DOI: 10.18372/2225-5036.29.18069

ДОСЛІДЖЕННЯ ВРАЗЛИВОСТЕЙ У ЧАТБОТАХ З ВИКОРИСТАННЯМ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ

Андріян Піскозуб, Даниїл Журавчак, Анастасія Толкачова

Національний університет «Львівська політехніка»

ПІСКОЗУБ Андріян Збігнєвич, к.т.н., доц.
Рік та місце народження: 1969 рік, м. Львів, Львівська область, Україна.
Освіта: Національний університет «Львівська Політехніка».
Посада: доцент кафедри захисту інформації Національного університету «Львівська політехніка».
Наукові інтереси: інформаційна безпека, комп'ютерні мережі, захист інформації в комп'ютерних мережах.
Публікації: 75 наукових публікацій, серед яких монографії, наукові статті та тези і матеріали доповідей на конференціях.
E-mail: azpiskozub@gmail.com.
Orcid ID: 0000-0002-3582-2835.

ЖУРАВЧАК Даниїл Юрійович, аспірант
Рік та місце народження: 1996 рік, м. Львів, Львівська область, Україна.
Освіта: Національний університет «Львівська Політехніка».
Посада: асистент кафедри захисту інформації Національного університету «Львівська політехніка».
Наукові інтереси: інформаційна безпека, реагування на інциденти інформаційної безпеки, штучний інтелект.
Публікації: більше 15 наукових публікацій, серед яких монографії, наукові статті та тези і матеріали доповідей на конференціях.
E-mail: danyil.y.zhuravchak@lpnu.ua.
Orcid ID: 0000-0003-4989-0203.

ТОЛКАЧОВА Анастасія Юріївна, студентка
Рік та місце народження: 2000 рік, м. Київ, Київська область, Україна.
Освіта: Національний університет «Львівська політехніка», 2023 рік.
Посада: старший фахівець з тестування систем захисту інформації з 2022 року.
Наукові інтереси: інформаційна безпека, реагування на інциденти інформаційної безпеки, тестування на проникнення.
Публікації: більше 5 наукових публікацій, серед яких наукові статті, матеріали та тези доповідей на конференціях.
E-mail: anastasiia.tolkachova.mkbst.2022@lpnu.ua.
Orcid ID: 0000-0002-8196-7963.

Анотація*. У сучасному світі штучний інтелект, особливо в області великих мовних моделей, набуває все більшого значення, зокрема, у формі чатботів. Але разом із бурхливим розвитком цієї технології зростає і кількість потенційних вразливостей. У цій науковій статті автори ретельно досліджують можливі вразливості таких чатботів, звертаючи особливу увагу на аспекти безпеки, включаючи специфічні функції, параметри та взаємодію з зовнішніми ресурсами. Окрім того, стаття наголошує на недостатніх аспектах сучасних методів тестування цих додатків, які переважно орієнтуються на сценарії атаки потенційного зловмисника, не розглядаючи повну картину можливих загроз. Відіграючи важливу роль, пропозиції щодо покращення тестування включають детальну перевірку коду на вразливості, систематичну валідацію вхідних даних, контроль взаємодії з зовнішніми ресурсами та формулювання конструктивних рекомендацій щодо усунення виявлених вразливостей. Враховуючи наближення ери все більш розповсюдженого застосування ШІ, ці пропозиції є особливо актуальними для підтримки високого рівня безпеки в чатботах, що використовують великі мовні моделі, та подальшого розвитку безпечних практик у цій сфері.*

Ключові слова*: чатбот, Штучний Інтелект, вразливості, кібербезпека, ChatGPT, LLM, owasp.*

Постановка проблеми

До недавнього часу про чат-боти, які допомагають бізнесу, ніхто і не знав, але зараз вони одні із лідерів на ринку як інструмент, котрий підвищує ефективність, економить час та гроші компанії. Перевагою чат-боту є швидкість, тому що клієнт одразу отримує відповідь і може обробляти одразу дуже багато заявок. Тому, використання ботів на даний момент, є дуже актуальним для ведення бізнесу онлайн в наші часи.

У сучасному цифровому світі штучний інтелект (ШІ) та великі мовні моделі (LLM) поступово стають складовими взаємодії людей з технологіями. Чатботи, розроблені на основі великих мовних моделей, стають все більш поширеними [10]. Проте, незважаючи на всі їх переваги, ці інструменти мають свої вразливості, що можуть бути використані зловмисниками для зловживань.

