Міністерство освіти і науки України
Київський національний університет імені Тараса Шевченка

Факультет інформаційних технологій
Кафедра кібербезпеки та захисту інформації

ДОПУСТИТИ ДО ЗАХИСТУ:
В.о. завідувача кафедри
кібербезпеки та захисту
інформації
_____ Іван ПАРХОМЕНКО
« » червня 2025 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА

кваліфікаційної роботи

галузь знань 12 Інформаційні технології
(шифр і назва галузі знань)
спеціальність 125 Кібербезпека
(код і назва спеціальності)
освітній ступень бакалавр
освітня програма Кібербезпека
(назва освітньо-професійної програми)
на тему: «Методи детектування фішингових URL на основі евристичних правил»

Виконавець: студентка IV курсу, групи КБ-42

_____ **Маргарита ТОЛСТЯК**
(підпис) (ім'я, прізвище)

	Підпис	**Ім'я ПРІЗВИЩЕ**
Керівник		Сергій БУЧИК
Нормоконтроль		Леся БАРАНОВСЬКА

Київ 2025

Міністерство освіти і науки України
Київський національний університет імені Тараса Шевченка

Факультет інформаційних технологій
Кафедра кібербезпеки та захисту інформації

ЗАТВЕРДЖЕНО:
В.о. завідувача кафедри
кібербезпеки
та захисту інформації
_____ Іван ПАРХОМЕНКО
«29» листопада 2024 р.

ЗАВДАННЯ
на виконання кваліфікаційної роботи

спеціальності 125 Кібербезпека
(код і назва спеціальності)

освітньої програми Кібербезпека
(назва освітньо-професійної програми)

Студентці **КБ-42** **Толстяк Маргариті Сергіївні**
(група) (прізвище ім'я по батькові)

Тема кваліфікаційної роботи «Методи детектування фішингових URL на основі евристичних правил»

1. ПІДСТАВИ ДЛЯ ПРОВЕДЕННЯ РОБОТИ

Тема кваліфікаційної роботи затверджена на засіданні кафедри кібербезпеки та захисту інформації протокол №6 від 28.11.2024 р.

2. ВИХІДНІ ДАНІ ДЛЯ ПРОВЕДЕННЯ РОБІТ

Фішингові URL, отримані з відкритих джерел (наприклад https://phishtank.org/).

3. ЗМІСТ РОЗРАХУНКОВО-ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ

Аналіз фішингових атак та методів їх виявлення, характеристика фішингових вебсайтів на основі евристичних ознак, систематизація й класифікація евристичних правил, розробка ментальної мапи як засобу структурування знань, програмна реалізація моделі виявлення фішингових

4. ВИМОГИ ДО РЕЗУЛЬТАТІВ ВИКОНАННЯ РОБОТИ

Практична цінність Програмне рішення, яке спрямоване на виявлення

фішингових вебсайтів.

5. ДАТА ВИДАЧІ ЗАВДАННЯ

Дата видачі завдання: 29 листопада 2024 року

Завдання видав _____ Сергій БУЧИК
(підпис) (ім'я, прізвище)

Завдання прийняла до виконання _____ Маргарита ТОЛСТЯК
(підпис) (ім'я, прізвище)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Найменування етапів робіт	Строки виконання робіт (початок-кінець)	Відмітка про виконання
1	Уточнення постановки задачі	29.11 – 07.12	виконано
2	Аналіз літератури	09.12 – 16.12	виконано
3	Дослідження характерних ознак фішингових вебсайтів	17.12 – 09.01	виконано
4	Формалізація евристичних правил для виявлення фішингових сайтів	10.01 – 20.01	виконано
5	Розробка ментальної мапи як графічного представлення класифікації	21.01 – 29.01	виконано
6	Реалізація програмного рішення	30.01 – 02.03	виконано
7	Формування стратегій та методологій	03.03 – 17.03	виконано
8	Тестування та оцінка ефективності програмного рішення	18.03 – 03.04	виконано
9	Проведення порівняння методів виявлення фішингу	04.04 – 15.04	виконано
10	Надання рекомендації для оптимізації евристичної моделі	16.04 – 05.05	виконано
11	Оформлення пояснювальної записки	06.05 – 24.05	виконано
12	Підготовка до захисту	25.05 – 13.06	виконано

Завдання видав _____ Сергій БУЧИК
(підпис) (ім'я, прізвище)

Завдання прийняла до виконання _____ Маргарита ТОЛСТЯК
(підпис) (ім'я, прізвище)

Термін подання кваліфікаційної роботи до ЕК 13 червня 2025 року

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи складається зі вступу, трьох розділів, загальних висновків, списку використаних джерел та додатків. Основний текст займає 74 сторінки, включає в себе вступ, 3 розділи кваліфікаційної роботи та висновки. Крім того, робота містить 2 додатки із загальною кількістю сторінок 14. У пояснювальній записці кваліфікаційної роботи міститься 9 рисунків і 4 таблиці.

Метою кваліфікаційної роботи є систематизація, класифікація евристичних правил детектування фішингових сайтів на основі їхніх характеристик та розробка ПЗ з метою покращення ефективності систем виявлення фішингу.

Об'єктом дослідження є процес виявлення фішингових вебсайтів у цифровому середовищі.

Предметом дослідження є методи застосування формалізованих евристичних правил для виявлення фішингових вебресурсів на основі їх структурних та поведінкових характеристик з метою підвищення ефективності їх детектування.

Для досягнення зазначеної мети були поставлені наступні завдання:

- визначити схеми сучасних фішингових атак;
- визначити ключові характеристики фішингових сайтів, що використовуються для їх виявлення;
- розробити класифікацію евристичних правил за основними критеріями;
- розробити програмне рішення використання окремих методів (правил) виявлення фішингових URL-адрес;

Практичною цінністю отриманих результатів є класифікація евристичних правил у вигляді ментальної мапи та програмне рішення, спрямоване на детектування фішингових URL-адрес.

Науковою новизною роботи є удосконалення класифікації евристичних правил для детектування фішингових сайтів, яка враховує багатофакторний підхід різних типів ознак.

Ключові слова: фішинг, фішингові сайти, евристичні правила, класифікація, легітимна URL-адреса, ефективність правил.

ЗМІСТ

ВСТУП 8
РОЗДІЛ 1. АНАЛІЗ ЕВРИСТИЧНИХ ПРАВИЛ ДЛЯ ВИЗНАЧЕННЯ ФІШИНГОВИХ URL 10
1.1. Соціальна інженерія та фішинг 10
1.2. Класифікація евристичних правил для визначення фішингових сайтів 14
1.3. Продуктивність правил та ймовірність їх спрацювання 43
Висновки за розділом 1 46
РОЗДІЛ 2. МОДЕЛЬ ВИЗНАЧЕННЯ ЙМОВІРНОСТІ ПРАВИЛЬНОГО ВИЗНАЧЕННЯ ФІШИНГОВИХ URL НА ОСНОВІ ПРАВИЛ 47
2.1. Структурно-логічна схема моделі визначення ймовірності фішингових вебсайтів 47
2.2. Реалізація моделі виявлення фішингових URL-адрес 50
2.3. Аналіз ймовірностей спрацювання евристичних правил 52
Висновки за розділом 2 54
РОЗДІЛ 3. ОЦІНКА ЕФЕКТИВНОСТІ ВИЗНАЧЕНИХ ФІШИНГОВИХ URL НА ОСНОВІ ПРАВИЛ 56
3.1. Ефективність спрацювання правил 56
3.2. Порівняльний аналіз методів виявлення фішингових сайтів 58
3.3. Інтеграція додаткових підходів та оптимізація евристичної моделі 62
Висновки за розділом 3 66
ВИСНОВКИ 68
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 71
ДОДАТКИ 75

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

IT – Information Technology

ПЗ – Програмне забезпечення

URL – Uniform Resource Locator

IP-адреса – Internet Protocol address

HTTP – Hyper Text Transfer Protocol

HTTPS – Hyper Text Transfer Protocol Secure

HTML – HyperText Markup Language

SSL – Secure Sockets Layer

ШІ – Штучний інтелект

ML – Машинне навчання

ВСТУП

Розвиток цифрових технологій та глобальної мережі Інтернет відкрив безпрецедентні можливості для обміну інформацією, комунікації, ведення бізнесу, надання державних та фінансових послуг. Проте разом із зростанням обсягів онлайн-активності зростає і рівень загроз, пов’язаних із використанням інформаційних технологій у злочинних цілях. Однією з найпоширеніших і водночас найнебезпечніших форм соціальної інженерії в кіберпросторі є фішинг – спроба отримати конфіденційні дані користувачів шляхом обману, здебільшого через створення підроблених вебсайтів, які імітують легітимні ресурси [28].

Зважаючи на масштабність і динамічність фішингових атак, а також їх здатність адаптуватися до нових умов, традиційні методи виявлення, що базуються лише на сигнатурному або чорному чи білому списках відомих доменів, часто виявляються недостатньо ефективними [23]. У відповідь на це зростає інтерес до методів, що дозволяють визначати шкідливу активність на основі поведінкових характеристик вебсайтів, зокрема через застосування евристичних правил, які враховують візуальні, структурні, мережеві та функціональні особливості фішингових ресурсів.

Незважаючи на наявність значної кількості досліджень у цій галузі, актуальним залишається питання узагальнення й упорядкування різноманітних евристичних підходів до детектування фішингових сайтів. Часто ці правила подаються розрізнено, без чіткої класифікації, що ускладнює їх застосування на практиці, а також інтеграцію у комплексні системи кіберзахисту.

У межах цієї кваліфікаційної роботи увагу зосереджено на систематизації та класифікації існуючих евристичних правил виявлення фішингових вебсайтів, що ґрунтуються на аналізі їхніх внутрішніх і зовнішніх характеристик. Запропоновано ментальну мапу як інструмент представлення

та впорядкування знань, що дозволяє ефективно структурувати наявні евристичні підходи. Такий підхід забезпечує можливість подальшої автоматизації процесу аналізу та розробки програмних рішень, які здатні оперативно реагувати на новітні виклики кіберзлочинності.

Таким чином, дана робота має як науково-методологічне, так і прикладне значення. Її результати та рекомендації згідно адаптації моделі можуть бути корисними фахівцям у сфері кібербезпеки, розробникам аналітичних систем та дослідникам, які працюють над удосконаленням методів захисту від фішингу та інших проявів інформаційних атак.

Апробація роботи. Основні результати роботи:

- були представлені у вигляді наукової роботи до Всеукраїнського конкурсу студентських наукових робіт з природничих, технічних та гуманітарних наук у 2025 році зі спеціальності «Кібербезпека»;
- були представлені на VIII Міжнародній науково-практичній конференції «Проблеми кібербезпеки інформаційно-комунікаційних систем (PCSICS)».

РОЗДІЛ 1
АНАЛІЗ ЕВРИСТИЧНИХ ПРАВИЛ ДЛЯ ВИЗНАЧЕННЯ ФІШИНГОВИХ URL

1.1. Соціальна інженерія та фішинг

Соціальна інженерія – це термін, що використовується для позначення широкого спектру зловмисних дій, що здійснюються через людську взаємодію, маніпулюючи людьми, змушуючи їх ділитися інформацією, якою вони не повинні ділитися, завантажувати програмне забезпечення, яке не повинні завантажувати, відвідувати вебсайти, які не повинні відвідувати, надсилати гроші злочинцям або робити інші помилки, які ставлять під загрозу їхню особисту або організаційну безпеку.

Незважаючи на те, що форма обману існувала завжди, вона значно еволюціонувала з розвитком технологій. У цьому новому контексті методи соціальної інженерії в IT можна розглядати під двома різними кутами:

- за допомогою психологічних маніпуляцій для отримання подальшого доступу до IT-системи, де знаходиться справжня мета шахрая. Як приклад, видаючи себе за важливого клієнта за допомогою телефонного дзвінка, заманити жертву до перегляду шкідливого вебсайту з метою зараження робочої станції жертви та отримання доступу до неї;
- за допомогою використання IT-технологій для підтримки методів психологічного маніпулювання для досягнення мети поза межами IT-сфери. Наприклад, отримання банківських реквізитів за допомогою фішингової атаки для подальшого викрадень грошей жертви та її конфіденційних даних.

Зростаюче використання IT-технологій природно призвело до збільшення використання таких методів, а також їх комбінування, до такої міри, що більшість кібератак сьогодні включають в себе ту чи іншу форму соціальної інженерії.

Фішинг – це форма соціальної інженерії, яка використовує шахрайські електронні листи, текстові повідомлення, телефонні дзвінки або вебсайти, щоб обманом змусити людей поділитися конфіденційними даними, завантажити шкідливе ПЗ або іншим чином наразити себе на небезпеку кіберзлочинності.

Атаки можуть статися як в особистому, так і в професійному житті та дуже часто відбуваються за однаковою схемою: кіберзлочинці видають себе за довірену особу чи організацію, щоб надсилати шахрайські повідомлення. Водночас вони маніпулюють емоціями читачів і створюють відчуття терміновості, наприклад, видають термінове попередження або можливість великої винагороди, щоб жертви не могли критикувати повідомлення. Фішинг може стосуватися не лише традиційної форми електронної пошти, але й інших методів, таких як голосовий фішинг – викрадення конфіденційних даних за допомогою телефонних дзвінків з незнайомих спеціально заготовлених номерів або голосової пошти і SMS-фішинг – надсилання шахрайських текстових повідомлень, які спрямовують користувачів на недобросовісні вебсайти через їх смартфони [13].

Основою для розуміння і класифікації різних методів, тактик і процедур, що використовуються зловмисниками під час кібератаки є матриця MITRE ATT&CK. Матриця ATT&CK була розроблена MITRE, некомерційною організацією, яка співпрацює з урядом та промисловістю для покращення кібербезпеки.

Матриця MITRE ATT&CK широко використовується у спільноті кібербезпеки як довідник для виявлення та реагування на кіберзагрози. Вона використовується аналітиками з безпеки, спеціалістами з реагування на інциденти та іншими фахівцями з кібербезпеки для кращого розуміння тактик та методів, що використовуються зловмисниками, для розробки більш ефективних стратегій захисту та покращення загального стану безпеки.

Матриця ATT&CK складається з двох основних компонентів: тактики та методи. Тактика відображає цілі зловмисника, в той час як методи – це конкретні техніки, що використовуються для досягнення цих цілей. Матриця

ATT&CK розділена на кілька категорій, кожна з яких представляє окремий етап кібератаки.

Одна з тактик називається "Initial Access" [10]. Вона включає в себе методи, які використовують різні вектори входу для початкового закріплення в мережі, тобто доступу до системи, який зберігається, незважаючи на переривання користувача, такі як перезавантаження, перебої в роботі або відключення мережі. Методи, що використовуються для закріплення, включають цілеспрямований спамфішинг і використання вразливостей на публічних вебсерверах [18].

У даній тактиці фішинг (T1566) є однією з технік і має ще чотири підтехніки, а саме:

1. "Phishing: Spearphishing Attachment". Зловмисник може надіслати електронного листа зі шкідливим вкладенням, намагаючись отримати доступ до системи жертви. Фішингові вкладення є особливим варіантом цільового фішингу. Фішингове вкладення відрізняється від інших форм фішингу тим, що в електронному листі використовується шкідливе ПЗ. Усі форми цільового фішингу – це соціальна інженерія, що доставляється в електронному вигляді, націлена на конкретну особу, компанію чи галузь. У цьому сценарії зловмисник прикріплює файл до фішингового електронного листа і зазвичай покладається на виконання користувачем, для досягнення мети у вигляді запущенного шкідливого ПЗ. Такий вид фішингу також може включати методи соціальної інженерії, такі як видавання за надійне джерело. Вкладення можуть бути різних типів, включаючи документи Microsoft Office, виконувані файли, заархівовані файли або PDF-файли. Після відкриття вкладення корисне навантаження зловмисника використовує вразливість або безпосередньо виконується в системі користувача. У тексті фішингового електронного листа часто намагаються надати правдоподібну причину, чому файл необхідно відкрити, і можуть пояснити, як обійти захист системи, щоб зробити це. Вони також можуть пояснювати, як розшифрувати вкладені файли, наприклад, паролі до ZIP-архівів, щоб обійти захист електронної пошти. Зловмисники

часто маніпулюють розширеннями файлів і значками, щоб прикріплені виконувані файли виглядали як файли документів, або файли, що використовують одну програму, виглядали схожими на файли для іншої програми.

2. "Phishing: Spearphishing Link". Зловмисники можуть надсилати шпигунські електронні листи зі шкідливим посиланням, намагаючись отримати доступ до систем жертви. Цілеспрямований фішинг із посиланням відрізняється від інших форм фішингу тим, що використовує посилання для завантаження шкідливого ПЗ, що міститься в електронному листі, замість того, щоб прикріпити шкідливі файли до самого листа, щоб обійти засоби захисту, які перевіряють вкладення електронної пошти. Як правило, посилання супроводжується текстом, що вимагає, щоб користувач активного натиснув або скопіював та вставив URL-адресу у свій браузер. Відвіданий вебсайт може скомпрометувати браузер за допомогою експлойта, або залежно від приводу в електронному листі, попросити користувача завантажити документ, програму, zip-файл або виконуваний файл.

