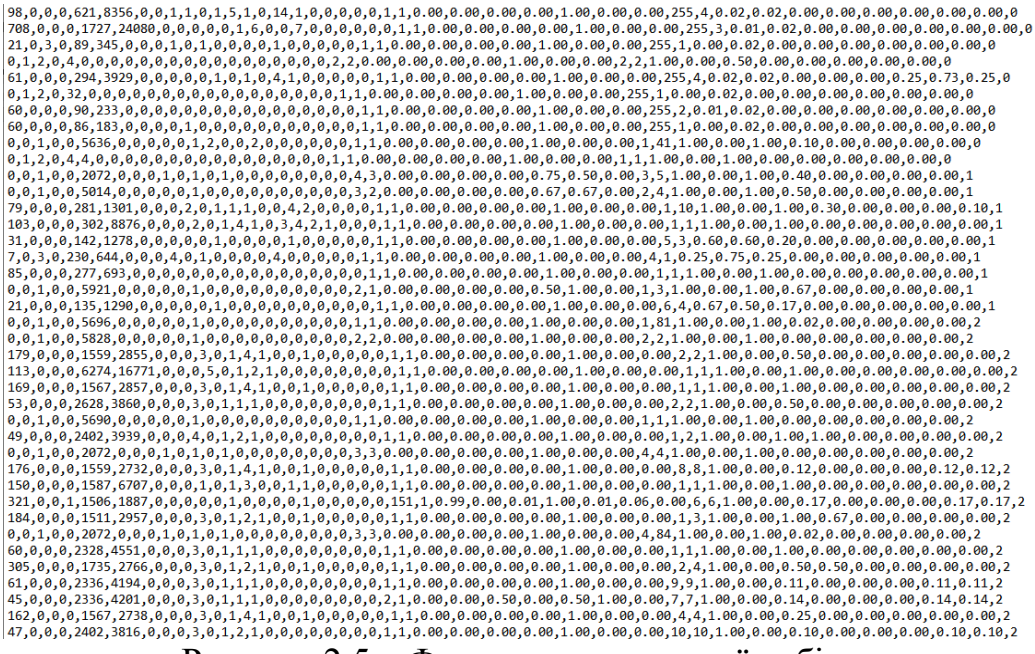

Рисунок 2.5 – Фрагмент навчальної вибірки

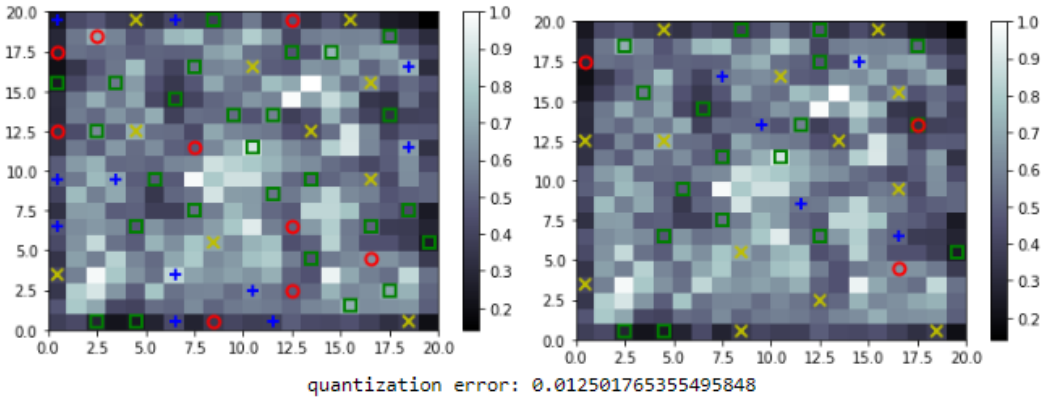

Рисунок 2.6 – Результат моделі «SOM_U2R» по навчанню та тестуванню: червоний круг – клас Rootkit; зелений квадрат – клас Loadmodule; синій хрест – клас Buffer overflow; жовтий хрест – Normal

Кількість епох навчання дорівнювалося 10; розмірність карти складала 20*20. Для тестування нейронної мережі використано вибірку, що містить 40 векторів (прикладів).