З розширенням використання великих мовних моделей у чатботах, стає все більш важливим забезпечити безпечність цих систем. Зокрема, необхідно глибше розуміння можливих вразливостей таких систем, а також розробка ефективних стратегій та методів їх усунення. Ці зусилля є ключовими для забезпечення надійності та довіри до використання великих мовних моделей в чатботах.

Нещодавно було помічено, що є загроза витоку інформації у чатботах з великими мовними моделями [11]. Це є серйозною проблемою, особливо в сферах, де обробляються конфіденційні дані. Це може статися через різні причини, включаючи помилки в коді, недостатній захист даних або неправильне використання технології.

Один з прикладів цього – коли код компанії стає доступним широкому загалу [9]. Це може статися, коли програміст використовує чатбот для автоматизації процесу написання коду, але не розуміє, що він може мати доступ в мережу. Це призводить до витоку конфіденційної інформації або навіть до злому системи. Також є ризик, що чатботи, такі як ChatGPT, можуть "видавати бажане за дійсне". Це означає, що вони можуть генерувати відповіді, які виглядають правдивими, але насправді є помилковими або вводять в оману. Особливо проблематично це в банківській сфері, де точність інформації є критично важливою. Тому в багатьох компаніях, зокрема в банках, ChatGPT заборонений. Вони можуть блокувати використання таких чатботів, щоб захистити свої системи та дані від потенційних загроз. Все це підкреслює важливість розуміння та управління ризиками, пов'язаними з використанням чатботів та інших форм автоматизованого взаємодії.

У цій статті ми проведемо дослідження вразливостей в чатботах, що використовують великі мовні моделі, а також надамо пропозиції щодо покращення тестування цих систем. Ми наголосимо на важливості валідації вхідних даних, контролю взаємодії з зовнішніми ресурсами та пошуку способів для усунення виявлених вразливостей. Нашою метою є надання прозорих та конструктивних рекомендацій, які допоможуть підтримувати високий рівень безпеки при використанні великих мовних моделей в чатботах, а також сприятимуть подальшому розвитку безпечних практик у цій сфері.

Історія чатботів починається ще в 1950-х роках, коли Алан Тюрінг представив свій відомий "тест Тюрінга", концепція якого полягала у визначенні, чи може машина вести себе так, щоб її не відрізнили від людини [5]. Ця ідея стала фундаментом для створення першого чатбота, Еліза, в 1966 році в MIT. Еліза була досить примітивна за сучасними стандартами, але вона вже вміла вести розмову, використовуючи попередньо запрограмовані відповіді на певні фрази або ключові слова.

Протягом наступних декількох десятиліть було створено багато інших чатботів, таких як Пері, А.Л.І.С.А та навіть Тренер Альба, що використовували більш складні методи моделювання розмови і були спрямовані на різні конкретні завдання [5].

Однак справжній прорив у розвитку чатботів стався з початком ери штучного інтелекту та машинного навчання. Сучасні асистенти, такі як Сірі від Apple та Алекса від Амазон, здатні аналізувати величезні обсяги даних та використовувати ці знання для покращення своєї взаємодії з користувачами (рис. 1).

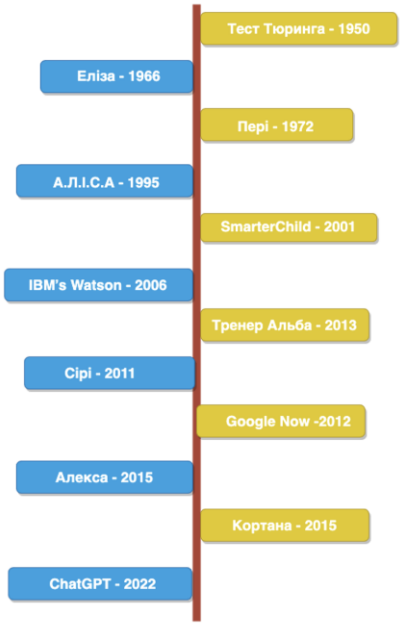

Рис. 1. Узагальнена хронологія створення чатботів

Великі мовні моделі, як такі, що використовуються в GPT-3 від OpenAI, представляють собою ще один крок у напрямку створення більш інтелектуальних та переконливих чатботів. Завдяки своїм 175 мільярдам параметрів [1], GPT-3 володіє здатністю ство-

рювати вражаюче людські текстові відповіді, що дозволило розробити нові та інноваційні чатботи.