3. "Phishing: Spearphishing via Service". Зловмисники можуть надсилати спам-повідомлення через сторонні сервіси, щоб отримати доступ до систем жертв, а не надсилати повідомлення безпосередньо через корпоративні канали електронної пошти. У цьому сценарії зловмисники надсилають повідомлення через різні соціальні мережі, особисті вебпоштові скриньки та інші сервіси поза контролем компанії. Ці сервіси, швидше за все, мають менш сувору та більш слабку політику безпеки, ніж підприємства. Потім зловмисник може надіслати шкідливі посилання або вкладення через ці сервіси. Поширеним прикладом є зв’язок з жертвою через соціальні мережі та надсилання контенту на особисту вебпошту, яку жертва використовує на своєму робочому комп'ютері. Таким чином зловмисник обходить деякі обмеження електронної пошти на робочому акаунті, і жертва з більшою ймовірністю відкриє файл, оскільки цього очікує. Якщо корисне навантаження

не працює так, як очікувалося, зловмисник може продовжувати комунікацію і попрацювати з жертвою над тим, як змусити його працювати.

4. "Phishing: Spearphishing Voice". Зловмисник може використати голосовий зв'язок, для отримання доступу до системи жертви. Голосовий фішинг є специфічним варіантом цілеспрямованого фішингу, що використовує маніпулювання користувачем для отримання доступу до систем за допомогою телефонного дзвінка або інших форм голосового зв'язку, покладаючись на User Execution, тобто виконання дії самою жертвою, для доставки та виконання. Зловмисники також можуть поєднувати голосовий фішинг з генерацією запитів на багатофакторну автентифікацію, щоб змусити користувачів розголошувати облікові дані або приймати запити на автентифікацію.

1.2. Класифікація евристичних правил для визначення фішингових сайтів

Поширені ознаки фішингу включають в себе: незнайоме привітання, небажані повідомлення, граматичні та орфографічні помилки, почуття терміновості, підозрілі посилання або вкладення, запити на отримання особистої інформації, невідповідності в електронних адресах, посиланнях, незвичайні запити, сповіщення про те, що ви щось виграли.

Якщо мова йде про основні ознаки, які вказують на необхідність негайно припинити спілкування з кимось, хто, можливо, намагається шахрайським шляхом отримати вашу конфіденційну інформацію, то це лише базовий рівень захисту від фішингових атак. Однак існують більш складні та спеціалізовані способи виявлення потенційно небезпечних вебсайтів, які можуть бути використані для фішингу. Одним із підходів є застосування спеціальних евристичних правил для аналізу URL-адрес.

Методи евристичного аналізу дозволяють виявляти фішингові сайти на основі багатьох характеристик, таких як нетипова структура посилань, наявність підозрілих символів або неправильної послідовності слів у

доменних іменах. Такі правила базуються на дослідженні шаблонів, які характеризують фішингові вебсайти, і дозволяють користувачам попереджати про потенційні загрози до того, як вони натиснуть посилання.

Для кращого розуміння правил їх можна класифікувати у декілька груп, а саме “Address Bar based Features”, “HTML and JavaScript based features”, “Domain based features”, “Abnormal based features” [24].

Перша група – це можливості на основі адресного рядка (Address Bar based Features).

1.1) Використання IP-адреси (Using the IP Address).

Зловмисники часто використовують IP-адреси замість доменних імен, щоб приховати шкідливі вебсайти та змусити користувачів повірити, що вони відвідують легальні сайти. Хоча доменні імена можна легко ідентифікувати або перевірити, IP-адреси, особливо коли вони перетворені в шістнадцяткову форму числення, ускладнюють для користувачів визначення потенційних загроз. З великою вірогідністю ці посилання ведуть на фішингові вебсайти, призначені для збору такої конфіденційної інформації, як фінансових відомостей, облікових даних для входу, чи інших особистих даних користувачів.

Використовуючи цей метод, зловмисники прагнуть приховати справжню незаконну мету даного посилання, збільшуючи ймовірність того, що користувачі натиснуть на нього, не усвідомлюючи небезпеки. Тобто якщо в URL-адресі IP-адреса використовується як альтернатива доменному імені, наприклад, "http://192.168.3.67/fake.html", користувачі можуть бути впевнені, що хтось намагається викрасти їхню особисту інформацію. Ще один схожий розповсюджений випадок, це коли IP-адреса перетворюється на шістнадцятковий код, як показано в наступному прикладі: "http://0xC2.0xA8.0x58.0xCC/2/paypal.ca/index.html" [16].

Правило:

Якщо частина домену має IP-адресу → Фішинг

У іншому випадку → Легітимна URL-адреса.

1.2) Довга URL-адреса для приховування підозрілої частини (Long URL to Hide the Suspicious Part).

Ще одним часто використовуваним методом є використання довгих URL-адрес з метою приховати підозрілі елементи в адресному рядку. Наприклад, фішингова URL-адреса може виглядати так:

"http://privat.bank.com.br/5f/app/574ab37c984d048f/?cmd=_home&dispatch=938bbc4893d0876a78494f90484684d78652aa734b47821bc3837d8@phishing.website.html"

Ця URL-адреса містить наче і законний домен на початку, але довший шлях приховує підозрілі частини нижче. Для того, щоб краще зрозуміти зв'язок між довжиною посилання та ймовірністю фішингу, було розраховано середню довжину URL-адрес у наборі даних. Аналіз показав, що URL-адреси довжиною 54 символи та більше були класифіковані як ймовірні спроби фішингу. З набору даних у 1220 URL-адрес мали довжину 54 символи або більше, що склало 48,8% від загальної кількості даних. Ця значна частина свідчить про те, що фішери часто покладаються на довгі URL-адреси, щоб замаскувати зловмисні наміри та уникнути виявлення.

Правило:

Якщо довжина URL-адреси < 54 → Легітимна URL-адреса

Якщо довжина URL-адреси ≥ 54 та ≤ 75 [21] → Підозріла URL-адреса

У іншому випадку → Фішинг.

1.3) Використання служб скорочення URL-адрес "TinyURL" (Using URL Shortening Services "TinyURL").

Задля того, щоб сховати фішингову адресу, зловмисники використовують метод скорочення URL-адрес. Цей метод зазвичай використовується, щоб зробити довгі та складні URL-адреси легшими для керування, особливо в контекстах, де простір обмежений, наприклад у публікаціях у соціальних мережах чи на інших платформах обміну повідомленнями. Цей процес створення фішингової адреси передбачає створення скороченої версії вихідної URL-адреси, яка все ще переспрямовує

користувачів на заплановану зловмисну вебсторінку. Це досягається за допомогою механізму «HTTP Redirect», де для представлення довшої URL-адреси використовується скорочене доменне ім’я.

Наприклад, URL-адресу на зразок "http://portal.hud.ac.uk/" можна скоротити до набагато простішої версії, наприклад "bit.ly/23DYPs7". Скорочена URL-адреса візуально привабливіша, нею легше поділитися в Інтернет-просторі з цільовими до фішингу користувачами та менш схильна до помилок при введенні вручну. Незважаючи на коротший формат, користувач плавно перенаправляється на оригінальну довгу URL-адресу, коли натискає або вводить скорочену версію.

Правило:

Якщо "TinyURL" → Фішинг

У іншому випадку → Легітимна URL-адреса.

1.4) URL-адреса містить символ «@». (URL’s having “@” Symbol).

Наразі використання символу «@» в URL-адресі є досить старою угодою, головним чином призначеною для вбудовування облікових даних для входу (логін та пароль) у URL-адресу, наприклад "http://username:password@website.com". Однак ця практика значною мірою вийшла з ладу через проблеми безпеки, а сучасні браузери зі встановленими на них механізмами безпеки зазвичай видають попередження, коли зустрічаються такі URL-адреси.

Однак зловмисники знайшли спосіб використовувати символ «@» у фішингових цілях. Якщо в URL-адресу включено символ «@», багато браузерів можуть проігнорувати все, що йому передує, і оброблятимуть лише ту частину URL-адреси, яка йде після символу «@».

Наприклад, можна розглянути наступну URL-адресу: "http://legitimate-site.com@phishing-site.com/login". На перший погляд може здатися, що вона походить із надійного джерела, оскільки користувачі зазвичай зосереджуються на частині «legitimate-site.com». Насправді браузер ігнорує все, що передує символу «@», і спрямовує користувача на "phishing-site.com",

який, скоріше за все, є шкідливим сайтом, створеним для викрадення конфіденційної інформації користувача, що перейшов за данним посиланням.

Правило:

Якщо URL-адреса містить символ «@» → Фішинг

У іншому випадку → Легітимна URL-адреса.

1.5) Переспрямування за допомогою “//” (Redirecting using “//”).

Наявність подвійних скісних риск ("//") в URL-адресі служить певній меті. Дана комбінація символів відокремлює протокол (наприклад, HTTP або HTTPS) від решти URL-адреси, сигналізуючи про початок доменного імені. Однак якщо після першого звичного користувачам «http://» або «https://» слідує інший «http://» або «https://» у межах шляху, це може вказувати на переспрямування на інший вебсайт, що є технікою для проведення фішингових атак. Наприклад, URL-адреса на кшталт "http://www.legitimate.com//http://www.phishing.com" спочатку може вести на законний сайт, але зрештою переспрямовує користувача на зловмисний "http://www.phishing.com".

Щоб краще зрозуміти цю поведінку, було проаналізовано розташування "//" в URL-адресах. Якщо URL-адреса починається з "HTTP", перше входження "//" має бути на шостій позиції, одразу після протоколу, наприклад, у "http://". Для URL-адрес, які використовують "HTTPS", "//" має відображатися на сьомій позиції після "https://". Якщо додатковий «//» з’являється пізніше в URL-адресі, особливо після доменного імені, це часто є тим самим червоним прапорцем, що сигналізує про небезпеку для користувача у вигляді переспрямування на інший сайт.

Правило:

Якщо позиція останнього входження "//" " в URL-адресі > 7 → Фішинг

У іншому випадку → Легітимна URL-адреса.

1.6) Додавання до домену префікса або суфікса, розділених (-) (Adding Prefix or Suffix Separated by (-) to the Domain).

Символ тире ("-") насправді дуже рідко зустрічається в легітимних URL-адресах, особливо в основному доменному імені. Проте зловмисники часто користуються цим ходом, вставляючи тире в домен. Ця тактика передбачає додавання префіксів або суфіксів до добре відомих доменних імен, розділених тире, щоб створити ілюзію легітимності. Не кожен користувач помітить цей маленький символ, який повністю відрізняє фішингову URL-адресу від легітимної. Наприклад, адреса "http://www.Confirme-paypal.com/", на перший погляд може виглядати так, ніби вона належить PayPal, але додавання префікса "Confirme-" є оманливою модифікацією, призначеною для обману користувачів [16].

У більшості випадків законні компанії та організації уникають використання тире у своїх основних доменних іменах, віддаючи перевагу простішим і професійнішим структурам URL-адрес. Тому зустріч з тире в основній частині URL-адреси, особливо в поєднанні з впізнаваною назвою бренду, має викликати підозру [5].

Правило:

Якщо частина імені домену містить символ (-) → Фішинг

У іншому випадку → Легітимна URL-адреса.

1.7) Субдомен і кілька субдоменів (Sub Domain and Multi Sub Domains).

У контексті аналізу URL-адрес розуміння того, як розбити доменне ім’я на різні компоненти, має вирішальне значення для виявлення потенційних спроб фішингу. Наприклад, у посиланні "http://www.ukraine.com.ua", структура доменного імені відповідає певній ієрархії.

Міжнародні чи загальні (gTLD) домени у даному прикладі це com (commercial), що підходить для міжнародних комерційних компаній та інтернет-магазинів. Національні (ccTLD) домени позначають країни сайтів та компаній, тобто .ua (Україна) і цей домен доступний тільки для зареєстрованих

торгових марок і ім'я має містити точну назву торгової марки. Домен другого рівня містить уже абревіатуру або назву самого сайту.

Щоб систематично витягувати та класифікувати URL-адреси на основі цієї структури, необхідно:

По-перше, пропускати префікс «www», який є субдоменом і не впливає на оцінку основного доменного імені. Видалення цього дозволить нам зосередитися на основних компонентах URL-адреси.

По-друге, видалити ccTLD, якщо URL-адреса містить домен верхнього рівня з кодом країни, наприклад «.uk», «.ua» або інше.

Наостанок, порахувати крапки, що залишилися після пропуску «www» і видалення ccTLD, що дає уявлення про кількість наявних субдоменів, що є ключовим показником того, чи може URL-адреса бути підозрілою чи шкідливою.

Якщо кількість крапок перевищує одну, URL класифікується як «Підозрілий». Це означає, що існує один субдомен, що не характерно для більшості законних вебсайтів. Якщо кількість крапок більше двох, це класифікується як «фішинг».

Це правило ґрунтується на передумові, що законні URL-адреси мають простішу структуру, тоді як фішингові URL-адреси часто використовують кілька субдоменів, щоб заплутати або обдурити користувачів. Наприклад, фішингове посилання може виглядати як "http://login.security.paypal.com/", де «login» і «security» — субдомени, створені для того, щоб ввести користувача в оману, змусивши його повірити, що він отримує доступ до безпечної сторінки входу PayPal.

Правило:

Якщо кількість крапок в частині домену = 1 → Легітимна URL-адреса.

Якщо кількість крапок в частині домену = 2 → Підозріла URL-адреса.

У іншому випадку → Фішинг.

1.8) HTTPS (Hyper Text Transfer Protocol with Secure Sockets Layer).

Наявність HTTPS в URL-адресі, хоча й необхідна для встановлення безпечного з'єднання, не є остаточним показником легітимності вебсайту. Зловмисники все частіше почали використовувати HTTPS, щоб дати користувачам хибне відчуття безпеки, що призводить до помилкового уявлення про те, що саме існування HTTPS гарантує надійний сайт. Але наразі для підтвердження легітимності сайту потрібна більш ретельна перевірка, зокрема щодо сертифіката SSL/TLS, призначеного вебсайту [4].

Щоб краще оцінити легітимність сайту, важливо оцінити два ключові аспекти. По-перше, це надійність центру сертифікації, тобто що сертифікат вебсайту видається центром сертифікації, і рівень довіри залежить від його репутації. Не всі центри сертифікації однаково надійні, деякі з них більш схильні до видачі сертифікатів зловмисникам. Центри сертифікації найвищого рівня, які постійно визнані своєю надійністю, включають [20] Comodo SSL, GeoTrust, GlobalSign, GoDaddy, DigiCert, Thawte, Network Solutions, RapidSSL, і VeriSign. Сертифікати від цих органів вважаються більш надійними показниками законного вебсайту.

По-друге, ще одним фактором, який слід враховувати, є вік сертифіката SSL. Щойно видані сертифікати іноді можуть бути тривожним сигналом, оскільки зловмисники, як правило, використовують короткочасні та зовсім нещодавно отримані сертифікати для здійснення атак, перш ніж ці фішингові вебсайти розкриють та занесуть у чорний список. Навпаки, законні вебсайти часто мають довгострокові сертифікати у декілька років. Проаналізувавши наш набір даних, було виявлено, що авторитетні сертифікати зазвичай мають мінімальний вік двох років. Це свідчить про те, що нові сертифікати, особливо тим, яким менше року, можуть вимагати більш ретельного вивчення.

Тож тактика зловмисників – це використання недорогих або повністю безкоштовних сертифікатів від менш авторитетних центрів сертифікації, при тому що походження та тривалість сертифіката є критично важливими для визначення надійності сайту.

Правило:

Якщо використовується HTTPS, емітент надійний і вік сертифіката ≥ 1 року → Легітимна URL-адреса

Якщо використовується HTTPS та емітент ненадійний → Підозріла URL-адреса

У іншому випадку → Фішинг.

1.9) Наявність маркера «HTTPS» у доменній частині URL-адреси (The Existence of “HTTPS” Token in the Domain Part of the URL).

Щоб збити з пантелику користувачів, зловмисники часто використовують їх довіру до безпечних протоколів, вбудовуючи оманливі маркери, такі як «HTTPS», безпосередньо в доменну частину URL-адреси. Ця стратегія випливає з загальноприйнятої думки, що «HTTPS» в URL-адресі пов’язаний із безпекою, однак наявність «HTTPS» у доменному імені насправді є тривожним сигналом. Типова фішингова URL-адреса, що використовує цей трюк, може виглядати приблизно так: "http://https-www-paypal-it-mpp-home.soft-hair.com/". Спочатку ця URL-адреса може здатися законною через включені в неї «https» і «paypal», але подальший аналіз показує, що вона насправді є шахрайською.

Цей метод особливо вигідний, оскільки багато користувачів не розуміють структуру URL-адрес і можуть не помітити сам домен, замість цього зосереджуючись на загальних "безпечних" компонентах, розташованих у субдомені. Токен «HTTPS» вставляється для посилення відчуття безпеки, навіть якщо саме з'єднання не шифрується, на що вказує використання «HTTP», а не «HTTPS» в розділі протоколу URL-адреси.

Правило:

Якщо використовується маркер HTTPS в доменній частині URL-адреси → Фішинг

У іншому випадку → Легітимна URL-адреса.

Другою групою в класифікації правил детектування фішингових сайтів є ненормальні функції (Abnormal Based Features).

2.1) URL-адреса запиту (Request URL).

URL-адреса запиту пов’язана з дослідженням зовнішніх ресурсів, як-от зображень, відео та звукових файлів, вбудованих у вебсторінку. Ці об’єкти зазвичай розміщуються в одному домені на законних вебсайтах, оскільки це означає, що сайт керує власним вмістом. Однак фішингові вебсайти часто беруть ці об’єкти з зовнішніх джерел, щоб приховати їх справжню особу або уникнути виявлення.