2.4 Дослідження параметрів якості визначення атак на програмній моделі

Проведені на створеній програмній моделі «SOM_U2R» дослідження похибки самоорганізуючої карти Кохонена за різною кількістю епох (10, 20 та 50) при різних розмірах карти: 5*5; 10*10; 20*20. Отримані результати зведені до таблиці 2.1.

Таблиця 2.1 – Результати, що отримані на програмній моделі «SOM_U2R»

Кількість епох	Розмір карти	Помилка
50	5*5	0,7725
	10*10	0,2537
	20*20	0,0034
20	5*5	0,7241
	10*10	0,2305
	20*20	0,0095
10	5*5	0,7181
	10*10	0,1724
	20*20	0,00056

Із таблиці видно, що найменше значення помилки досягається для самоорганізуючої карти Кохонена, розмір якої 20*20, при 10 епохах.

Оцінка якості виявлення мережевих атак на створеній програмній моделі «SOM_U2R» виконана за наступними параметрами: TP (True Positive) – класифікатор правильно відніс об’єкт до класу, що розглядається; TN (True Negative) – класифікатор правильно стверджує, що об’єкт не належить до класу, що розглядається; FP (False Positive) – класифікатор невірно відніс об’єкт до класу, що розглядається; FN (False Negative) – класифікатор неправильно стверджує, що об’єкт не належить до класу, що розглядається; TPR (True Positive Rate) – показує частку знайдених об’єктів класу; FPR (False Positive Rate) – показує частку неправильних спрацьовувань класифікатора до загальної кількості об’єктів поза класу; accuracy (точність) – показує частку правильних класифікацій; precision (точність) – показує частку об’єктів класу серед об’єктів, виділених класифікатором; recall (повнота) – показує частку знайдених об’єктів класу до загальної кількості об’єктів класу. Одними із основних є помилка другого роду (FN, False Negative); отримані результати зведені до таблиць 2.2.

Таблиця 2.2 – Результати досліджень різних мережевих класів на «SOM_U2R»

Rootkit				Loadmodule				Buffer_overflow			
TP	FP	TP, %	FP, %	TP	FP	TP, %	FP, %	TP	FP	TP, %	FP, %
3	2	19	12	15	2	48	6	2	2	12	12
FN	TN	FN, %	TN, %	FN	TN	FN, %	TN, %	FN	TN	FN, %	TN, %
0	11	0	69	3	11	10	36	1	11	6	70

Із таблиці видно, що найбільше значення помилки другого роду 10 % досягається при виявленні мережевого класу Loadmodule. Отримані значення інших параметрів зведені до таблиці 2.3.

Таблиця 2.3 – Параметри оцінки якості виявлення атак на «SOM_U2R»

Показник	TP	FP	FN	TN	TPR	FPR	Accuracy	Precision	Recall

Buffer_overflow	2	2	1	11	0,67	0,15	0,81	0,50	0,67
Loadmodule	15	2	3	11	0,83	0,15	0,84	0,88	0,83
Rootkit	3	2	0	11	1,00	0,15	0,88	0,60	1,00
Normal	11	2	2	11	0,85	0,15	0,85	0,85	0,85

Із таблиці видно, що створена програмна модель «SOM_U2R» добре визначає мережеві атаки класів buffer overflow, loadmodule та rootkit точність визначення яких складає більше 0.80.

Проведено дослідження F-мірки виявлення мережевих класів атак на створеній програмній моделі «SOM_U2R» за різною кількістю епох навчання (рисунок 2.7). F-мірка являє собою середнє гармонічне значення між точністю та повнотою.