З розвитком великих мовних моделей чатботи стають все більш продуктивними і переконливими у своїх відповідях, надаючи можливість для їх більш широкого застосування в різних сферах: від обслуговування клієнтів до освіти, медицини, розваг та багатьох інших. Однак разом із цим розвитком з'являється необхідність у ефективніших та надійних механізмах безпеки, які б враховували специфічні вразливості, які можуть мати великі мовні моделі в чатботах.

Аналіз останніх досліджень і публікацій

Великі мовні моделі, які використовуються в чатботах, викликають певні питання з приводу безпеки. У зв'язку з тим, що чатботи часто взаємодіють з великими об'ємами даних та зовнішніми системами, вони можуть стати потенційними цілями для кібератак. Основними проблемами тут є контроль за вхідними даними, безпека зберігання та обробки даних, а також взаємодія з іншими системами.

Організації та дослідники вже вивчають ці питання. Наприклад, OpenAI активно досліджує ці проблеми і працює над тим, щоб зробити свої моделі, такі як GPT-3, більш безпечними. Вони вже розробили ряд методів для зменшення ризику, включаючи фільтрацію вмісту, обмеження швидкості відповіді та контроль за джерелами даних.

Організація OWASP (Open Web Application Security Project), відома своїми напрацюваннями у сфері кібербезпеки, включає великі мовні моделі в список найпопулярніших вразливостей веб-додатків, що показує, наскільки важливими є ці питання.

Також багато вчених та дослідників вже вивчають можливі вразливості в чатботах з використанням великих мовних моделей [4]. Вони виявили, що деякі атаки, які були успішними проти веб-додатків, також можуть бути ефективними проти чатботів. Зокрема, техніки, такі як SQL-ін'єкція або атаки переповнення буфера, можуть бути адаптовані для використання проти чатботів [6].

Проте, є ще багато роботи в цій області, і важливо продовжувати досліджувати і розробляти нові методи захисту для забезпечення безпеки чатботів.

Великі зусилля докладаються, щоб розкрити ризики та загрози, пов'язані з недостатньою безпекою чатботів, які можуть призвести до несанкціонованого доступу до конфіденційної інформації, порушення приватності користувачів та інших проблем [7]. Зазначена інформація буде корисною для розробників чатботів, спеціалістів з кібербезпеки, а також всіх, хто цікавиться розумінням та покращенням безпеки використання великих мовних моделей у чатботах.

Мета та постановка завдання

Ціллю нашого дослідження є пропозиція розробки варіантів для усунення або пом'якшення виявлених вразливостей, а також залучення уваги до необхідності подальшого дослідження та усунення вразливість у чатботах, які використовують LLM. Наступна частина статті буде присвячена детальному аналізу основних вразливостей і порівнянню зі списком OWASP Top 10 для веб-додатків, що допоможе зрозуміти і діагностувати потенційні ризики та побудувати більш безпечні та надійні чатботи.

Метою даної роботи є дослідження та порівняння існуючих загроз, а також надання пропозицій для покращення рівня безпеки додатків із використанням великих мовних моделей.

Виклад основного матеріалу дослідження

У рамках дослідження розглянуті основні вразливості зі списку OWASP Top 10 для великих мовних моделей та їх порівняння з відповідними вразливостями з OWASP Top 10 2021 року для веб-додатків [3]. Основні типи атак, які було опубліковано OWASP для великих мовних моделей:

1. Ін'єкція у вікно введення запиту. Дозволяє змінити велику мовну модель (LLM) через введення спеціальних шкідливих даних, що викликає непередбачувані дії;

2. Незахищена обробка виводу. Без фільтрації даних це може призвести до таких атак як міжсайтовий скриптинг (XSS), підвищення привілеїв або виконання віддаленого коду;

3. Отруєння навчальних даних. Це відбувається, коли навчальні дані LLM підробляються, вносячи вразливості або упередження, які ставлять під загрозу безпеку, ефективність або етичну поведінку. Джерела включають Common Crawl, WebText, OpenWebText та книги;

4. Модель відмови в обслуговуванні. Надсилаючи велику кількістю запитів можна викликати відмову у обслуговуванні. Також сама робота LLM вимагає великих обсягів ресурсів, а користувачі мають можливість запитувати будь-яку інформацію;

5. Вразливості ланцюгів постачання. Вразливості можуть бути на будь-якому рівні додатку. Тобто, це може бути застаріле програмне забезпечення чи інші неправильні конфігурації на рівні серверу, сервісів та плагінів;