Коли вебсторінка завантажується, різні зовнішні ресурси (наприклад, зображення, таблиці стилів, сценарії, звукові файли або відео) можуть бути отримані з різних доменів. Функція Request URL обчислює відсоток цих об’єктів, отриманих з доменів, відмінних від домену вебсайту. Якщо значна частина цих об’єктів розміщена на зовнішньому хостингу, це може означати, що сайт намагається ввести користувачів в оману.

Створюючи правило, необхідно зазначити, що на законних вебсайтах переважна більшість ресурсів, таких як мультимедійні файли, сценарії та інше ресурси, зазвичай обслуговуються з того самого домену, що й сама вебсторінка. Це свідчить про послідовність у використанні домену та надійність.

На підозрілих вебсайтах деякий зовнішній вміст може бути законним, як, наприклад, вбудовані відео, реклама чи інтеграція в соціальні мережі. Однак, якщо більш ніж мінімальна частка зовнішніх ресурсів з різних доменів, це може бути червоним прапорцем.

У той час фішингові вебсайти часто завантажують великий відсоток своїх ресурсів із різних доменів. Це частина стратегії, спрямованої на те, щоб сайт виглядав легітимним, одночасно зводячи до мінімуму витрати чи зусилля на розміщення ресурсів.

Для реалізації необхідно проаналізувати вихідний код HTML вебсторінки, щоб визначити всі теги `<img>`, `<script>`, `<video>`, `<audio>` та інші медіа-теги, які посилаються на зовнішні домени та порівняти домени, з яких завантажуються ці ресурси, з основним доменом вебсторінки. Після

чого підрахувати у відсотках, скільки зовнішніх доменів використовується для завантаження об’єктів.

Правило:

Якщо % Request URL < 22% → Легітимна URL-адреса

Якщо % Request URL ≥ 22% і < 61% → Підозріла URL-

У іншому випадку → Фішинг. адреса

2.2) Анкор посилання (URL of Anchor).

Дана властивість перевіряє URL-адреси, вказані в тегах `<a>`, щоб визначити, чи становлять вони загрозу безпеці користувачів. Якорі - це елементи HTML, які використовуються для створення гіперпосилань, що перенаправляють користувачів до різних місць на вебсторінці або за її межами. Аналіз URL-адрес у цих тегах допомагає виявити потенційні спроби фішингу, особливо якщо ці URL-адреси відрізняються від домену вебсторінки або містять підозрілі шаблони.

Якщо URL-адреса в тезі якоря вказує на домен, відмінний від основного домену вебсторінки, це може свідчити про те, що посилання перенаправляє користувачів на зовнішній, потенційно шкідливий, сайт. Це схоже на аналіз URL-адрес запитів, де зовнішні домени можуть бути позначені як підозрілі.

Також ще однією властивістю є анкори, які не посилаються на дійсну вебсторінку або зовнішній сайт, наприклад:

- <a href="#" >
- <a href="#content" >
- <a href="#skip" >
- <a href="JavaScript::void(0)" >

Згідно проведених досліджень, якщо менше 31% анкорів вказують на зовнішні домени або не є посиланнями, вебсторінка класифікується як легальна. Це означає, що більшість тегів якоря або є посиланнями в межах одного домену, або слугують справжнім навігаційним цілям.

Якщо від 31% до 67% анкорів вказують на зовнішні домени або не мають посилань, вебсторінка класифікується як підозріла. Це свідчить про помірний

рівень зовнішніх посилань або нефункціональних посилань, що може вимагати подальшого розслідування.

Якщо більше 67% URL-адрес посилань ведуть на зовнішні домени або не мають посилань, вебсторінка класифікується як фішингова. Високий відсоток зовнішніх або нефункціональних посилань часто характерний для фішингових сайтів, призначених для обману користувачів або перенаправлення їх на шкідливий контент.

Правило:

Якщо % of URL Of Anchor < 31% → Легітимна URL-адреса

Якщо % of URL Of Anchor ≥ 31% і ≤ 67% → Підозріла URL-адреса

У іншому випадку → Фішинг.

2.3) Посилання в тегах <Meta>, <Script> і <Link> (Links in <Meta>, <Script> and <Link> tags).

Посилання в тегах <Meta>, <Script> і <Link> є важливими компонентами HTML-структури вебсторінки. Законні вебсайти зазвичай використовують ці теги для покращення функціональності, завантаження зовнішніх ресурсів або надання метаданих про сторінку. Однак фішингові вебсайти часто зловживають цими тегами, посилаючись на зовнішні або шкідливі домени, що може поставити під загрозу безпеку сторінки.

Метатеги надають метадані про вебсторінку, такі як описи або ключові слова, які використовують пошукові системи та браузери. Законні сайти зазвичай пов’язують мета-теги з ресурсами в межах одного домену, а також вони використовуються для вбудовування або посилання на зовнішні файли JavaScript, які запускаються на стороні клієнта. Легітимні вебсайти часто мають сценарії з довірених доменів, у той час фішингові можуть посилатися на шкідливі сценарії, розміщені на зовнішньому хості для виконання шкідливих дій.

Якщо відсоток посилань у тегах <Meta>, <Script> і <Link>, які вказують на зовнішні домени, становить менше 17%, вебсайт класифікується як законний, що свідчить, що більшість ресурсів розміщено в одному домені, що

типово для безпечних легітимних вебсайтів. Якщо відсоток зовнішніх посилань у цих тегах становить від 17% до 81%, вебсайт класифікується як підозрілий, але в деяких випадках може бути законним, то ж потребує більш ретельного вивчення. Але якщо відсоток зовнішніх посилань перевищує 81%, вебсайт, швидше за все, з великою вірогідністю є фішинговим.

Правило:

Якщо % посилань в "<Meta>", "<Script>" і "<"Link> < 17% → Легітимна URL-адреса

Якщо % посилань в "<Meta>", "<Script>" і "<"Link> ≥ 17% і ≤ 81% → Підозріла URL-адреса

Інакше→ Фішинг.

2.4) Серверний обробник форм (Server Form Handler (SFH)).

Ключовим аспектом вебформ є серверний обробник форм, де дані, надіслані користувачами, надсилаються на обробку. Коли користувачі надсилають особисту або конфіденційну інформацію через вебформу, SFH визначає, куди ця інформація буде спрямована. Легітимні вебсторінки зазвичай гарантують, що SFH спрямовує дані на надійний сервер у тому самому домені, тоді як фішингові вебсайти часто маніпулюють цим механізмом із бажаною зловмисною метою. Форма, яка містить порожній SFH або обробник дії зі значенням "about:blank", викликає підозру, оскільки не вказує чіткого напрямку, куди саме надсилатимуться дані. Це тактика, яку часто використовують зловмисники, щоб тимчасово обійти механізми виявлення. Якщо в обробнику форми не вказано законну дію, вебсторінка, ймовірно, є спробою фішингу.

Ще одним можливим показником фішингу є те, коли домен у SFH відрізняється від домену вебсторінки. Законні вебсайти досить рідко надсилають дані користувача в інший домен, оскільки це було б порушенням стандартів безпеки. Якщо SFH вказує на зовнішній домен, це може означати, що дані користувача пересилаються на підозрілий сторонній сервер, що з великою ймовірністю контрольований зловмисниками.

Правило:

Якщо SFH є «about: blank\» або порожній → Фішинг

Якщо SFH «відноситься до іншого домену → Підозріла URL-адреса

У іншому випадку → Легітимна URL-адреса.

2.5) Надсилання інформації на електронну пошту (Submitting Information to Email).

Надсилання інформації електронною поштою – це тактика фішингу, коли особиста інформація, введена у вебформу, перенаправляється не на безпечний сервер, а на вказану адресу електронної пошти. Зловмисники можуть використовувати мови сценаріїв на стороні сервера, як-от PHP, щоб отримувати дані форми та надсилати їх на свою електронну пошту. Зазвичай для цього використовується функція "mail()", яка надсилає введені користувачем дані безпосередньо на електронну пошту зловмисника.

У деяких випадках фішингові вебсайти можуть використовувати іншу функцію "mailto:", яка є клієнтським методом спрямування надсилання форми на електронну адресу. Тож для забезпечення власної безпеки та безпеки інших користувачів необхідно проаналізувати вихідний код вебсторінки, щоб перевірити наявність функції `mail()` у сценаріях на сервері або посилання `mailto:` у надсиланні форм.

Правило:

Якщо використовуються функції «mail()» або «mailto:» для відправки інформації про користувача → Фішинг

У іншому випадку → Легітимна URL-адреса.

2.6) Аномальна URL-адреса (Abnormal URL).

Аномальна URL-адреса – це посилання, яке не містить очікуваного імені хоста або іншого ідентифікатора вебсайту. На легальних вебсторінках URL-адреса зазвичай містить доменне ім'я або частину хоста, яку можна ідентифікувати. Однак фішингові вебсайти часто використовують незвичайні URL-адреси, щоб заплутати користувачів, опускаючи ім'я хоста, додаючи

оманливі назви або використовуючи складні або непов'язані між собою доменні структури. Цю інформацію можна отримати з баз даних WHOIS [29].

Правило:

Якщо ім'я хоста не включено в URL → Фішинг

У іншому випадку → Легальна URL-адреса.

Третя група класифікується як властивості фішингових посилань на базі HTML та JavaScript (HTML and JavaScript based Features)

3.1) Переадресація вебсайту (Website Forwarding).

Фішингові сайти часто використовують кілька переспрямувань, щоб приховати свої зловмисні наміри. Вони проводять користувачів через низку проміжних сторінок, перш ніж потрапити на кінцеву оманливу сторінку. Ця техніка, яка використовується для приховування справжнього призначення URL-адреси, що ускладнює користувачам перевірку законності сайту, який вони відвідують, відома як перенаправлення сайту. Даний метод може ефективно замаскувати справжню природу вебсайту та змусити користувачів надати конфіденційну інформацію.

Фішингові вебсайти можуть використовувати послідовність перенаправлень, щоб замаскувати справжній кінцевий пункт призначення. Це може включати кілька рівнів пересилання, кожен з яких потенційно приховує шахрайський характер сайту.

Щоб ідентифікувати фішинг на основі переспрямування, слід відстежити шлях URL-адреси та підрахувати кількість переадресацій. Це може свідчити про наміри сайту: менша кількість перенаправлень свідчить про законний сайт, тоді як більша – про потенційний фішинг. Якщо вебсайт містить більше трьох переадресацій, він класифікується як фішинговий. У випадку вебсторінки з двома-трьома перенаправленнями, він класифікується як підозрілий, хоча цей рівень переспрямування ще не є остаточним доказом фішингу, він вимагає подальшого розслідування. Вебсайти з невеликою кількістю перенаправлень або взагалі без них класифікуються як легальні [24].

Правило:

Якщо сторінок перенаправлення ≤ 1 → Легітимна URL-адреса

Якщо сторінок перенаправлення ≥ 2 та < 4 → Підозріла URL-адреса

У іншому випадку → Фішинг.

3.2) Налаштування рядка стану (Status Bar Customization).

Ще один прийом, який використовують зловмисники, щоб змусити користувачів повірити, що вони взаємодіють із законним вебсайтом – це маніпулювання рядком стану – невеликою областю внизу вікна вебпереглядача, яка зазвичай відображає URL-адресу посилання, на яке наводиться курсор. Зловмисники у даній схемі зазвичай представляють фальшиву URL-адресу. Ця тактика має на меті змусити користувачів подумати, що вони перебувають на надійному сайті, навіть якщо вони насправді знаходяться на фішинговій сторінці.

Для цієї мети зазвичай використовується подія "onMouseOver". Вставляючи на вебсторінку код, який змінює текст рядка стану. Тому, щоб перевірити, чи були зроблені зміни, необхідно перевірити саме подію "onMouseOver".

Правило:

Якщо "onMouseOver" змінює рядок стану → Фішинг

У іншому випадку → Легітимна URL-адреса.

3.3) Вимкнення клацання правою кнопкою миші (Disabling Right Click).

Як було вже зазначено, зловмисники часто використовують JavaScript у власних корисних цілях. Дана властивість попереджає користувачів про можливе вимкнення функції натискання правою кнопкою миші на вебсторінках, не даючи користувачам отримати доступ до контекстного меню, яке дозволяє їм переглядати та перевіряти вихідний код. Вимикаючи дану функцію, зловмисники намагаються ускладнити користувачам перевірку структури та вмісту сторінки, що може допомогти виявити справжню природу вебсайту.

Цей метод обману схожий на попередню тактику використання "onMouseOver", щоб приховати посилання. У цьому випадку вимкнення функції клацання правою кнопкою миші додає ще один рівень обфускації, що ускладнює для технічно підкованих користувачів або аналітиків безпеки дослідження наміру вебсторінки. Для цієї функції необхідно знайти подію "event.button==2" у вихідному коді вебсторінки і перевірити, чи відключено клацання правою кнопкою миші.

Правило:

Вимкнено правою кнопкою миші → Фішинг

У іншому випадку → Легітимна URL-адреса.

3.4) Використання спливаючих вікон (Using Pop-up Window).

Поширеною тактикою для обману користувачів і збору особистої інформації є використання спливаючих вікон. Спливаюче вікно – це менше за розміром вікно, яке раптово з’являється на передньому плані вебсторінки, яку переглядає користувач. Хоча законні вебсайти також використовують спливаючі вікна для відображення оголошень, реклами, попереджень або іншої додаткової інформації, вони не вимагають користувачів ввести через ці вікна конфіденційні дані. Тим часом зловмисники використовують спливаючі вікна, вбудовуючи форми, які просять користувачів надати особисту інформацію. Ці спливаючі вікна часто з’являються як частина того, що виглядає як законний вебсайт, що спонукає користувачів вірити у безпеку власних даних.

Правило:

Якщо спливаюче вікно містить текстові поля → Фішинг

У іншому випадку → Легітимна URL-адреса.

3.5) Перенаправлення IFrame (IFrame Redirection).

Використання перенаправлення IFrame – це техніка, яку зазвичай використовують для вставки шкідливого вмісту з однієї вебсторінки на іншу. IFrame (Inline Frame) — це тег HTML, який дозволяє вбудовувати іншу

вебсторінку в поточну вебсторінку, фактично завантажуючи зовнішній вміст без відома користувача.

За допомогою IFrame завантажується фішинговий вміст із зовнішнього джерела, при цьому не змінюючи зовнішній вигляд вебсторінки, що ускладнює його виявлення для користувачів. Один з поширених трюків передбачає встановлення атрибута frameBorder на нуль, що робить рамку невидимою для користувача, завдяки чому зловмисний вміст виглядає як частина законної вебсторінки, створюючи безперебійну ілюзію, що користувачі взаємодіють із запланованим сайтом [2].

Правило:

Якщо використовується IFrame → Фішинг

У іншому випадку → Легітимна URL-адреса.

3.6) Іконка (Favicon).

"Favicon" – це невелике графічне зображення, яке зазвичай відображається в адресному рядку браузера, закладках або вкладках, і слугує візуальним представленням ідентичності вебсайту. Іконка є ключовою частиною брендингу сайту і часто допомагає користувачам швидко розпізнати вебсайт, який вони відвідують. Більшість легальних вебсайтів розміщують свої іконки на тому ж самому домені, що й решту контенту, що посилює зв'язок між іконкою та ідентичністю вебсайту.

Однак у спробах фішингу зловмисники мають тенденцію використовувати іконку, розміщену на іншому домені, ніж той, що відображається в адресному рядку браузера. Ця розбіжність може бути вагомим індикатором шахрайської діяльності, оскільки ще одна мета фішингових сайтів – намагатись імітувати зовнішній вигляд легальних сайтів щоб уникнути виявленню. А оскільки більшість користувачів покладаються на візуальні підказки, такі як логотипи та іконки, для перевірки автентичності сайту, ця тактика є особливо ефективною у фішингових схемах.

Правило:

Якщо завантажено іконку із зовнішнього домену → Фішинг

У іншому випадку → Легітимна URL-адреса.

3.7) Відсутній тег (Missing Title).

Тег заголовка HTML дуже важливий для легального вебсайту, оскільки він визначає назву вебсторінки, яка відображається у вкладках браузера та результатах пошукової видачі. Це сприяє як зручності користування, так і SEO (пошуковій оптимізації). Відсутність тегів заголовків і погана структура часто є характерними ознаками фішингових сайтів. Зловмисники, як правило, опускають або ігнорують теги заголовків, зосереджуючись на оманливому контенті вебсайту.

Фішингові сайти часто мають відсутні, неточні або нерелевантні назви, щоб обманом змусити користувачів ввести особисту інформацію. Ці заголовки можуть бути порожніми, заповненими випадковими символами, що ніяк не пов'язані з сайтом або іншими нерелевантними фразами. Це тому, що зловмисники не піклуються про оптимізацію пошукової системи чи досвід користувача.

Правило:

Якщо тег title відсутній, порожній або нерелевантний вмісту вебсторінки → Фішинг

У іншому випадку → Легітимна URL-адреса.

3.8) HTML-посилання на сторонні ресурси (HTML links to third-party resources like Google Analytics, Facebook, Cloudflare, etc).