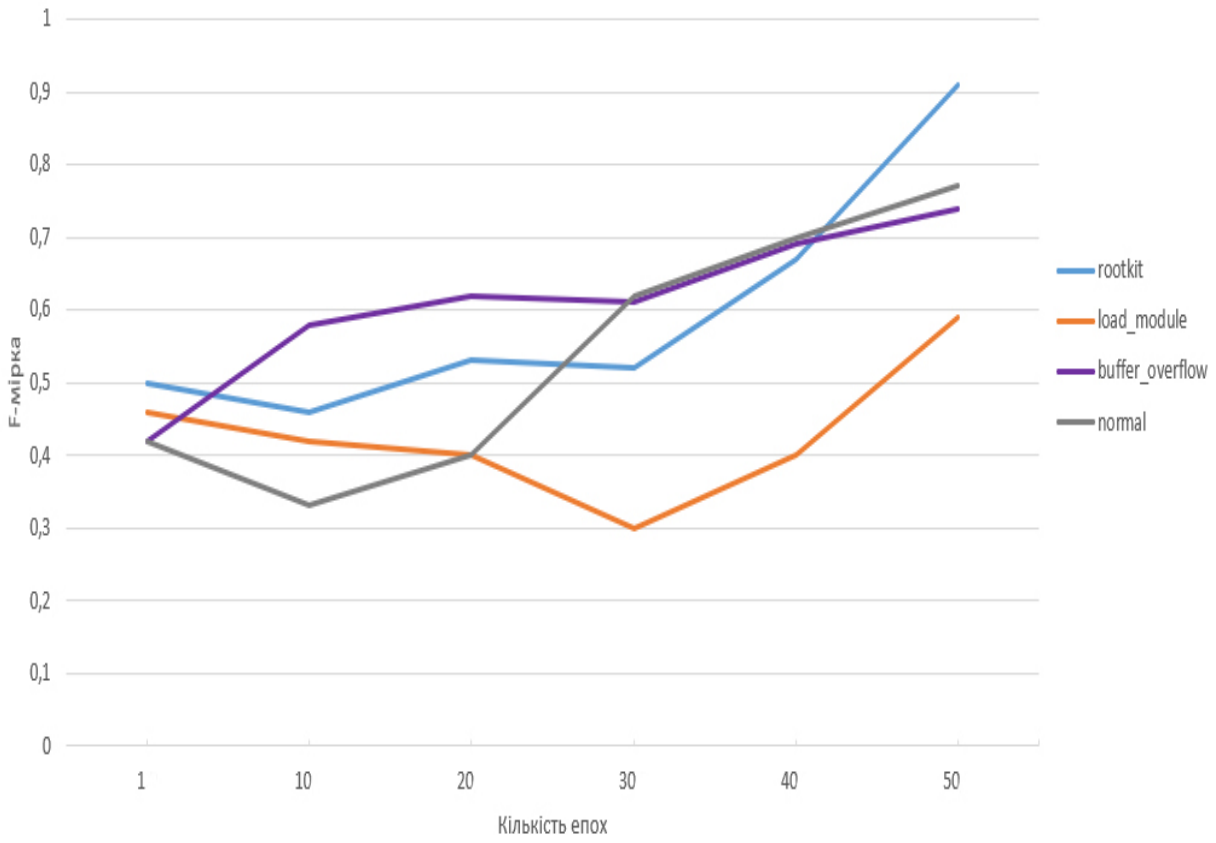

Рисунок 2.7 – Значення F-мірки за різною кількістю епох навчання

Із рисунку видно, що для всіх типів атак, F-мірка зростає, досягаючи свого максимального значення на відмітці 50 епох.

2.5 Використання розробленої програми «SOM_U2R» в навчальному процесі

Дослідження, які проведені на створеній програмній моделі «SOM_U2R» [29] подавалися до міжнародного конкурсу студентських наукових проектів «Black Sea Sciense 2022», який проводився в Одеському національному технологічному університеті за підтримкою Міністерства освіти і науки України під егідою Black Sea Universities Network та ISEKI-Food Association за напрямком «інформаційні технології, автоматизація та робототехніка». У тому числі 45 робіт зарубіжних університетів, 27 робіт з ОНАХТ та 10 з коледжів, і 143 роботи з інших українських закладів [28]. Роботи надійшли з 27 закордонних університетів таких країн як США, Німеччина, Індія, Литва, Латвія, Румунія, Сербія, Польща, Болгарія, Казахстан, Молдова, Грузія і Киргистан. До складу журі Конкурсу надійшли 117 спеціалістів – науковців, з них 73 з 19 зарубіжних країн. Підсумкові бали учасників за напрямом «Інформаційні технології, автоматизація та робототехніка» представлені у вигляді таблиці (Додаток В).

Пропонується для здобувачів першого(бакалаврського) рівня спеціальності «Кібербезпека» при виконанні самостійної роботи з дисципліни «Локальні мережі» [30] додати наступну тему: Визначення мережевих атак категорії U2R на створеній програмній моделі «SOM_U2R». Інструкція по використанню програми представлена в Додатку Б.