6. Розкриття конфіденційної інформації. Чатботи можуть дати відповідь із конфіденційними даними;

7. Небезпечна архітектура плагіна. Вразливою частиною плагінів є поля для вводу. Їх можна використати для ін’єкцій, що пізніше може стати причиною віддаленого виконання коду. Також сюди відносять проблеми із керуванням доступу;

8. Надмірна свобода дій. Причиною вразливості є надмірна функціональність, надмірні дозволи або надмірна автономія, яку використовують зловмисники у своїх цілях;

9. Надмірна довіра. Користувачі можуть отримувати дезінформацію через неправильно згенерований контент чатботом;

10. Крадіжка моделі. Тобто, все що стосується несанкціонованого доступу, порушення конфіденційності та цілісності запатентованих моделей LLM. Наслідками можуть стати фінансові збитки та втрату конкурентних переваг.

Основні типи атак, які було опубліковано OWASP TOP 10 2021 року (табл. 1):

1. Порушення контролю доступу. Дозволяє несанкціонованим особам отримати необмежений або несанкціонований доступ до обмежених ресурсів та даних;

2. Криптографічні помилки. Вразливість пов'язана з слабким використанням криптографічних алгоритмів та методів шифрування. Недостатня захищеність криптографії може призвести до зламу даних, порушення конфіденційності та іншим проблемам з безпекою системи [8];

3 . Ін'єкції. Зловмисник впроваджує шкідливий код або небезпечні дані у вхідні параметри додатка, викликаючи тим неочікувану поведінку;

4. Небезпечна архітектура. Додаток міг бути неналежним чином розробленим. Це в свою чергу створює слабкі місця та вразливості, які можуть бути використані зловмисниками для атак і компрометації системи;

5. Неправильна конфігурація безпеки. Налаштування додатка або системи прописуються з помилками або залишаються за замовчуванням. Це створює вразливості та ризики для злому і несанкціонованого доступу;

6. Вразливі та застарілі компоненти. Додаток використовує компоненти з відомими вразливостями або застарілі версії, які можуть створити ризик для безпеки системи та спричинити можливість злому додатка через використання цих слабкостей;

7. Помилки ідентифікації та аутентифікації. Додаток може мати логічні проблеми з реалізацією аутентифікації чи ідентифікації. Через це зловмисник може отримати доступ до чужих акаунтів або даних;

8. Порушення цілісності програмного забезпечення та даних. Вразливість дозволяє змінювати або порушувати цілісність програмного забезпечення або даних. Призводить до неконтрольованої зміни інформації, втрати даних або спотворення їх правильного функціонування;

9. Порушення моніторингу інцидентів у системі безпеки. Процес систематичного спостереження за подіями та активністю у системі з метою виявлення потенційних кібератак або порушень безпеки. Якщо у процесі є порушення чи інші фактори, що створюють слабкі місця, то їх може бути використано.

10. Підробка запитів на стороні сервера (Server-Side Request Forgery). Вразливість, при якій зловмисник може використовувати додаток для виконання шкідливих запитів до внутрішніх ресурсів сервера або зовнішніх систем. Це може призвести до розкриття конфіденційної інформації, атак на внутрішні сервіси або навіть компрометації системи.

Таблиця 1

Порівняння найпопулярніших вразливостей

№	OWASP Top 10 для веб-додатків	OWASP Top 10 для Великих мовних моделей
1	Порушення контролю доступу	Ін'єкція у вікно введення запиту
2	Криптографічні помилки	Незахищена обробка виводу
3	Ін'єкції	Отруєння навчальних даних
4	Небезпечна архітектура	Модель відмови в обслуговуванні
5	Неправильна конфігурація безпеки	Вразливості ланцюгів постачання
6	Вразливі та застарілі компоненти	Розкриття конфіденційної інформації
7	Помилки ідентифікації та аутентифікації	Небезпечна архітектура плагіна
8	Порушення цілісності програмного забезпечення та даних	Надмірна свобода дій
9	Порушення моніторингу інцидентів у системі безпеки	Надмірна довіра
10	Підробка запитів на стороні сервера	Крадіжка моделі

OWASP Top 10 2021 року для веб-додатків орієнтований на вразливості пов'язані зі створенням, реалізацією та взаємодією з веб-додатками. Вони фокусуються на слабких місцях, які можуть бути експлуатовані через зловмисний ввід даних, неправильні механізми аутентифікації та інші проблеми, що можуть здатися систематичними та впливати на безпеку вебдодатків.