Деякі методи намагаються зробити так, щоб фішингові вебсайти виглядали легітимними, шляхом включення HTML-посилань на сторонні ресурси від перевірених постачальників, таких як Google Analytics, Facebook, Cloudflare та інші. Ці джерела часто використовуються легальними сайтами для відстеження поведінки користувачів, надання зовнішнього контенту і підвищення продуктивності вебсторінок.

Включаючи відомі сторонні служби, зловмисники намагаються приховати свій шкідливий вміст у морі законного коду. Інструменти безпеки, якщо вони неправильно налаштовані, можуть не помітити ці сайти через

наявність надійних сторонніх елементів. Наприклад, на сторінці можуть бути вбудовані служби відстеження або плагіни соціальних мереж, як кнопки «Подобається» у Facebook або менеджер тегів Google, щоб приховати факт крадіжки конфіденційної інформації. А такий інструмент, як Google Analytics, використовують не для законних бізнес-цілей, а для відстеження кількості користувачів, які взаємодіяли з їхнім фішинговим сайтом або надсилали особисті дані.

Правило:

Якщо вебсторінка містить велику кількість посилань на сторонні ресурси, які не відповідають домену вебсторінки → Фішинг

У іншому випадку → Легітимна URL-адреса.

3.9) Довжина <Body> в тегах (<Body> length in tags).

Зловмисники, щоб обійти механізми виявлення, часто маніпулюють довжиною тегу <body>. Тег <body> у документі HTML містить основний вміст вебсторінки, включаючи текст, зображення, форми та посилання. Поширені фішингові методи штучно збільшують, додавши невидимий вміст, наприклад, прихований текст, порожні елементи div або невикористовувані сценарії, або зменшують довжину вмісту, всередині тегу з метою обійти алгоритми сканування на зловмисний характер сайту.

Фішингові вебсайти можуть маніпулювати тегом body, щоб вставити вміст, який виходить далеко за межі відображуваного екрана. Цей прийом використовується для відволікання уваги від шкідливих елементів або для приховування шкідливих посилань у нижній частині сторінки, що робить їх менш помітними для користувача.

Правило:

Якщо довжина тегу <body> ненормально довга або коротка → Фішинг

У іншому випадку → Легітимна URL-адреса.

Останньою групою класифікації є властивості на основі домену (Domain based Features).

4.1) Тривалість реєстрації домену (Domain Registration Length).

З огляду на те, що фішингові вебсайти зазвичай розроблені для короткочасного використання, вони рідко активні протягом тривалого часу. Задля того, щоб уникнути виявлення, зловмисники створюють і швидко залишають ці сайти. У результаті, шахрайські домени, як правило, мають набагато коротші періоди реєстрації порівняно з легальними. Надійні домени, з іншого боку, часто оплачуються та обслуговуються на кілька років наперед, що відображає їхню довгострокову відданість користувачам і бізнес-операціям.

Під час аналізу набору даних було виявлено різкий контраст між тривалістю життя законних і фішингових доменів. Найдовші шахрайські домени були зареєстровані не більше одного року. Навпаки, авторитетні вебсайти, особливо ті, що належать відомим підприємствам, навчальним закладам чи державним установам, часто захищають свої домени протягом кількох років, або вже десятків років поспіль. Ця довгострокова інвестиція не тільки забезпечує безперервний доступ для користувачів, але й свідчить про легітимність і надійність вебсайту.

Таким чином, різниця в періодах реєстрації домену може служити цінним показником при оцінці легітимності вебсайту. Хоча щойно зареєстрований або короткостроковий домен не завжди є шахрайським, поєднання короткого терміну служби з іншими ознаками, такими як підозрілі структури URL-адрес або відсутність надійного сертифіката SSL, може свідчити про потенційну фішингову діяльність.

Правило:

Якщо термін дії доменів закінчується ≤ 1 рік → Фішинг

У іншому випадку → Легітимна URL-адреса.

4.2) Вік домену (Age of Domain).

Дане правило виявлення фішингу зосереджено на віці домену, отриманого з бази даних WHOIS [12]. Оскільки фішингові вебсайти часто існують недовго, вік домену є важливим фактором при оцінці легітимності вебсайту. Зловмисники, як правило, використовують нещодавно зареєстровані доменні імена, які знищуються із закінченням фішингової кампанії, тоді як легальні вебсайти, як правило, підтримують свої домени протягом тривалого часу.

База даних WHOIS зберігає важливу інформацію про власника домену, дату реєстрації та закінчення терміну дії, що як раз і дозволяє визначити вік домену. Проаналізувавши це, було виявлено, що мінімальний вік легальних доменів становить щонайменше 6 місяців. Це робить вік домену цінною характеристикою для виявлення фішингових сайтів, оскільки новіші домени з більшою ймовірністю можуть бути пов'язані з шахрайською діяльністю.

Правило:

Якщо вік домену ≥ 6 місяців → Легітимна URL-адреса

У іншому випадку → Фішинг.

4.3) Записи системи доменних імен (DNS Record).

Система доменних імен (DNS) є важливою частиною інфраструктури Інтернету, відповідальною за переклад доменних імен в IP-адреси, які комп’ютери використовують для обміну даними. Записи DNS містять важливу інформацію про домен, зокрема його право власності та пов’язані з ним служби. DNS-записи для легальних вебсторінок зазвичай добре створені та підтримуються, оскільки вони забезпечують належне функціонування сайту.

З іншого боку, фішингові вебсайти, як правило, часто використовують доменні імена, які або не мають дійсних записів DNS, або пов’язані з тимчасовими чи шкідливими службами. Ці сайти можуть використовувати фальшиві ідентифікаційні дані або підробку відомих організацій, хоча заявлену особу неможливо перевірити через базу даних WHOIS, яка зберігає

інформацію про власників доменів. То ж це вагомий показник того, що вебсайт може бути шахрайським.

Правило:

Якщо немає DNS-запису для домену → Фішинг

У іншому випадку → Легітимна URL-адреса.

4.4) Відвідуваність сайту (Website Traffic).

Популярність вебсайту, виміряна такими інструментами, як Alexa, дає цінну інформацію про його легітимність. Alexa – це відома вебінформаційна компанія, яка відстежує трафік вебсайту, включаючи кількість відвідувачів, кількість сторінок, які ті відвідують, та інші ключові показники, які відображають популярність і надійність вебсайту [3]. Сайти з високим трафіком, як правило, заслуговують на більший рівень довіри, оскільки вони, швидше за все, широко визнані, на них посилаються та їх відвідує більше людей.

З іншого боку, фішингові сайти, як правило, існують дуже недовго і не накопичують значного трафіку, а тому не розпізнаються базою даних Alexa. У дослідженні було помічено, що законні вебсайти, навіть у найгіршому випадку, входять до топ-100 000 на Alexa. Це означає, що легальні вебсайти, навіть якщо вони не є одними з найпопулярніших, все одно мають базовий рівень трафіку, який можна виміряти за допомогою Alexa.

На противагу цьому, фішингові сайти або взагалі не займають місця в рейтингу, або демонструють надзвичайно низький трафік. На основі цього спостереження було сформулюване наступне правило для класифікації вебсайтів на основі їх відвідуваності.

Правило:

Якщо рейтинг сайту <100,000 → Легітимна URL-адреса

Якщо рейтинг сайту>100,000 → Підозріла URL-адреса

У іншому випадку → Фішинг.

4.5) Ранжування вебсторінок (PageRank).

PageRank – це алгоритм, розроблений компанією Google для оцінки важливості та релевантності вебсторінок в Інтернеті. Він працює шляхом аналізу кількості та якості посилань, що вказують на вебсторінку з припущенням, що більш цінні або надійні сайти, як правило, мають більше високоякісних посилань, які вказують на них. Значення PageRank варіюється від 0 до 1, причому вищі значення вказують на більшу важливість та вплив в Інтернеті.

Фішингові сайти часто мають дуже низьке значення PageRank або взагалі не мають його через свою ефемерну природу і відсутність легальних вхідних посилань. У наборі даних було виявлено, що приблизно 95% фішингових вебсторінок не мають PageRank, що означає, що вони не накопичили достатньо надійних посилань, щоб їх розпізнав алгоритм. Решта 5% фішингових вебсайтів можуть мати значення PageRank, але воно рідко перевищує 0,2, що вказує на мінімальну важливість або довіру в Інтернеті.

Легітимні вебсайти, навпаки, мають тенденцію накопичувати вищий PageRank з часом, оскільки на них посилаються інші авторитетні сайти, вони широко розповсюджуються та набувають надійної присутності в Інтернеті. Це робить PageRank корисною функцією для розрізнення легальних і фішингових вебсайтів.

Правило:

Якщо значення PageRank < 0.2 → Фішинг

У іншому випадку → Легітимна URL-адреса.

4.6) Індексування Google (Google Index).

Система індексації Google призначена для сканування, аналізу та каталогізації вебсторінок, щоб вони з'являлися в результатах пошуку. Це означає, що вебсайт визнається і приймається як достатньо релевантний для включення в результати пошуку Google, що є важливим показником його достовірності та видимості в Інтернеті.

Фішингові сайти зазвичай не індексуються Google через їх короткий термін існування. Зазвичай вони створюються для тимчасового використання, і до того часу, як пошукові роботи Google їх індексують, багато фішингових сайтів уже було видалено або заблоковано.

Правило:

Якщо вебсторінка проіндексована Google → Легітимна URL-адреса

У іншому випадку → Фішинг.

4.7) Кількість посилань, що вказують на сторінку (Number of Links Pointing to Page).

Кількість зовнішніх посилань, що вказують на вебсторінку, є ще одним важливим показником її легітимності. Цей показник ґрунтується на припущенні, що законні вебсайти мають тенденцію накопичувати зовнішні посилання з інших сайтів, оскільки пошукові системи посилаються на них, обмінюються ними, індексують їх. А так як фішингові сайти не існують довго, то і не мають рівня довіри, щоб залучити посилання із зовнішніх джерел.

Іншими словами, вебсайти з великою кількістю зовнішніх посилань, як правило, мають вищий рівень легітимності, оскільки вони з більшою ймовірністю інтегровані в ширшу вебекосистему. Таким чином, аналіз набору даних про фішинг показує, що 98% фішингових вебсайтів не мають зовнішніх посилань, що свідчить про їхню ефемерну природу.

З іншого боку, легальні сайти зазвичай мають щонайменше два зовнішні посилання. Ці посилання є сигналом довіри і видимості, оскільки вони вказують на те, що інші сайти знають про контент і послуги, які пропонує сайт.

Правило:

Якщо посилань, що вказують на вебсторінку = 0 → Фішинг

Якщо посилань, що вказують на вебсторінку > 1 і ≤ 2 → Підозріла URL-адреса

У іншому випадку → Легітимна URL-адреса.

4.8) Функція на основі статистичних звітів (Statistical-Reports Based Feature).

Різні організації, такі як PhishTank і StopBadware, відіграють важливу роль у моніторингу фішингових вебсайтів і складанні звітів про них. Ці організації надають регулярні статистичні звіти, щоб допомогти визначити тенденції та закономірності фішингової діяльності. PhishTank, наприклад, публікує щомісячні та щоквартальні звіти, в яких висвітлюються ключові дані, зокрема "10 найпопулярніших доменів" і "10 найпопулярніших IP-адрес", залучених до фішингових атак [19]. Подібним чином StopBadware зосереджується на виявленні зловмисних хостів шляхом складання списків, такі як "50 найпопулярніших IP-адрес, пов’язаних із шкідливою чи фішинговою діяльністю".

Правило:

Якщо хост належить до найпопулярніших фішингових IP-адрес або найпопулярніших фішингових доменів → Фішинг

У іншому випадку → Легітимна URL-адреса.

4.9) Використання нестандартного порту (Using Non-Standard Port).

Контроль відкритих портів має вирішальне значення для підтримки безпеки сервера та мінімізації ризику несанкціонованого доступу. Порти - це точки входу, які дозволяють службам або додаткам взаємодіяти через мережу, і кожен порт зазвичай асоціюється з певним протоколом або службою, як показано в табл. 1.1. Порти ідентифікуються двобайтовими числами, які починаються з 1 до 65535, і призначаються операційною системою для представлення процесів прикладного рівня. Номер порту описує конкретний протокол, який слід використовувати під час встановлення з’єднання з певною мережевою службою. Ці порти є віртуальними точками контакту, які полегшують зв'язок між комп’ютерами.

Для захисту від потенційних вторгнень важливо дотримуватися принципу «найменших привілеїв», що означає, що тільки необхідні порти повинні бути відкритими, а всі інші залишатися закритими або обмеженими.

Така практика обмежує площу поверхні, доступну для використання зловмисниками.

Брандмауери, проксі-сервери та системи трансляції мережевих адрес (NAT) відіграють ключову роль в управлінні доступом до портів. За замовчуванням багато систем безпеки блокують всі порти і відкривають лише ті, які явно необхідні для роботи основних служб. Такий контрольований доступ значно зменшує ймовірність запуску несанкціонованих сервісів на сервері. Якби всі порти були відкриті, зловмисники могли б легко скористатися вразливостями, запустивши шкідливі служби, скомпрометувавши дані користувачів або отримавши несанкціонований контроль над системою.

Таблиця 1.1

Загальні порти для перевірки

ПОРТ	Сервіс	Значення	Бажаний статус
21	FTP	Передача файлів з одного хоста на інший	Close
22	SSH	Безпечний протокол передачі файлів	Close
23	Telnet	Забезпечує двонаправлений інтерактивний текстовий зв'язок	Close
80	HTTP	Гіпертестовий протокол передачі даних	Open
443	HTTPS	Захищений протокол передачі гіпертексту	Open
445	SMB	Надання спільного доступу до файлів, принтерів	Close
1521	ORACLE	Доступ до бази даних oracle з Інтернету.	Close
3306	MySQL	Доступ до бази даних MySQL з Інтернету.	Close
3389	Remote Desktop	Забезпечення віддаленого доступу та віддаленої співпраці	Close

Як показано в табл. 1.1, критичні сервіси, такі як HTTP (порт 80) і HTTPS (порт 443), зазвичай є єдиними портами, які повинні залишатися відкритими на вебсервері, оскільки вони обробляють стандартний вебтрафік. Однак інші порти, пов’язані з передачею файлів (наприклад, FTP на порту 21), доступом до віддаленого робочого столу (наприклад, Remote Desktop на порту 3389) або доступом до баз даних (наприклад, MySQL на порту 3306), повинні бути закриті, якщо це не є абсолютно необхідним. Якщо залишити ці порти відкритими без потреби, зловмисники можуть отримати можливість скомпрометувати конфіденційну інформацію або запустити подальші атаки.

Як наслідок, закриття неважливих портів і надання довірених служб лише через певні порти є ефективним методом підвищення безпеки сервера. Постійне спостереження за відкритими портами та відключення непотрібних служб також вважається розумним підходом до підтримки безпечного середовища.

Правило:

Якщо номер використовуваного порту перемкнений з бажаного статусу → Фішинг

У іншому випадку → Легітимна URL-адреса.

Узагальнення евристичних правил, які були згруповані за окремими категоріями, дозволило створити класифікацію у вигляді ментальної мапи (рис.1.1), що є одним із ключових результатів даної роботи. Ментальна мапа, як інструмент візуалізації, забезпечує структурований і логічно впорядкований підхід до аналізу методів виявлення фішингових вебсайтів. Вона наочно демонструє взаємозв’язки між різними правилами, спрямованими на ідентифікацію підозрілої поведінки вебресурсів, а також дозволяє чітко простежити класифікацію характеристик, які притаманні фішинговим атакам.

Новизна запропонованого підходу полягає в інтеграції та вдосконаленні матеріалів, отриманих із різних джерел інформації, що дозволило систематизувати наявні методи детектування фішингу та подати їх у більш зручній для аналізу формі. Завдяки ментальній мапі спеціалісти з

інформаційної безпеки отримують зручний інструмент для практичного використання у виявленні фішингових вебсайтів.

Рисунок 1.1 – Ментальна мапа класифікації евристичних правил детектування фішингових сайтів

Отже, розроблена класифікація у вигляді ментальної мапи є результатом ретельного аналізу та вдосконалення зібраного матеріалу, що не лише розширює наукову базу, але є початком для розробки практичного інструменту для боротьби з сучасними кіберзагрозами. Цей підхід може слугувати основою для подальших досліджень у сфері виявлення фішингу, а також сприяти розробці нових методів захисту від атак, заснованих на соціальній інженерії.

1.3. Продуктивність правил та ймовірність їх спрацювання

Тестові дані використовуються для вимірювання продуктивності моделі бінарної класифікації, тоді як навчена модель використовується для прогнозування класу URL-адрес у тестових даних. У результаті цих оцінок кожна URL-адреса поділяється на чотири категорії, а саме: True Positive, True Negative, False Positive, False Negative [1]. Кожну правильно класифіковану фішингову URL-адресу позначають як True Positive. Правильно класифіковані легітимні URL-адреси позначаються як True Negative. У випадку, коли легальну URL-адресу було визначено неправильно, ту відносять до категорії False Positive. І, нарешті, False Negative – це фішингова URL-адреса, яку було визначено невірно.

Ці чотири класи утворюють матрицю помилок. Рівняння матриці помилок 1.1–1.5 показують, як обчислюються ці метрики.