2.6 Основні висновки

1. Для виявлення мережевих атак категорії U2R у якості математичного апарату обрано самоорганізуючу карту, де шар Кохена містить чотири нейрони для визначення Buffer_overflow, Loadmodule, Rootkit та відсутності атаки.

2. Для виявлення мережевих класів категорії U2R створена в Python програмна модель «SOM_U2R» з використанням апарату самоорганізуючої

карти (шар Кохонена із чотирьох нейронів), на вхід якої подавалися 41 параметр мережевого трафіку з використанням бази NSL-KDD.

3. На програмній моделі «SOM_U2R» проведені дослідження помилки за різною кількістю епох (10, 20 та 50) при різних розмірах карти: 5*5; 10*10; 20*20. Визначено, що найменше значення помилки 0,00056 досягається на карті 20*20 при 10 епохах.

4. На створеній програмній моделі «SOM_U2R» проведено дослідження F-мірки (середнє гармонічного значення між точністю та повнотою) за різною кількістю епох навчання самоорганізуючої карти Кохонена. Визначено, що з використанням створеної програмної моделі «SOM_U2R» для всіх мережевих класів атак категорії U2R F-мірка зростає, досягаючи свого максимального значення на відмітці 50 епох.

5. Рекомендовано використання програмної моделі «SOM_U2R» першого (бакалаврського) рівня спеціальності «Кібербезпека» при виконанні самостійної роботи з дисципліни «Локальні мережі».

ВИСНОВКИ ТА РЕКОМЕНДАЦІЇ

1. Огляд наукових джерел щодо виявлення мережевих атак показав можливість використання наступних нейронних мереж: багатошаровий персептрон, самоорганізуюча карта, радіально-базисна мережа, нейронечітка мережа. Огляд наукових джерел щодо виявлення мережевих атак продемонстрував можливість реалізації нейронних мереж за допомогою нейропакетів: Deductor Studio, GENESIS, NEURON, Brian, NEST, XNBC, Neural Planner, Partek Discovery, ThinksPro, RapidMiner, BrainMaker Process Advisor та інші; та мов програмування: Python, Lisp, C++, Java, R, Prolog.

2. Для виявлення мережевих класів категорії U2R створена в Python програмна модель «SOM_U2R» з використанням апарату самоорганізуючої карти (шар Кохонена із чотирьох нейронів), на вхід якої подавалися 41 параметр мережевого трафіку з використанням бази NSL-KDD.

3. На програмній моделі «SOM_U2R» проведені дослідження помилки за різною кількістю епох (10, 20 та 50) при різних розмірах карти: 5*5; 10*10; 20*20. Визначено, що найменше значення помилки 0,00056 досягається на карті 20*20 при 10 епохах.

4. На створеній програмній моделі «SOM_U2R» проведено дослідження F-мірки (середнє гармонічного значення між точністю та повнотою) за різною кількістю епох навчання самоорганізуючої карти Кохонена. Визначено, що з використанням створеної програмної моделі «SOM_U2R» для всіх мережевих класів атак категорії U2R F-мірка зростає, досягаючи свого максимального значення на відмітці 50 епох.

5. Рекомендовано використання програмної моделі «SOM_U2R» першого (бакалаврського) рівня спеціальності «Кібербезпека» при виконанні самостійної роботи з дисципліни «Локальні мережі».

ПЕРЕЛІК ПОСИЛАНЬ

1. Kruti C., Bhavin S., Ompriya K. Improving user-to-root and remote-to-local attacks using growing hierarchical self organizing map [Електронний ресурс] // International Journal of Engineering Sciences & Research Technology. – 2015. – Вип. 6. – С. 611–618. – Режим доступу: http://paper.researchbib.com/view/paper/45808. – Заголовок з екрану.

2. M. Karpinski, A. Shmatko, S. Yevseiev, D. Jancarczyk, S. Milevskyi Detection Of Intrusion Attacks Using Neural Networks // Міжнар. наук.-практ. конф. «Інформаційна безпека та інформаційні технології», Харків – Одеса, 13-19 вер. 2021 р. : матер. конф. – Харків : ХНЕУ ім. С. Кузнеця, 2021. – С. 117-124.

3. Zhukovyts'kyy I. V., Pakhomova V. M., Ostapets D. O., Tsyhanok O. I. Detection of attacks on a computer network based on the use of neural network complex. Science and Transport Progress. 2020. № 5(89). 68-79. URL: https://doi.org/10.15802/stp2020/218318.