Достовірність (Accuracy)"– це основний показник оцінки, який вимірює відсоток правильних прогнозів, зроблених моделлю. Він обчислюється шляхом ділення кількості правильних прогнозів на загальну кількість прогнозів, зроблених моделлю:

$$Accuracy = \frac{TruePositive + TrueNegative}{TP + TN + FP + FN}. \quad (1.1)$$

Однак достовірність іноді може вводити в оману, якщо набір даних незбалансований, тобто один клас має значно більше зразків, ніж інший. У таких випадках модель може частіше передбачати клас більшості, що призводить до високої точності, але поганої продуктивності для класу меншості.

Істинно позитивний рівень (TPR) або частка всіх фактичних позитивних результатів, які були правильно класифіковані як позитивні, також відомий як відгук:

$$TruePositiveRate(Recall) = \frac{TruePositive}{TruePositive + FalseNegative}. \quad (1.2)$$

У прикладі з класифікацією фішингових URL-адрес, відгук вимірює частку фішингових посилань, які були класифіковані як фішинг. Гіпотетична ідеальна модель матиме нуль хибнонегативних результатів і, отже, показник відкликання (TPR) дорівнюватиме 1,0, тобто 100% рівень виявлення. У незбалансованому наборі даних, де кількість фактичних позитивних результатів дуже і дуже мало, скажімо, всього 1–2 приклади, відгук менш значущий і менш корисний як показник.

Тобто це показник, який вимірює, як часто модель машинного навчання правильно визначає справжні позитивні результати з усіх фактичних позитивних зразків у наборі даних.

Рівень хибних спрацьовувань (FPR) – це частка всіх фактичних негативних результатів, які були помилково класифіковані як позитивні, також відомі як вірогідність помилкової тривоги:

$$FalsePositiveRank = \frac{FalsePositive}{FalsePositive + TrueNegative}. \quad (1.3)$$

Якщо брати приклад з класифікацією фішингових посилань, то FPR вимірює частку законних легітимних посилань, які були помилково класифіковані як фішинг. Ідеальна модель повинна мати нуль помилкових спрацьовувань і, отже, FPR 0,0 тобто рівень помилкових тривог 0%. У незбалансованому наборі даних, де кількість фактичних негативів дуже і дуже мало, скажімо, всього 1-2 приклади, FPR менш значущий і менш корисний як показник.

Точність (Precision) – це частка всіх позитивних класифікацій моделі, які насправді є позитивними:

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}. \quad (1.4)$$

Точність вимірює частку URL-адрес, класифікованих як фішинг, які були фішингом. Точність підвищується в міру зменшення хибнопозитивних результатів, а повнота покращується в міру зменшення хибно-негативних результатів. Але збільшення порога класифікації має тенденцію зменшувати кількість хибнопозитивних результатів і збільшувати кількість хибнонегативних результатів, тоді як зменшення порога має протилежний ефект. В результаті точність та відгук часто демонструють зворотну залежність, коли поліпшення одного з них погіршує інше.

Тобто, можна сказати, що точність - це основний показник оцінки, який вимірює відсоток правильних прогнозів, зроблених моделлю.

Оцінка F-1 – це середньозважена оцінка точності та відгуку, де ваги однакові. Він використовується для збалансування компромісу між точністю та пригадуванням.

Оцінка F-1 розраховується наступним чином:

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}. \quad (1.5)$$

Наприклад, якщо модель має високу точність, але низький коефіцієнт пригадування, це означає, що вона робить менше хибних спрацьовувань, але пропускає багато вірних спрацьовувань. І навпаки, модель з високим рівнем пригадування, але низькою точністю робить більше хибних спрацьовувань, але фіксує більше істинних спрацьовувань. У таких випадках оцінка F-1 може допомогти нам визначити, яка модель є кращою.

Висновки за розділом 1

Підбиваючи підсумки першого розділу, розвиток технологій розширює можливості зловмисників, внаслідок чого їм доводиться використовувати різні форми фішингу за допомогою яких вони можуть отримати доступ до конфіденційних даних жертв. Захистом від таких атак є багатогранні підходи, починаючи від технічних засобів і закінчуючи обізнаністю користувачів.

У першому підрозділі було детально описано різні форми фішингу, які прикладами поєднання психологічних тактик та технологічних засобів на людину, а саме через оманливі електронні листи, шахрайські телефонні дзвінки або шкідливі текстові повідомлення фішингові атаки експлуатують довіру та терміновість, щоб змусити жертв розкрити конфіденційну інформацію або поставити під загрозу їхню власну безпеку.

У другому підрозділі було розглянуто класифікацію евристичних правил виявлення фішингових вебсайтів, заснованих на аналізі URL-адрес та ненормальних функцій. Описані в підрозділі правила дозволяють ідентифікувати потенційно небезпечні сайти за такими ознаками, як використання IP-адрес, довгі або скорочені URL, наявність символів «@» та «//», додавання суфіксів або префіксів, кількість субдоменів, некоректне використання маркерів HTTPS та інших ознак, що вказані в класифікації у вигляді ментальної мапи, де описані всі евристичні правила детектування, зібрані з різних джерел. Дана мапа є науковою новизною роботи.

У третьому підрозділі, маючи наглядну класифікацію евристичних правил детектування, описано, як можна виміряти за допомогою ключових метрик ефективності алгоритму. До основних показників оцінки відносяться достовірність (Accuracy), істинно позитивний рівень (Recall), рівень хибних спрацьовувань (False Positive Rate), точність (Precision) та F1-оцінка. Аналіз цих метрик є важливим етапом перевірки роботи системи виявлення загроз і забезпечення її максимальної точності при мінімізації хибних спрацьовувань.

РОЗДІЛ 2
МОДЕЛЬ ВИЗНАЧЕННЯ ЙМОВІРНОСТІ ПРАВИЛЬНОГО ВИЗНАЧЕННЯ ФІШИНГОВИХ URL НА ОСНОВІ ПРАВИЛ

2.1. Структурно-логічна схема моделі визначення ймовірності фішингових вебсайтів

У сьогоднішньому контексті швидкого розвитку цифрових технологій та зростання кіберзагроз, зокрема фішингу, існує термінова необхідність у розробці ефективних способів визнання шкідливих вебресурсів. У науковій літературі та практиці безпеки інформаційних систем існує багато підходів для оцінки ймовірності. Ці підходи включають статистичні методи, баєсівські моделі та методи, засновані на алгоритмах машинного навчання [17].

Кожен з цих підходів має свої характеристики, що визначають доцільність використання в певних умовах. Метод Байєса дозволяє використовувати апостеріорні ймовірності. Це дозволяє адаптувати наявну інформацію про ознаки вебресурсів та їхню комбінацію. Методи машинного навчання, з іншого боку, забезпечують високу точність шляхом визначення складних моделей за рахунок здатності виявляти складні закономірності у великих обсягах структурованих та неструктурованих даних. Однак їх впровадження часто вимагає значних обчислювальних ресурсів та складних етапів модельного навчання.

Попри зазначене, статистичні методи не втрачають своєї актуальності. Вони характеризуються відносною простотою впровадження, високою інтерпретацією результатів та можливістю оперативної оцінки ефективності окремих ознак або правил, що застосовуються для виявлення фішингових загроз. Зважаючи на це, статистичні підходи для визначення ймовірності фішингу залишаються придатними для використання в системах попереднього аналізу або як інтегральний компонент комбінованих моделей [9].

Вибір конкретного методу визначення ймовірності фішингових вебсайтів у рамках даного дослідження зумовлений кількома ключовими факторами. По-перше, статистичний підхід забезпечує прозорість та простоту інтерпретації отриманих результатів, що є важливим аспектом при розробці систем автоматичного виявлення загроз. Використання чітко визначених евристичних правил дозволяє не лише якісно оцінити кожен із параметрів URL-адреси, але й обґрунтувати прийняття рішень на основі кількісних характеристик.

По-друге, статистичні методи є менш ресурсомісткими у порівнянні з алгоритмами машинного навчання, що дозволяє реалізувати модель на базі доступних обчислювальних потужностей і зберегти високу швидкість обробки даних. Це має особливе значення для систем, які мають працювати в реальному часі або на обмежених за потужністю платформах.

По-третє, враховуючи характер дослідження, орієнтованого на оцінку ефективності набору евристичних правил, статистичний аналіз надає змогу однозначно визначити значущість окремих ознак та їхній внесок у загальний рівень виявлення фішингових URL. Таким чином, застосування статистичного методу дозволяє виконати як якісний, так і кількісний аналіз, що є основою для подальшої оптимізації моделі.

З огляду на зазначене, у даній роботі прийнято за основу статистичний підхід до моделювання ймовірності правильного визначення фішингових вебсайтів, що дає змогу поєднати простоту реалізації, інтерпретованість результатів та ефективність виявлення.

З метою формалізації процесу виявлення фішингових вебресурсів на основі статистичного підходу розроблено структурно-логічну схему моделі визначення ймовірності фішингу (рис.2.1). Схема відображає послідовність основних етапів обробки вхідних даних, формування ознак, їх оцінювання за статистичними критеріями та прийняття рішення щодо ймовірної фішингової активності. Кожен елемент схеми логічно поєднано із наступними діями, що

забезпечує цілісність процесу моделювання та уможливлює подальше впровадження розробленої моделі в автоматизовані системи кіберзахисту.

Рисунок 2.1 – Структурно-логічна схема моделі

2.2. Реалізація моделі виявлення фішингових URL-адрес

У рамках даного дослідження було використано два набори даних, що містять URL-адреси: PhishingDataset.csv (рис.2.2), який складається з фішингових посилань, та LegitimateDataset.csv (рис.2.3), що містить легітимні URL-адреси. З обох датасетів були випадковим чином відібрані по 40000 посилань з більшого набору даних для забезпечення збалансованої вибірки і коректного аналізу.

Аналіз URL-адрес здійснювався за допомогою власнорозробленої програми на мові програмування Python, яка реалізує низку евристичних правил для оцінки рівня довіри до вебресурсів. Зокрема, це дослідження базується на методології "функції на основі адресного рядка" (Address Bar based Features), що передбачає виявлення характерних ознак фішингових сайтів шляхом аналізу структури URL.

Програма (додаток А) зчитує набори даних, перевіряє кожне посилання на відповідність заданим правилам і підраховує, скільки з них відповідають певним критеріям. Для оцінки достовірності було використано вісім наборів функцій (рис.2.4), які поступово розширюють можливості перевірки даних. Спочатку застосовується перевірка на наявність у вебпосиланні IP-адреси в десятковому або шістнадцятковому вигляді. Надалі з кожною функцією додається по одному евристичному правилу: перевірка наявності «@», довжини URL-адреси, наявності перенаправлень, маркера «HTTPS/HTTP» у доменній частині, використання служб скорочення URL-адрес, наявності «-» та використання субдоменів.

То ж, аналіз полягав у визначенні того, скільки URL-адрес з кожного набору даних відповідають певним критеріям. Зібрані результати (рис.2.5) дають змогу оцінити ефективність різних евристичних правил для автоматичного виявлення фішингових сайтів.

Рисунок 2.2 – Датасет фішингових URL-адрес

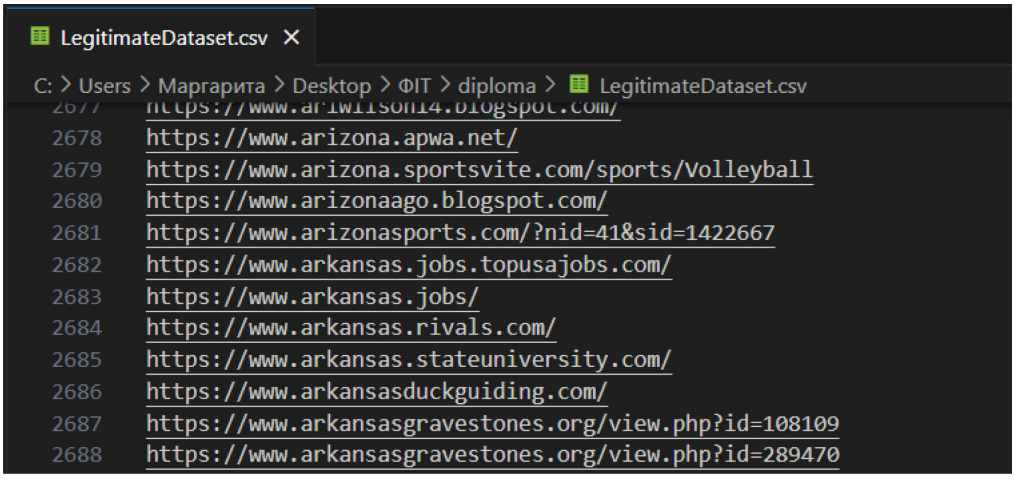

Рисунок 2.3 – Датасет легітимних URL-адрес

```python
#Список наборів функцій для аналізу
function_sets = [
    [havingIP],
    [havingIP, haveAtSign],
    [havingIP, haveAtSign, getLength],
    [havingIP, haveAtSign, getLength, redirection],
    [havingIP, haveAtSign, getLength, redirection, httpDomain],
    [havingIP, haveAtSign, getLength, redirection, httpDomain, tinyURL],
    [havingIP, haveAtSign, getLength, redirection, httpDomain, tinyURL, prefixSuffix],
    [havingIP, haveAtSign, getLength, redirection, httpDomain, tinyURL, prefixSuffix, countSubdomains]
]
```

Рисунок 2.4 – Набори функцій для аналізу

```
PS C:\Users\Маргарита> & C:/Users/маргарита/AppData/Local/Programs/Python/Python312/python.exe
Function Set 1: Phishing = 340, Legitimate = 0
Function Set 2: Phishing = 830, Legitimate = 3
Function Set 3: Phishing = 17045, Legitimate = 1258
Function Set 4: Phishing = 17076, Legitimate = 1258
Function Set 5: Phishing = 17086, Legitimate = 1258
Function Set 6: Phishing = 18573, Legitimate = 4772
Function Set 7: Phishing = 25768, Legitimate = 6762
Function Set 8: Phishing = 31481, Legitimate = 7751
```

Рисунок 2.5 – Результат виконання програми

2.3. Аналіз ймовірностей спрацювання евристичних правил

Було проведено оцінку ймовірностей спрацювання кожного окремого правила, реалізованого як функції в розробленому програмному коді. Для кожного набору функцій було обчислено ймовірність спрацювання (табл.2.1) та ймовірність їх появи (табл.2.2) на фішингових ресурсах. Зібрані статистичні дані дали змогу проаналізувати кожне правило та визначити його внесок у загальну модель.

Таблиця 2.1

Ймовірності спрацювання евристичних правил

Правило	Ймовірність спрацювання
Використання IP-адреси	1
URL-адреса містить символ «@»	0,9916
Довга URL-адреса для приховування підозрілої частини	0,9776
Переспрямування за допомогою “//”	0,9686
Наявність маркера «HTTPS» у доменній частині URL-адреси	0,9685
Використання служб скорочення URL-адрес	0,8849
Додавання до домену префікса або суфікса, розділених «-»	0,8567
Субдомен і кілька субдоменів	0,8304

Отримані результати свідчать про високу ефективність використаних в програмному коді евристичних правил до виявлення фішингових вебсайтів. Найбільш ймовірними ознаками фішингової поведінки стали використання IP-адрес замість доменних імен, наявність символу «@» в URL-адресі, надмірна

довжина URL-адреси, використання механізму перенаправлення за допомогою подвійної косої риски та наявність «https» або «http» у доменній частині. Ці ознаки вказують на високий рівень виявлення, що підтверджується відповідною ймовірністю виявлення, близькою до 1. Менш поширені, але не менш значущі ознаки, такі як використання сервісів скорочення URL-адрес, наявність префіксів або суфіксів в доменних іменах і використання численних субдоменів, також є потенційно шкідливими ресурсами і є важливими факторами для їх виявлення. Ці ознаки були зафіксовані з ймовірністю понад 0.8, що свідчить про їхню потенційну корисність у процесі виявлення загроз.

Для оцінки ефективності також має бути визначена ймовірність виявлення кожного правила на обраних з датасету URL-адресах. Ці ймовірності показують, наскільки часто певна ознака зустрічається серед них. Нижче наведена таблиця, що містить результати цього аналізу.

Таблиця 2.2

Розподіл виявлення евристичних правил

Правило	Ймовірність виявлення
Використання IP-адреси	0,0085
URL-адреса містить символ «@»	0,01225
Довга URL-адреса для приховування підозрілої частини	0,4054
Переспрямування за допомогою “//”	0,000775
Наявність маркера «HTTPS» у доменній частині URL-адреси	0,00025
Використання служб скорочення URL-адрес	0,0372
Додавання до домену префікса або суфікса, розділених «-»	0,1799
Субдомен і кілька субдоменів	0,1428

Табл. 2.2 демонструє розподіл ймовірностей виявлення різних евристичних правил. Аналіз показує, що деякі ознаки зустрічаються значно частіше, наприклад, маніпуляції з довжиною URL-адреси, використання субдоменів або додавання префіксів і суфіксів до доменного імені. Інші характеристики, зустрічаються рідше, але все ще можуть свідчити про потенційну загрозу.

Загалом, отримані дані підтверджують ефективність застосування комплексу евристичних правил для виявлення фішингових сайтів, оскільки навіть малопоширені ознаки в сукупності допомагають ідентифікувати шкідливі ресурси. В сумі ймовірності всіх правил дорівнюють 0.787, що і є істинно позитивним рівнем спрацювання усіх правил в сукупності.