4. Wang M., Lu Y., Qin J. A dynamic MLP-based DDoS attack detection method using feature selection and feedback[Електронний ресурс] // Computers & Security. – 2020. – Вип. 88. – С.1-14. – Режим доступу: https://www.sciencedirect.com/science/article/pii/S0167404819301890?via%3Dihub.

5. Vinícius de M. R., Pedro R.M, Damien M., Mario F. Detection of reduction-of-quality DDoS attacks using Fuzzy Logic and machine learning algorithms [Virtual resource] // Computer Networks. – 2021. – Vol. 186, 26 March. – 19 p. – Access Mode: URL : https://hal.archives-ouvertes.fr/hal-03182934/file/rios-cn2021.pdf. – Title from Screen. – Date of Access: 16 May 2022.

6. Gunes H. K., Zincir-Heywood A. N., Heywood M. I. Hierarchical SOM-based intrusion detection system [Virtual resource] // Engineering Applications of Artificial Intelligence. – 2007. – Vol. 20, N. 4. – 439 – 451 p. – Access mod: https://researchonline.gcu.ac.uk/en/publications/a-hierarchical-som-based-intrusion-detection-system. – Title from Screen.

7. Мустафаев А.Г. — Нейросетевая система обнаружения компьютерных атак на основе анализа сетевого трафика // Вопросы безопасности. – 2016. – № 2. – С. 1 - 7. – Режим доступу : DOI : 10.7256/2409-7543.2016.2.18834 URL: https://nbpublish.com/library_read_article.php?id=18834.

8. Zhukovyts'kyy I. V., Pakhomova V. M. Identifying threats in computer network based on multilayer neural network. // Science and Transport Progress. – 2018. – № 2 (74). – Access Mode : DOI : https://doi.org/10.15802/stp2018/130797.

9. Amini M., Rezaeenour J., Hadavandi E. A Neural Network Ensemble Classifier for Effective Intrusion Detection using Fuzzy Clustering and Radial Basis Function Networks // International Journal on Artificial Intelligence Tools. – 2016. Vol. 25. N. 02. – Access Mode : DOI : https://doi.org/10.1142/s0218213015500335.

10. Браницкий А. А. Обнаружение аномальных сетевых соединений на основе гибридизации методов вычислительного интеллекта : автореф. дис. ... канд. техн. наук. 05.13.19 / Браницкий А.А. ; Санкт-Петербургский институт информатики и автоматизации Российской академии наук. — Санкт-Петербург, 2018. – 18 с.

11. Kruti C., Bhavin S., Ompriya K. Improving Network Intrusion Detection with Growing Hierarchical Self-Organizing Maps [Virtual resource] // International journal of engineering sciences & researchtechnology. – 2011. – Vol. 6. – P. 611–618. – Access Mode: https://www.semanticscholar.org/paper/Improving-Network-Intrusion-Detection-with-Growing-Ortiz-Ortega/f3fbcf7dfd84d9f2f2ace73580c32eb7c469b6e7. – Title from Screen.

12. Ring M., Wunderlich S., Scheuring D., Landes D., Hotho A. / A Survey of Network-based Intrusion Detection Data Sets. // Computers & Security. – 2019. –Vol. 86. – Access Mode : DOI : 10.1016 / j.cose.2019.06.005.

13. NSL-KDD dataset [Електронний ресурс] // UNB. – 2009. – Режим доступу до ресурсу: https://www.unb.ca/cic/datasets/nsl.htmlю.

14. Pakhomova V. M., Konnov M. S. Research of two approaches to detect network attacks using neural network technologies // Information and communication technologies and mathematical modeling. – 2020. № 3 (87). P. 81-93. – Access Mode : DOI : https://doi.org/10.15802/stp2020/208233.

15. Жульков Е. В. Построение модульных нейронных сетей для обнаружения классов сетевых атак : автореф. дис. ... канд. техн. наукю: 05.13.19 / Е. В. Жульков ; Санкт-Петербургский государственный политехнический университет. – Санкт-Петербург, 2007. –17 с.