Таким чином, отримані результати з табл. 2.1 і 2.2 підтверджують ефективність застосування евристичних правил для попереднього аналізу фішингових вебресурсів. Ці можливості можуть бути використані для виявлення загрозливих URL-адрес з високою точністю, що важливо для автоматизованих систем кібербезпеки.

Висновки за розділом 2

У другому розділі було проведено комплексний аналіз ефективності евристичних правил для виявлення фішингових сайтів на основі структури URL-адрес. Дане дослідження проводилось на збалансованій вибірці, що включала рівну кількість фішингових та легітимних URL-адрес.

У першому підрозділі була запропонована модель, яка використовує чітко визначені евристичні правила та кількісні критерії оцінки ймовірності фішингової активності у вебресурсах. Структурно-логічна схема формалізує описаний процес, ілюструючи послідовність від введення даних до статистичної оцінки та оцінки ефективності моделі. Ця структура закладає основу для подальшої оптимізації та інтеграції в автоматизовані рішення кібербезпеки.

У другому підрозділі була продемонстрована програмна реалізація моделі з використанням двох датасетів з фішинговими та легітимними URL-адресами. Було розроблено власну програму на Python, яка застосовує структурований набір евристичних правил, зосереджених на методології "Особливості на основі адресного рядка", та було зібрано необхідні для подальшого аналізу результати кожного правила.

У третьому підрозділі було розраховано ймовірності виявлення та спрацьовування евристичних правил та було виявлено, що ймовірність виявлення найбільша у правилах: довга URL-адреса, додавання до домену префікса або суфікса, розділених «-», субдомен і кілька субдоменів, а в той же час ймовірність спрацювання є особливо високою в таких ознаках, як використання IP-адреси замість доменного імені, наявність символу «@» у URL-адресі, надмірна довжина адреси, наявність «https» або «http» у доменній частині та застосування переспрямувань. Отримані результати дозволяють оцінити ступінь спрацювання цих ознак серед фішингових ресурсів і надалі будуть необхідні для розрахунку ефективності.

РОЗДІЛ 3
ОЦІНКА ЕФЕКТИВНОСТІ ВИЗНАЧЕНИХ ФІШИНГОВИХ URL НА ОСНОВІ ПРАВИЛ

3.1. Ефективність спрацювання правил

Після попереднього етапу аналізу, який базується на спрацюванні евристичних правил класифікації ознак на основі адресного рядка для виявлення фішингових сайтів, необхідно оцінити ефективність розробленої моделі. Для цього використовуються ключові метрики, які визначають якість алгоритму та його здатність розрізняти фішингові та легітимні сайти.

Згідно з підрозділом 1.3 було визначено наступні основні показники: достовірність, істинно позитивний рівень, рівень хибних спрацьовувань, точність, оцінка F-1. Для розрахунку буде використано формули 1.1 – 1.5.

Достовірність (Accuracy)" показує, наскільки правильно класифіковані раніше евристичні правила визначили усі перевірені URL-адреси.

$$Accuracy = \frac{31481 + 32249}{31481 + 32249 + 7751 + 8519} = 0{,}7966. \quad (2.1)$$

Істинно позитивний рівень (TPR), також відомий як Recall, дозволяє оцінити, якою мірою метод виявляє фішингові сайти, не залишаючи їх без уваги. Високий TPR означає, що алгоритм ефективно виявляє фішингові загрози; низький TPR означає, що багато фішингових сайтів залишаються нерозпізнаними, що є серйозною проблемою для безпеки користувачів.

$$TruePositiveRate(Recall) = \frac{31481}{31481 + 8519} = 0{,}787. \quad (2.2)$$

Рівень хибних спрацьовувань відображає частку легітимних сайтів, помилково класифікованих як фішингові. Якщо FPR алгоритму занадто високий, це означає, що його прогнози надто агресивні і можуть негативно вплинути на користувачів, оскільки вони можуть втратити доступ до легальних ресурсів. У таких випадках необхідно оптимізувати евристичні правила, щоб знизити FPR без суттєвого зменшення TPR.

$$FalsePositiveRank = \frac{7751}{7751 + 32249} = 0{,}1938. \quad (2.3)$$

Точність показує, яка частка всіх вебсайтів, які алгоритм ідентифікує як фішингові, насправді є фішинговими. Висока точність означає, що алгоритм рідко помиляється у виявленні загроз і має мало хибних спрацьовувань.

$$Precision = \frac{31481}{31481 + 7751} = 0{,}8024. \quad (2.4)$$

Так як оцінка F-1 є середнім гармонійним значенням точності і істинно позитивним рівнем, а отже, дозволяє оцінити, наскільки добре алгоритм балансує між виявленням загроз і мінімізацією хибних спрацьовувань. Ця метрика особливо корисна, коли важливо уникнути помилкової класифікації легальних сайтів, а також виявлення фішингових сайтів; високе значення F1 означає, що алгоритм успішно вирішує обидва завдання – правильного виявлення загроз і мінімізації помилкових спрацьовувань.

$$F1 - Score = 2 \times \frac{0{,}8024 \times 0{,}787}{0{,}8024 + 0{,}787} = 0{,}7946. \quad (2.5)$$

Отримане значення F1-score = 0.7946 свідчить про достатньо високий рівень ефективності застосованих евристичних правил. Це означає, що алгоритм демонструє хорошу збалансованість між точністю та повнотою, що

особливо важливо для задач кібербезпеки, де як помилкове блокування легітимних сайтів (False Positives), так і пропуск фішингових загроз (False Negatives) можуть мати серйозні наслідки.

3.2. Порівняльний аналіз методів виявлення фішингових сайтів

Підходи до виявлення фішингових атак, окрім евристики, включають ще кілька основних методів [14]: білі списки, чорні списки, методи візуальної схожості, машинне навчання та штучний інтелект (табл.3.1).

Виявлення на основі білих списків залишається основною стратегією в багатьох системах безпеки, зосереджуючись на дозволі лише перевірених легітимних посилань [21]. Цей метод корисний у контрольованих середовищах, таких як корпоративні інтрамережі, де набір дозволених доменів обмежений і добре керований. Наприклад, він забезпечує високу точність у статичних сценаріях загроз, мінімізуючи вплив невідомих ризиків, але його жорсткість часто призводить до розчарування користувачів, коли легітимні сайти блокуються через застарілі списки.

Методи чорних списків, навпаки, покладаються на реактивні бази даних відомих фішингових посилань, що робить їх поширеною першою лінією захисту в поштових фільтрах і браузерах [25]. Їх простота дозволяє швидко розгортати, а також має переваги у виявленні зловмисників-рецидивістів. Однак до недоліків можна віднести вразливість до тактик, що швидко змінюються, таких як перетікання доменів, коли зловмисники генерують нові URL-адреси швидше, ніж оновлюються чорні списки, що призводить до більш високих показників помилкових спрацьовувань в динамічних середовищах.

Методи візуальної схожості порівнюють дизайн сайтів, ідентифікуючи підроблені сторінки за зовнішнім виглядом [11]. Ці методи ефективно виявляють візуально оманливі сайти, але, незважаючи на це, їхні обчислювальні вимоги можуть сповільнити процеси виявлення, і вони можуть

позначати нешкідливі сайти зі схожим дизайном, збільшуючи кількість помилкових спрацьовувань у різних вебсередовищах.

Підходи штучного інтелекту охоплюють низку технологій, включаючи нейронні мережі, для аналізу та прогнозування фішингових загроз на основі великих масивів даних [8]. Однак, як підкреслюється в рекомендаціях щодо соціальної інженерії на основі ШІ, існують проблеми, пов'язані з необхідністю отримання високоякісних даних і ризиком упередженості моделі, що може призвести до помилкових класифікацій.

Машинне навчання, часто інтегроване зі штучним інтелектом, зосереджується на ітеративному навчанні на основі даних для класифікації фішингових посилань [27], підвищуючи точність з часом за допомогою таких методів, як контрольоване навчання. У порівнянні з традиційними методами, ML перевершує їх у боротьбі із загрозами «нульового дня».

Таблиця 3.1

Порівняння методів виявлення фішингових сайтів

Метод	**Білий список**	**Чорний список**	**Методи візуальної схожості**	**Евристичний метод**	**Штучний інтелект та машинне навчання**
Опис	База даних відомих легітимних URL-адрес/доменів	База даних відомих фішингових URL-адрес/доменів	Аналіз візуальних елементів вебсторінок для виявлення схожості із легітимними сайтами	Система на основі правил для аналізу характеристик URL на предмет виявлення підозрілих шаблонів	Використання різних методів ШІ та/або ML, включаючи експертні системи, навчені моделі ML для класифікації URL-адрес і контенту вебсайтів

продовження таблиці 3.1

Метод	**Білий список**	**Чорний список**	**Методи візуальної схожості**	**Евристичний метод**	**Штучний інтелект та машинне навчання**
Переваги	Швидкий захист від невідомих загроз шляхом обмеження доступу до перевірених сайтів; простий у впровадженні та підтримці для статичних середовищ	Простий у розгортанні та ефективний для блокування відомих фішингових сайтів; обчислювально ефективний для блокування в режимі реального часу	Здатний виявляти складні спроби фішингу, які уникають перевірок на основі URL-адрес; ефективний проти атак з використанням візуальних обманів	Швидкий і не потребує великих наборів даних для навчання; добре підходить для виявлення поширених індикаторів фішингу	Високоадаптивний і ефективний проти загроз, що еволюціонують; може обробляти величезні обсяги даних у режимі реального часу для виявлення аномалій, зменшує кількість хибних спрацьовувань завдяки вдосконаленому розпізнаванню образів [7].

продовження таблиці 3.1

Метод	**Білий список**	**Чорний список**	**Методи візуальної схожості**	**Евристичний метод**	**Штучний інтелект та машинне навчання**
Недоліки	Не виявляє нові, еволюціонуючі фішингові сайти, додаючи в білий список, негнучкий і вимагає постійних оновлень, що призводить до потенційних проблем у використані	Високий рівень помилкових спрацьовувань на нові варіанти фішингу, які ще не внесені до списку або визначені як помилково легітимні; вимагає частих оновлень і може бути перевантажений обсягом нових загроз	Ресурсомісткий і може давати помилкові спрацьовування на легітимні сайти, які мають візуальну схожість; менш ефективний без високоякісних еталонних зображень	Може бути легко обійдений зловмисниками, які адаптують свою тактику; високий рівень хибних спрацьовувань, якщо правила не налаштовані належним чином	Потребує значних обчислювальних ресурсів, великих наборів даних для навчання. Залежить від якісних навчальних даних; може бути вразливим до атак, які маніпулюють вхідними даними

продовження таблиці 3.1

Метод	**Білий список**	**Чорний список**	**Методи візуальної схожості**	**Евристичний метод**	**Штучний інтелект та машинне навчання**
Швидкість детектування	Дуже швидко	Дуже швидко	Середньо	Швидко	Середньо
Адаптивність до нових загроз	Низька	Низька	Середня	Середня	Висока

3.3. Інтеграція додаткових підходів та оптимізація евристичної моделі

Як було визначено у попередніх розділах, евристичні моделі покладаються на заздалегідь визначені правила для виявлення підозрілих шаблонів в URL-адресах, вмісті вебсторінок або поведінці користувачів. Як було підтверджено при практичній реалізації, евристичні правила є обчислювально ефективними та ефективними для виявлення відомих шаблонів фішингу. Однак їхня залежність від статичних правил обмежує їхню здатність протидіяти атакам нульового дня, що призводить до хибнопозитивних або хибнонегативних спрацьовувань [26]. Це негативно впливає на ефективність моделі, знижуючи при цьому гнучкість та адаптивність у динамічному середовищі кіберзагроз.

Модель, що складається лише з евристичного методу, може бути значно покращена шляхом інтеграції методів штучного інтелекту та машинного навчання [6]. Така інтеграція використає переваги кожного підходу: евристика забезпечує прозорість та швидкість, тоді як ШІ та ML пропонують

масштабованість і розпізнавання невідомих для евристики образів на основі навчання моделей на основі великих даних.

Така оптимізація може передбачати застосування різних підходів. Один з них може бути ансамблевий метод оптимізації, коли кілька моделей об'єднуються для використання сильних сторін евристики, наприклад, швидких рішень на основі правил, зі здатністю ШІ або ML вчитися на основі даних і адаптуватися до нових загроз [22]. Ансамблеві методи, такі як бегінг, бустінг або стекінг використовують алгоритми ML для вдосконалення евристичних правил, підвищуючи точність і зменшуючи кількість хибних спрацьовувань шляхом зважування рішень на основі історичних даних.

Бегінг означає, що необхідно багато разів навчати ансамбль на випадкових вибірках даних і в кінцевому підсумку усереднити надані відповіді (рис.3.1). Для виявлення фішингу ансамбль бегінгу може поєднувати евристичні правила з кількома моделями машинного навчання, наприклад, Random Forest [30], для класифікації URL-адрес. Модель Random Forest, навчена за допомогою алгоритму, є колекцією дерев вибору, кожне з яких навчається незалежно і паралельно, а результат класифікації визначається загальним голосуванням. Серед переваг цього алгоритму варто відзначити його простоту, швидкість порівняно з іншими алгоритмами бегінга та бустингу та високу точність.

Наприклад, евристичне правило може позначати URL-адреси, що містять «https» у доменній частині, тоді як Random Forest аналізує додаткові функції, такі як вік домену або підозрілі дії. Ансамбль усереднює їхні результати, щоб отримати більш надійний прогноз.

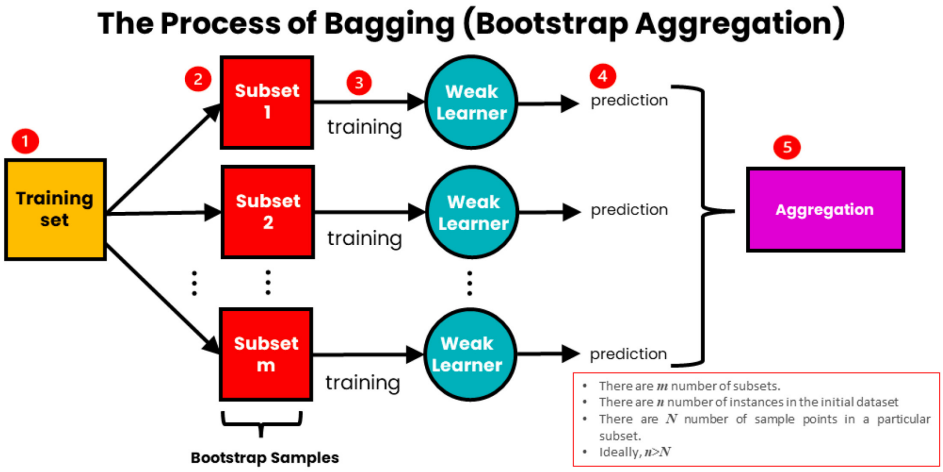

Рисунок 3.1 – Схематичний процес бегінгу [15]

Спосіб бустінгу же включає більш послідовне навчання алгоритмів. Спершу проводиться навчання першого алгоритму і відзначаються місця, де алгоритм помилився. Потім навчаємо другий, особливу увагу приділяючи місцям, на яких помилявся перший алгоритм, і продовжуємо навчання до необхідного результату (рис.3.2). То ж, на практиці евристична модель може неправильно класифікувати фішингову URL-адресу зі структурою, що виглядає легітимно. Алгоритм призначає більшу вагу цій неправильній класифікації, навчаючи наступні моделі машинного навчання зосереджуватися на подібних граничних випадках, тим самим покращуючи загальну точність. У результаті досягається найбільш якісний результат.

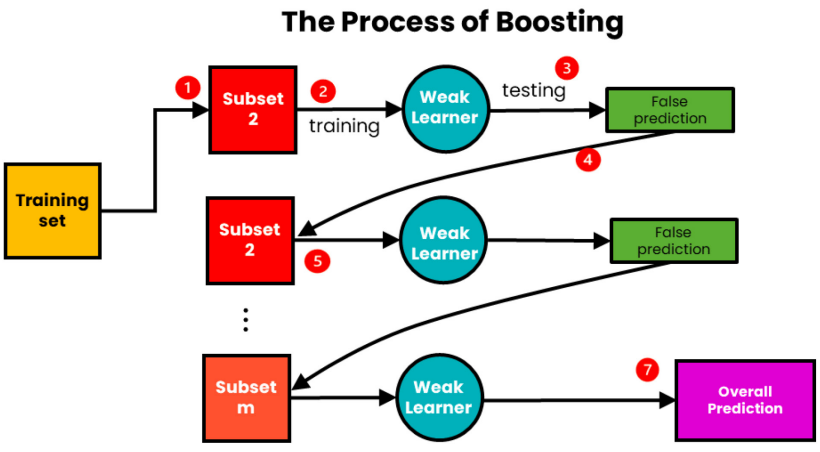

Рисунок 3.2 – Схематичний процес бустінгу [15]

Стекінг поєднує результати кількох базових моделей, наприклад, евристичних правил, модель Random Forest, за допомогою метамоделі, наприклад, логістичної регресії або нейронної мережі, для створення остаточного прогнозу (рис.3.3). Дана метамодель навчається оптимально зважувати внесок кожної базової моделі на основі їхніх сильних сторін. Наприклад, ансамбль стекування використовує евристичні правила для швидкої початкової фільтрації, Random Forest – для класифікації на основі передбачень згідно попередніх даних та модель візуальної подібності для виявлення підробки вигляду вебсторінок. Метамодель поєднує ці результати для отримання остаточного показника ймовірності фішингу.