16. Дешуніна, Д. С., Грайворонський, М. В. Виявлення атак типу R2L та U2R методами машинного навчання [Електронний ресурс] // Теоретичні і прикладні проблеми фізики, математики та інформатики : матеріали XVI Всеукраїнської науково-практичної конференції студентів, аспірантів та молодих вчених. – 2018. – 26 Травня. – Режим доступу : URL : https://ela.kpi.ua/bitstream/123456789/25265/1/S.96-99.pdf . – Загл. З екрану. – Дата звертання: 1 Травня 2022.

17. Buffer Overflow Attack [Електронний ресурс] // Imperva.com. –2020. – Режим доступу до ресурсу: https://www.imperva.com/learn/application-security/buffer-overflow/.

18. 1993 CERT Advisories [Електронний ресурс] // Resources.cmu.edu. – Режим доступу до ресурсу: https://resources.sei.cmu.edu/asset_files/WhitePaper/1993_019_001_496248.pdf.

19. Intrusion Detection Attacks Database [Електронний ресурс] // Lincoln Laboratory. – 2019. – Режим доступу до ресурсу: https://archive.ll.mit.edu/ideval/docs/attackDB.html#navtop.

20. What is rootkit [Електронний ресурс] // Tremd Micro. – 2020. – Режим доступу до ресурсу: https://www.trendmicro.com/vinfo/us/security/definition/rootkit.

21. Єременко В. С., Переїденко А. В., Монченко О. В. Застосування нейромережевих технологій у системах неруйнівного контролю [Електронний ресурс] // Техническая диагностика и неразрушающий контроль. –2012. – Вип. 1. – С. 35–41. – Режим доступу: https://er.nau.edu.ua/bitstream/NAU/25040/1/%D0%9D%D0%B0%D1%83%D1%87%D0%BD%D0%BE-%D1%82%D0%B5%D1%85%D0%BD%D0%B8%D1%87%D0%B5%D1%81%D0

%BA%D0%B8%D0%B9%20%D1%80%D0%B0%D0%B7%D0%B4%D0%B5%D0%BB.PDF. –Загол. з екрану.

22. Harb H. M., Zaghrot A. A. , Gomaa M. A., Desuky A. S. Selecting Optimal Subset of Features for Intrusion Detection Systems [Virtual resource] // Advances in Computational Sciences and Technology. – Vol. 4. – P.179–192. – Access Mode: http://dspace.must.edu.eg/bitstream/handle/123456789/674/SelectingOptimalSubsetofFeaturesforIntrusion%20%281%29.pdf?sequence=1&isAllowed=y. – Title from Screen.

23. Self-organizing map [Електронний ресурс] // Wikipedia. – 2011. – Режим доступудо ресурсу: https://en.wikipedia.org/wiki/Self-organizing_map.

24. Numpy documentation [Електронний ресурс] // Numpy.org. – 2022. – Режим доступу до ресурсу: https://numpy.org/doc/stable/user/whatisnumpy.html.

25. Matplotlib documentation [Електронний ресурс] // Matplotlib. – 2022. – Режим доступу до ресурсу: https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.

26. MiniSom documentation [Електронний ресурс] // Github.com. – 2022. – Режим доступу до ресурсу: https://github.com/JustGlowing/minisom.

27. Демків Т.М., Демків Л.С. Пакети Python для моделювання фізичних процесів [Електронний ресурс] // Electronic archive of Ternopil National Ivan Puluj Technical University. – 2015. – 23 Квітня. – С.21-22 – Режим доступу до ресурсу : URL : ahds.ac.uk/e-science/documents/Terras-report.pdf. – Загл. з екрану. – Дата звернення: 1 Травня 2022.

28. International Competition of Student Scientific Works «Black Sea Science» [Електронний ресурс] // Одеський національний університет – 2022. – Режим доступу до ресурсу: http://isc.ontu.edu.ua/?lang=en.

29. Мегельбей Є. О., Пахомова В. М. Створення нейронної мережі для визначення мережевих атак категорії U2R. Молода академія 2022. Збірка ТЕЗ доповідей. Всеукраїнська науково-техніка конференція. Студентів і молодих учених. Дніпро : УДУНТ, 2022. С.144.

30. Дистанційний курс з дисципліни «Локальні мережі»; укладач: доцент Пахомова В.М. [Електронний ресурс]. – Режим доступу до ресурсу: https://lider.diit.edu.ua/enrol/index.php?id=344.