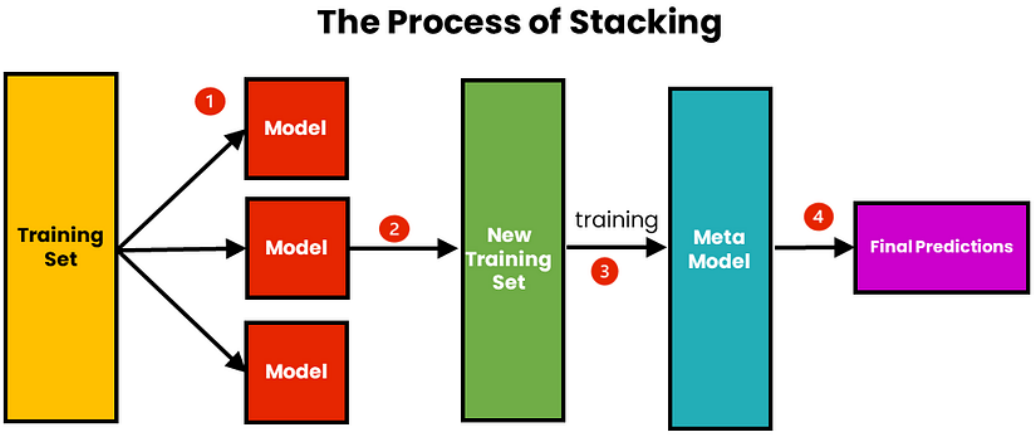

Рисунок 3.3 – Схематичний процес стекінгу [15]

Завдяки впровадженню штучного інтелекту та машинного навчання система отримує можливість навчатися на великих масивах даних, виявляти тонкі або складні закономірності та адаптуватися до нових загроз, які не можуть бути враховані статичними правилами. Моделі ШІ/ML можуть автоматично оновлювати критерії виявлення, коли зловмисники змінюють свої стратегії, забезпечуючи постійний захист як від відомих, так і від раніше небачених спроб фішингу.

Цей гібридний підхід використовує найкраще з обох світів: ефективність і зрозумілість евристичної моделі, а також адаптивність і глибину ШІ/ML. В

результаті отримана надійна, масштабована і високоточна система виявлення, яка мінімізує помилкові спрацьовування, швидко реагує на нові загрози і надає чітке обґрунтування своїх рішень, що робить її добре пристосованою до мінливого ландшафту фішингових атак.

Висновки за розділом 3

У третьому розділі було оцінено загальну ефективність запропонованого евристичного алгоритму, зроблено порівняльний аналіз різних методів виявлення фішингу, а також була запропонована інтеграція декількох методів для створення більш гнучкої для нових системи виявлення на базі евристичного методу..

У першому підрозділі була проведена оцінка точності класифікації вебресурсів на основі ключових метрик дозволила визначити загальну ефективність алгоритму. Отримане значення F1-score, що дорівнює 0.7946, свідчить про високу збалансованість між точністю та повнотою виявлення загроз. Це підтверджує, що застосовані евристичні правила дозволяють з високою ймовірністю правильно ідентифікувати фішингові сайти, мінімізуючи при цьому кількість помилкових класифікацій. Таким чином, результати дослідження демонструють ефективність запропонованого підходу до автоматичного виявлення фішингових загроз.

У другому підрозділі порівняння різних методів виявлення фішингових вебсайтів показало, як методи виявлення фішингу відрізняються за своїм підходом: традиційні методи, такі як білі та чорні списки, є простішими, але менш ефективними проти сучасних загроз, тоді як евристичні методи є одними с найшвидших у спрацюванні, а ШІ та ML пропонують розширену адаптивність ціною складності. Кожен метод є в міру ефективним для різних потреб, але гібридні підходи все частіше рекомендуються для комплексного захисту.

У третьому підрозділі було описано, що інтеграція та оптимізація евристичних моделей за допомогою ШІ та ML є потужною еволюцією в кібербезпеці, що поєднує теоретичні переваги з практичним застосуванням для створення адаптивних систем. Використовуючи ансамблеві методи, евристичні моделі можуть подолати притаманні їм обмеження, пропонуючи розширені можливості виявлення. Ці системи зможуть поєднувати швидкість та ефективність евристичних правил з адаптивними можливостями ШІ та ML, що дозволить створити надійні, масштабовані та високоефективні рішення для виявлення як відомих, так і нових фішингових загроз.

ВИСНОВКИ

У даній кваліфікаційній роботі було розроблено програму, яка автоматизує виявлення фішингових URL-адрес на основі евристичних правил, з категорії "Особливості на основі адресного рядка". Дослідження даної роботи успішно поєднує теоретичні принципи кібербезпеки з практичною алгоритмічною реалізацією для виявлення загроз.

У першому розділі було проведено комплексний аналіз сучасних фішингових атак, досліджено різновиди їх механізмів, у тому числі згідно матриці MITRE ATT&CK. Було розроблено ментальну карту евристичних правил для ідентифікації фішингових URL-адрес шляхом аналізу їх характерних ознак. Класифікація евристичних правил дозволяє ідентифікувати специфічні патерни активності, які є типовими для фішингових ресурсів, наприклад, неправильна інтеграція зовнішніх джерел чи маніпулятивна обробка введених користувачами даних. Маючи наглядну класифікацію евристичних правил детектування, значно підвищується точність ідентифікації фішингових вебсайтів, захищаючи користувачів від можливих атак. Ця точність вимірюється за допомогою ключових метрик ефективності алгоритму.

У другому розділі було практично реалізовано програмний інструмент на основі мови програмування Python для виявлення фішингових вебсайтів. За допомогою програми було зчитано набори даних, перевірено кожне посилання на відповідність заданим як функції правилам і підраховано, скільки з них відповідають певним критеріям. Також було проведено статистичний ймовірнісний аналіз кожного евристичного правила, що дозволило попередньо підтвердити ефективність застосування евристичних правил для аналізу фішингових вебресурсів.

У третьому розділі продуктивність розробленої системи виявлення була кількісно оцінена за допомогою ключових показників класифікації, таких як точність, повнота, F1-оцінка та коефіцієнт хибнопозитивних результатів. Досягнутий F1-бал, що дорівнює 0.7946, демонструє досить високу

ефективність евристичного підходу лише згідно однієї групи правил з класифікації. Крім того, порівняльний аналіз з іншими методами показав, що евристичні методи пропонують швидкий час реагування, тоді як підходи на основі штучного інтелекту/машинного навчання забезпечують чудову адаптивність до загроз, що розвиваються.

Загалом, ця робота не лише підкреслює критичну важливість боротьби з фішинговими загрозами в сучасному цифровому середовищі, але й пропонує практичний автоматизований інструмент виявлення. Поєднуючи класичні евристичні правила з систематичним аналізом, дослідження закладає основу для адаптивних систем захисту, які можна інтегрувати в існуючі інфраструктури кібербезпеки. Враховуючи еволюціонуючий характер тактики фішингу, майбутні дослідження повинні зосередитися на впровадженні машинного навчання та методів штучного інтелекту для підвищення гнучкості та стійкості системи до нових загроз.

Виходячи із поставленої мети кваліфікаційної роботи були виконані наступні завдання:

- досліджено та характеризовано сучасні методи фішингу;
- зібрано та систематизовано евристичні правила для виявлення фішингу у вигляді ментальної мапи;
- розроблено та впроваджено програмний інструмент для аналізу URL-адрес;
- оцінено ефективність виявлення фішингу за допомогою ПЗ з використанням показників продуктивності;
- визначено перспективні напрямки для покращення детектуючої системи за допомогою інтеграції штучного інтелекту та машинного навчання.

Результати кваліфікаційної роботи не лише систематизують знання про сучасні методи виявлення фішингових сайтів, але й забезпечують практичну основу для створення автоматизованих систем захисту від однієї з найнебезпечніших форм соціальної інженерії – фішингу. Впровадження таких систем дозволяє мінімізувати ризики компрометації даних, підвищуючи рівень

безпеки як для окремих користувачів, так і для організацій, що особливо актуально в умовах постійного ускладнення методів зловмисників та їх адаптації до нових технологій.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. A hybrid DNN–LSTM model for detecting phishing URLs / A. Ozcan et al. *Neural Computing and Applications*. 2021. URL: https://doi.org/10.1007/s00521-021-06401-z (дата звернення: 25.01.2025).

2. Ahmed Nafies Okasha Mohamed. A New Heuristic Based Phishing Detection Approach Utilizing Selenium Web-driver. University of Tartu. 2017. URL: https://dspace.ut.ee/server/api/core/bitstreams/69ff5ad6-f739-476e-8d39-730181be4283/content (дата звернення: 25.01.2025).

3. Amazon Alexa. Meet the new Alexa. URL: https://www.alexa.com/ (дата звернення: 25.01.2025).

4. Amir Herzberg and Ahmad Jbara. “Security and identification indicators for browsers against spoofing and phishing attacks”. In: ACM Transactions on Internet Technology (TOIT) 8.4. 2008. P. 1–36.

5. Ankesh Anand, Kshitij Gorde, Joel Ruben Antony Moniz, Noseong Park, Tanmoy Chakraborty, and Bei-Tseng Chu. Phishing URL detection with oversampling based on text generative adversarial networks. Proceedings of the IEEE International Conference on Big Data (Big Data). 2018. P. 1168–1177. doi: 10.1109/BigData.2018.8622547

6. Awasthi A., Goel N. An Approach for Efficient and Accurate Phishing Website Prediction Using Improved ML Classifier Performance for Feature Selection. *International Journal of Experimental Research and Review*. 2024. Vol. 40. P. 73–89. URL: https://doi.org/10.52756/ijerr.2024.v40spl.006 (дата звернення: 25.01.2025).

7. B. Wei, R. A. Hamad, L. Yang, X. He, H. Wang, B. Gao. A deep-learning-driven light-weight phishing detection sensor. Sensors. 2019. Vol. 19, no. 19. P. 4258. doi: https://doi.org/10.3390/s19194258

8. Basit, A., Zafar, M., Liu. A comprehensive survey of AI-enabled phishing attacks detection techniques. Telecommun Syst 76. 2021. P. 139–154. doi: https://doi.org/10.1007/s11235-020-00733-2.

9. Dou Z, Khalil I, Khreishah A, Al-Fuqaha A, Guizani M. Systematization of knowledge (sok): a systematic review of software-based web phishing detection. *IEEE Commun. Surv. Tutorials*. 2018. Vol. 19(4). P. 2797–2819. DOI: 10.1109/COMST.2017.2752087.

10. Initial Access, Tactic TA0001 - Enterprise | MITRE ATT&CK®. *MITRE ATT&CK®*. URL: https://attack.mitre.org/tactics/TA0001/ (дата звернення: 25.01.2025).

11. Jain, A.K., & Gupta, B.B. Towards detection of phishing websites on client-side using machine learning based approach. *Telecommunication Systems*. 2018. Vol. 68(4). P. 687–700.

12. Jian Mao, Pei Li, Kun Li, Tao Wei, and Zhenkai Liang. BaitAlarm: detecting phishing sites using similarity in fundamental visual features. In: Proceedings of the IEEE International Conference on Intelligent Networking and Collaborative Systems. 2013 P. 790–795.

13. Chizari Hassan, Zulkurnain Ahmad, Hamidy Ahmad, Husain Affandi. Social Engineering Attack Mitigation. *International Journal of Mathematics and Computational Science*. 2015. Vol.1, No. 4. P. 10–11.

14. Bhagwat M. D., Patil P. H. and Vishawanath T. S. A Methodical Overview on Detection, Identification and Proactive Prevention of Phishing Websites. Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV). 2021. P. 1505–1508. doi: 10.1109/ICICV50876.2021.9388441

15. Mbali K. Bagging, Boosting and Stacking: Ensemble Learning in ML Models. URL: https://www.analyticsvidhya.com/blog/2023/01/ensemble-learning-methods-bagging-boosting-and-stacking (дата звернення: 25.01.2025).

16. Mohammad, Rami & Thabtah, Fadi & Mccluskey, T. An assessment of features related to phishing websites using an automated technique. 2012. P. 492–497.

17. Olukoya D., Ogunleye G. Heterogeneous Ensemble Feature Selection and Multilevel Ensemble Approach to Machine Learning Phishing Attack

Detection. *FUOYE Journal of Engineering and Technology*. 2023. Vol. 8, no. 4. URL: https://doi.org/10.46792/fuoyejet.v8i4.1105 (дата звернення: 25.01.2025).

18. Phishing, Technique T1566 - Enterprise | MITRE ATT&CK®. *MITRE ATT&CK®*. URL: https://attack.mitre.org/techniques/T1566/ (дата звернення: 25.01.2025).

19. Phishtank. *Phishtank.* URL: https://www.phishtank.com (дата звернення: 25.01.2025).

20. Pickavance M. Best SSL certificate service of 2024. *TechRadar*. URL: https://www.techradar.com/news/best-ssl-certificate-provider (дата звернення: 25.01.2025).

21. Basnet R B., Sung A. H. and Liu Q. Rule-Based Phishing Attack Detection. Proceedings of the International Conference on Security and Management-SAM 11, Las Vegas, NV, USA, 2011.

22. Rajab M. An anti-phishing method based on feature analysis. Proceedings of the 2nd International Conference on Machine Learning and Soft Computing. ACM. 2018. P. 133–139. doi: https://doi.org/10.1145/3184066.3184082

23. Samuel Marchal, Kalle Saari, Nidhi Singh, and N Asokan. Know your phish: Novel techniques for detecting phishing sites and their targets. Proceedings of the IEEE 36th International Conference on Distributed Computing Systems (ICDCS). 2016. P. 323–333.

24. Serhii Buchyk, Dmytro Shutenko and Serhii Toliupa. Phishing Attacks Detection. Information Technology and Implementation (IT&I-2022). 2022. P. 193–200.

25. Simon Bell and Peter Komisarczuk. An analysis of phishing blacklists: Google safe browsing, openphish, and phishtank. Proceedings of the Australasian Computer Science Week Multiconference. 2020. P. 1–11.

26. Suleman M. T., Awan S. M. Optimization of URL-based phishing websites detection through genetic algorithms. *Autom. Control. Comput. Sci*. 2019. Vol. 53. No. 4. P. 333–341. doi: 10.3103/S0146411619040102.

27. Ojewumi T. O., Ogunleye G. O., Oguntunde B.O., Folorunsho O., Fashoto S. G., Ogbu N. Performance evaluation of machine learning tools for detection of phishing attacks on web pages. *Scientific African*. 2022. Volume 16, e01165. ISSN 2468-2276.

28. Alkhalil Zainab , Hewage Chaminda , Nawaf Liqaa , Khan Imtiaz. Phishing attacks: a recent comprehensive study and a new anatomy. *Frontiers in computer science*. 2021. Vol. 3. doi: https://doi.org/10.3389/fcomp.2021.563060.

29. WHOIS. Search, Domain Name, Website, and IP Tools – Who.is. *WHOIS Search, Domain Name, Website, and IP Tools – Who.is*. URL: https://who.is (дата звернення: 25.01.2025).

30. Alsariera Y. A., Adeyemo V. E., Balogun A. O. and Alazzawi A. K. AI Meta-Learners and Extra-Trees Algorithm for the Detection of Phishing Websites. *IEEE Access*. 2020. Vol. 8. P. 142532–142542.

ДОДАТКИ

ДОДАТОК А

```
import pandas as pd
from urllib.parse import urlparse
import re

#Правило "Using the IP Address"
def havingIP(url):
    match = re.search(r'(([01]?\d\d?|2[0-4]\d|25[0-5])\.'
                      r'([01]?\d\d?|2[0-4]\d|25[0-5])\.'
                      r'([01]?\d\d?|2[0-4]\d|25[0-5])\.'
                      r'([01]?\d\d?|2[0-4]\d|25[0-5])\/)|'
                      r'((0x[0-9a-fA-F]{1,2})\.'
                      r'(0x[0-9a-fA-F]{1,2})\.'
                      r'(0x[0-9a-fA-F]{1,2})\.'
                      r'(0x[0-9a-fA-F]{1,2})\/)'
                      r'(?:[a-fA-F0-9]{1,4}:){7}[a-fA-F0-9]{1,4}', url)
    if match:
        return 1
    else:
        return 0

#Правило "URL’s having “@” Symbol"
def haveAtSign(url):
    if "@" in url:
        at_sign = 1
    else:
        at_sign = 0
    return at_sign
```

```
    #Правило "Long URL to Hide the Suspicious Part"
    def getLength(url):
        if len(url) < 54:
            length = 0
        else:
            length = 1
        return length

    #Правило "Redirecting using “//”"
    def redirection(url):
        pos = url.rfind('//')
        if pos > 7:
            return 1
        else:
            return 0

    #Правило "The Existence of “HTTPS” Token in the Domain Part of the URL"
    def httpDomain(url):
        domain = urlparse(url).netloc  # Витягуємо домен
        return int(bool(re.search(r'http|https', domain))) #Перевіряємо, чи є "http" або "https" в домені

    shortening_services = r"bit\.ly|goo\.gl|shorte\.st|go2l\.ink|x\.co|ow\.ly|t\.co|tinyurl|tr\.im|is\.gd|cli\.gs|" \
                          r"yfrog\.com|migre\.me|ff\.im|tiny\.cc|url4\.eu|twit\.ac|su\.pr|twurl\.nl|snipurl\.com|" \
                          r"short\.to|BudURL\.com|ping\.fm|post\.ly|Just\.as|bkite\.com|snipr\.com|fic\.kr|loopt\.us|" \
```

```
r"doiop\.com|short\.ie|kl\.am|wp\.me|rubyurl\.com|om\.ly|to\.ly|bit\.do|t\.co|lnkd\.in|db\.tt|" \

r"qr\.ae|adf\.ly|goo\.gl|bitly\.com|cur\.lv|tinyurl\.com|ow\.ly|bit\.ly|ity\.im|q\.gs|is\.gd|" \

r"po\.st|bc\.vc|twitthis\.com|u\.to|j\.mp|buzurl\.com|cutt\.us|u\.bb|yourls\.org|x\.co|" \

r"prettylinkpro\.com|scrnch\.me|filoops\.info|vzturl\.com|qr\.net|1url\.com|tweez\.me|v\.gd|" \
                    r"tr\.im|link\.zip\.net"

    #Правило "Using URL Shortening Services TinyURL"
    def tinyURL(url):
      match=re.search(shortening_services,url)
      if match:
         return 1
      else:
         return 0

    #Правило "Adding Prefix or Suffix Separated by (-) to the Domain"
    def prefixSuffix(url):
      if '-' in urlparse(url).netloc:
         return 1
      else:
         return 0

    #Правило "Sub Domain and Multi Sub Domains"
```

```
def countSubdomains(url):
    domain = urlparse(url).netloc
    domain = domain.lstrip('www.')
    parts = domain.split('.')
    known_ccTLD = [
        'ac', 'co', 'gov', 'org', 'edu', 'net', 'com', 'info', 'mil', 'int', 'eu',
        'uk', 'de', 'fr', 'ca', 'en', 'us', 'cn', 'jp', 'kr', 'au', 'br', 'in', 'it',
        'es', 'pl', 'nl', 'se', 'ch', 'at', 'be', 'cz', 'dk', 'fi', 'gr', 'hr', 'hu',
        'ie', 'lt', 'lv', 'lu', 'mt', 'no', 'pt', 'sk', 'si', 'tr', 'ua', 'za', 'tk', 'me',
        'tv', 'ws', 'co.uk', 'org.uk', 'gov.uk', 'ac.uk', 'edu.au', 'com.au', 'gov.au'
    ]

    #Видалення ccTLD (якщо є)
    if len(parts) > 1 and parts[-1] in known_ccTLD:
        domain_part = ".".join(parts[:-1])
    else:
        domain_part = domain

    #Підрахунок кількості крапок у скоригованому домені
    dot_count = domain_part.count('.')
    if dot_count == 1:
        return 0
    elif dot_count > 1:
        return 1


#Завантаження датасетів легітимних та фішингових URL-адрес
legiurl = pd.read_csv(r"C:\Users\Маргарита\Desktop\ФІТ\diploma\LegitimateDataset.csv").sample(n=40000)
```

```
legiurl.columns = ['URLs']
phishurl = pd.read_csv(r"C:\Users\Маргарита\Desktop\ФІТ\diploma\PhishingDataset.csv").sample(n=40000)
phishurl.columns = ['URLs']

#Набори функцій
def apply_functions(url, functions):
    return any(func(url) for func in functions)

#Список наборів функцій для аналізу
function_sets = [
    [havingIP],
    [havingIP, haveAtSign],
    [havingIP, haveAtSign, getLength],
    [havingIP, haveAtSign, getLength, redirection],
    [havingIP, haveAtSign, getLength, redirection, httpDomain],
    [havingIP, haveAtSign, getLength, redirection, httpDomain, tinyURL],
    [havingIP, haveAtSign, getLength, redirection, httpDomain, tinyURL, prefixSuffix],
    [havingIP, haveAtSign, getLength, redirection, httpDomain, tinyURL, prefixSuffix, countSubdomains]
]

#Аналіз кожного набору функцій, розрахунок кількості спрацювань
results = []
for i, funcs in enumerate(function_sets, 1):
    phishing_count = sum(apply_functions(url, funcs) for url in phishurl['URLs'])
    legit_count = sum(apply_functions(url, funcs) for url in legiurl['URLs'])
```

```
    results.append((i, phishing_count, legit_count))

#Вивід результатів
for i, phish, legit in results:
    print(f"Function Set {i}: Phishing = {phish}, Legitimate = {legit}")
```

ДОДАТОК Б
КОПІЯ НАУКОВОЇ ПУБЛІКАЦІЇ

Бучик С., Толстяк М. Проблеми кібербезпеки інформаційно-комунікаційних систем: Збірник матеріалів доповідей та тез; м. Київ, 11 квітня 2025 року; Київський національний університет імені Тараса Шевченка / Редкол.: В.В. Ільченко, д.ф-м.н., проф., (голова); та ін. – К.: ВПЦ "Київський університет", 2025. С. 87 – 89.

ДЕТЕКТУВАННЯ ФІШИНГОВИХ URL НА ОСНОВІ ЕВРИСТИЧНИХ ПРАВИЛ

Маргарита Толстяк

Студентка кафедри кібербезпеки та захисту інформації, Київський національний університет імені Тараса Шевченка

Науковий керівник: Сергій Бучик

Професор кафедри кібербезпеки та захисту інформації Київського національного університету імені Тараса Шевченка, доктор технічних наук, професор

Робота присвячена дослідженню методів виявлення фішингових вебсайтів на основі евристичних правил. Фішинг є однією з найпоширеніших і найнебезпечніших форм кіберзлочинів, що спрямована на отримання доступу до конфіденційної інформації користувачів, маніпулюючи їх довірою. Методологія атак соціальної інженерії стрімко розвивається, що створює виклики для фахівців з кібербезпеки. Дослідження фокусується на евристичних підходах до аналізу веб-ресурсів, які дозволяють виявляти потенційно небезпечні вебсайти на основі структури URL-адреси, використання ненормальних функцій, підозрілих зовнішніх запитів та специфічної поведінки серверів.

Класифікація евристичних правил

Розвиток технологій розширює можливості зловмисників, внаслідок чого їм доводиться використовувати різні форми фішингу за допомогою яких вони можуть отримати доступ до конфіденційних даних жертв. Фішинг – це спосіб соціальної інженерії, коли фальшиві повідомлення використовуються для крадіжки секретних даних або встановлення зловмисного софту [1]. Зловмисники маскуються під перевірені джерела, маніпулюючи емоціями, спонукаючи до термінових дій або обіцяючи призи.

Згідно MITRE ATT&CK, у тактиці "Initial Access" техніка "Phishing" (T1566) включає чотири підтехніки: атаки через файли-вкладення, посилання, сторонні платформи та голосові зв’язки [2]. Фішингові вкладення та посилання поширюються через соціальні мережі та особисті поштові сервіси, використовуючи вкладення та посилання для обходу механізмів безпеки. Поширеними ознаками фішингу є граматичні помилки, підозрілі посилання, запити на особисту інформацію та відчуття терміновості [3]. Для кращого виявлення фішингових сайтів можна застосовувати евристичні правила для аналізу URL-адрес.

Методи евристичного аналізу дозволяють виявляти фішингові сайти на основі багатьох характеристик, таких як нетипова структура посилань, наявність підозрілих символів або неправильної послідовності слів у доменних іменах. Такі правила базуються на дослідженні шаблонів, які характеризують фішингові вебсайти, і дозволяють користувачам попереджати про потенційні загрози до того, як вони натиснуть посилання.

Для кращого розуміння правил їх можна класифікувати у декілька груп, а саме “Address Bar based Features”, “HTML and JavaScript based features”, “Domain based features”, “Abnormal based features” [4].

Узагальнення евристичних правил, які були згруповані за окремими категоріями, дозволило створити класифікацію у вигляді ментальної мапи (рис.Б.1), що є одним із ключових результатів даної роботи. Ментальна мапа, як інструмент візуалізації, забезпечує структурований і логічно впорядкований

підхід до аналізу методів виявлення фішингових вебсайтів. Вона наочно демонструє взаємозв'язки між різними правилами, спрямованими на ідентифікацію підозрілої поведінки веб-ресурсів, а також дозволяє чітко простежити класифікацію характеристик, які притаманні фішинговим атакам.

Рис. Б.1. Ментальна мапа класифікації евристичних правил детектування фішингових сайтів

Дана мапа узагальнює ключові ознаки, що використовуються для аналізу URL-адрес, дозволяючи структуровано представити критерії оцінки потенційної загрози.

Новизна запропонованого підходу полягає в інтеграції та вдосконаленні матеріалів, отриманих із різних джерел інформації, що дозволило систематизувати наявні методи детектування фішингу та подати їх у більш зручній для аналізу формі. Завдяки ментальній мапі спеціалісти з інформаційної безпеки отримують зручний інструмент для практичного використання у виявленні фішингових вебсайтів.

Ефективність моделі

Тестові дані використовуються для оцінки точності моделі, що класифікує URL-адреси як фішингові або легітимні. У результаті цих оцінок кожна URL-адреса поділяється на чотири категорії, а саме: True Positive, True Negative, False Positive, False Negative [5]. Ці чотири класи формують матрицю помилок, яка використовується для оцінки ефективності моделі. Рівняння матриці помилок (1-5) показують, як обчислюються ці метрики.

Достовірність (Accuracy) це основний показник оцінки, який вимірює відсоток правильних прогнозів, зроблених моделлю. Він обчислюється шляхом ділення кількості правильних прогнозів на загальну кількість прогнозів, зроблених моделлю:

$$Accuracy = \frac{TruePositive + TrueNegative}{TP + TN + FP + FN}. \quad (Б.1)$$

Істинно позитивний рівень (TPR) або частка всіх фактичних позитивних результатів, які були правильно класифіковані як позитивні, також відомий як відгук:

$$TruePositiveRate(Recall) = \frac{TruePositive}{TruePositive + FalseNegative}. \quad (Б.2)$$

Рівень хибних спрацьовувань (FPR) - це частка всіх фактичних негативних результатів, які були помилково класифіковані як позитивні, також відомі як вірогідність помилкової тривоги:

$$FalsePositiveRank = \frac{FalsePositive}{FalsePositive + TrueNegative}. \quad (Б.3)$$

Точність (Precision) це частка всіх позитивних класифікацій моделі, які насправді є позитивними:

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}. \quad (Б.4)$$

Оцінка F-1 - це середньозважена оцінка точності та відгуку, де ваги однакові. Він використовується для збалансування компромісу між точністю та пригадуванням. Оцінка F-1 розраховується наступним чином:

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}. \quad (Б.5)$$

Наприклад, якщо модель має високу точність, але низький коефіцієнт пригадування, це означає, що вона робить менше хибних спрацьовувань, але пропускає багато вірних спрацьовувань. І навпаки, модель з високим рівнем відгуку, але низькою точністю робить більше хибних спрацьовувань, але фіксує більше істинних спрацьовувань. У таких випадках оцінка F-1 може допомогти визначити, яка модель є кращою.

Формування моделі та результати

Формування моделі виявлення фішингових URL-адрес базується на багатоступеневому підході, що включає підготовку вхідних даних, розробку алгоритмів їх обробки та оцінку ефективності застосованих евристичних правил. Спочатку для аналізу використовуються два датасети: один містить фішингові URL-адреси, а інший – легітимні. Для реалізації моделі розробляється ПЗ, яке використовує набір евристичних правил для перевірки URL-адрес.

Програма зчитує набори даних, перевіряє кожне посилання на відповідність заданим як функції правилам і підраховує, скільки з них відповідають певним критеріям. Для оцінки достовірності краще використовувати декілька наборів функцій, які поступово розширюють можливості перевірки даних, підраховуючи кількість збігів серед перевірених даних.

Наступним етапом є обчислення ймовірностей спрацювання кожного правила, а також ймовірностей виявлення цих ознак серед URL-адрес у вибірці. Для цього визначаються показники ймовірності спрацювання правил

на фішингових сайтах з обох датасетів, а також ймовірності їх виявлення серед обраних URL з датасету, тобто визначення істинно позитивного рівня спрацювання.

Таким чином, отримані результати попередньо підтверджують ефективність застосування евристичних правил для аналізу фішингових вебресурсів. Ці можливості можуть бути використані для виявлення загрозливих URL-адрес з високою точністю, що вкрай важливо для автоматизованих систем кібербезпеки.

Після попереднього етапу аналізу, який базується на спрацюванні евристичних правил класифікації, необхідно оцінити ефективність розробленої методології. Для цього використовуються ключові метрики, які визначають якість алгоритму та його здатність розрізняти фішингові та легітимні сайти, а саме: достовірність, істинно позитивний рівень, рівень хибних спрацьовувань, точність та оцінка F1. Ці метрики дозволяють оцінити якість алгоритму з точки зору його здатності правильно класифікувати URL-адреси, виявляти фішингові сайти, а також уникати помилкових спрацьовувань.

Достовірність показує, наскільки правильно класифіковані раніше евристичні правила визначили усі перевірені URL-адреси.

Істинно позитивний рівень дозволяє оцінити, якою мірою метод виявляє фішингові сайти, не залишаючи їх без уваги. Високий TPR означає, що алгоритм ефективно виявляє фішингові загрози; низький TPR означає, що багато фішингових сайтів залишаються нерозпізнаними, що є серйозною проблемою для безпеки користувачів.

Рівень хибних спрацьовувань відображає частку легітимних сайтів, помилково класифікованих як фішингові. Якщо FPR алгоритму занадто високий, це означає, що його прогнози надто агресивні і можуть негативно вплинути на користувачів, оскільки вони можуть втратити доступ до легальних ресурсів. У таких випадках необхідно оптимізувати евристичні правила, щоб знизити FPR без суттєвого зменшення TPR.

Точність показує, яка частка всіх вебсайтів, які алгоритм ідентифікує як фішингові, насправді є фішинговими. Висока точність означає, що алгоритм рідко помиляється у виявленні загроз і має мало хибних спрацьовувань.

Так як оцінка F-1 є середнім гармонійним значенням точності і істинно позитивним рівнем, а отже, дозволяє оцінити, наскільки добре алгоритм балансує між виявленням загроз і мінімізацією хибних спрацьовувань. Ця метрика особливо корисна, коли важливо уникнути помилкової класифікації легальних сайтів, а також виявлення фішингових сайтів; високе значення F1 означає, що алгоритм успішно вирішує обидва завдання – правильного виявлення загроз і мінімізації помилкових спрацьовувань.

Підсумкові результати, отримані шляхом обчислення зазначених метрик, підтверджують ефективність застосованих евристичних правил у виявленні фішингових вебсайтів та їх здатність ідентифікувати потенційні загрози з високою ймовірністю та точністю.

Застосування евристичних правил для виявлення фішингу дозволяє відчутно скоротити обчислювальні витрати, бо такі способи не вимагають складних математичних обчислень і здатні функціонувати у реальному часі. Через свою гнучкість правила легко підлаштовуються до нових загроз, які змінюються разом із тактиками зловмисників, забезпечуючи високу результативність моделі. Це робить евристичні підходи ефективним знаряддям для виявлення загроз у мінливому середовищі кібербезпеки.

Висновок

У даній роботі було проведено ґрунтовний аналіз методів виявлення фішингових вебсайтів на основі евристичних правил та класифіковано їх, згідно систематизованих ознак та інших методів оцінки поведінки вебсайтів. Класифікація евристичних правил дозволяє ідентифікувати специфічні патерни активності, які є типовими для фішингових ресурсів, наприклад, неправильна інтеграція зовнішніх джерел чи маніпулятивна обробка введених користувачами даних.

Результати дослідження підтвердили, що комплексний підхід до виявлення фішингових сайтів є найефективнішим. Маючи наглядну класифікацію евристичних правил детектування, значно підвищується точність ідентифікації фішингових вебсайтів, захищаючи користувачів від можливих атак. Цю точність можна виміряти за допомогою ключових метрик ефективності алгоритму.

Проведена робота не лише систематизує знання про сучасні методи виявлення фішингових сайтів, але й забезпечує практичну основу для створення автоматизованих систем захисту від однієї з найнебезпечніших форм соціальної інженерії – фішингу. Впровадження таких систем дозволяє мінімізувати ризики компрометації даних, підвищуючи рівень безпеки як для окремих користувачів, так і для організацій, що особливо актуально в умовах постійного ускладнення методів зловмисників та їх адаптації до нових технологій

Література

[1] Що таке фішинг? | Захисний комплекс Microsoft. Microsoft. URL: https://www.microsoft.com/uk-ua/security/business/security-101/what-is-phishing (дата звернення: 20.03.2025).

[2] Phishing, Technique T1566 - Enterprise | MITRE ATT&CK®. MITRE ATT&CK®. URL: https://attack.mitre.org/techniques/T1566/ (дата звернення: 20.03.2025).

[3] Alabdan R. Phishing attacks survey: types, vectors, and technical approaches. Future internet. 2020. Т. 12, № 10. С. 168. URL: https://doi.org/10.3390/fi12100168 (дата звернення: 20.03.2025).

[4] Serhii Buchyk, Dmytro Shutenko and Serhii Toliupa “Phishing Attacks Detection”. In: Information Technology and Implementation (IT&I-2022). 2022, pp. 193-200.

[5] A hybrid DNN–LSTM model for detecting phishing URLs / A. Ozcan et al. Neural Computing and Applications. 2021. URL: https://doi.org/10.1007/s00521-021-06401-z (дата звернення: 20.03.2025).