З іншого боку, вразливості великих мовних моделей стосуються безпеки самого алгоритму чи моделі, а також безпеки середовища виконання, таких як недостатні захист криптографічних ключів, контроль доступу до конфіденційних даних, безпека взаємодії з моделлю та інші аспекти.

Обидва набори вразливостей є важливими для забезпечення повноцінної безпеки в інформаційних системах та мовних моделях. Вони вимагають ретельного аналізу, попередньої перевірки та правильних заходів безпеки для ефективного захисту систем та даних від можливих загроз.

Однак, можна провести деяке порівняння між ними, що може бути корисним. Узагальнюючи, деякі

категорії вони можуть бути схожими (наприклад, недостатня аутентифікація). Загальне порівняння між вразливостями з OWASP Top 10 2021 року для веб-додатків та великими мовними моделями (LLM):

1. Характер вразливостей:

- OWASP Top 10 для веб-додатків – фокусується на вразливостях, пов'язаних із захистом та безпекою веб-додатків, які зазвичай взаємодіють з користувачами та опрацьовують вхідні дані;

- великі мовні моделі (LLM) – орієнтовані на вразливості, що стосуються самої моделі та її середовища виконання, а також можливості зловмисника вплинути на роботу моделі та отримати доступ до конфіденційної інформації;

2. Види атак:

- OWASP Top 10 для веб-додатків – включає вразливості, пов'язані зі зловживанням вхідних параметрів, недостатньою аутентифікацією, атаками на доступність та іншими аспектами безпеки веб-додатків;

- великі мовні моделі (LLM) – стосуються вразливостей, що можуть призвести до зміни роботи моделі, отримання доступу до конфіденційних даних, а також до атак на саму інфраструктуру виконання моделі;

3. Залежність від вхідних даних:

- OWASP Top 10 для веб-додатків – зазвичай вразливі до зловживання вхідних даних, таких як форми, параметри URL, куки, тощо;

- великі мовні моделі (LLM) – можуть працювати з текстовими даними та надходженням інформації через API або інші інтерфейси;

4. Аспекти безпеки:

- OWASP Top 10 для веб-додатків – зосереджено на аутентифікації, авторизації, валідації вхідних даних, обробці сесій, захисті конфіденційної інформації та забезпеченні безпеки сервера;

- великі мовні моделі (LLM) – стосуються недостатньої обробки текстових даних, криптографічних невдач, контролю доступу до моделі та інших аспектів безпеки мовних моделей;

5. Призначення:

- OWASP Top 10 для веб-додатків – захист веб-додатків та даних користувачів, збереження конфіденційності, цілісності та доступності веб-порталів, інтернет-магазинів та інших онлайн-сервісів;

- великі мовні моделі (LLM) – безпека мовних моделей, які використовуються для генерації тексту, перекладу, аналізу даних та інших завдань, де моделі мають значний вплив на рішення або взаємодію з користувачами.

Загалом, хоча обидва набори вразливостей мають різні характеристики, вони є важливими для забезпечення безпеки інформаційних систем та моделей. При розробці та використанні веб-додатків та великих мовних моделей, слід приділяти увагу попередньому тестуванню на вразливості, правильному налаштуванню захисту та впровадженню кращих практик безпеки, щоб забезпечити надійний рівень захисту даних та систем від можливих загроз.

Опираючись на ці дані, можна допустити, що так само, як і у тестуванні веб додатків, потрібно розвивати інструменти для автоматизації тестування, а також піднімати питання перевірки програмного коду на вразливості. Також оскільки чатботи мають інтерфейс для користувача, все одно є необхідність перевіряти чатбот на усі вразливості зі списку тестування для веб-додатків. Отже, можна зробити висновок, що OWASP Top 10 не вміщує в себе всі вразливості і його потрібно розширювати до більшої кількості пунктів. Оскільки для розроблення великих мовних моделей використовують найчастіше мови програмування C++, Python та Java, то першим кроком авторами пропонується створити перевірочний лист для коду на C++, Python та Java. Ці мови програмування володіють потужними інструментами для обробки тексту, навчання машин та обробки даних, що робить їх популярними серед дослідників та розробників великих мовних моделей.

Серед бібліотек та фреймворків, які використовуються для розробки великих мовних моделей, можна назвати такі, як Hugging Face Transformers, OpenAI GPT, BERT, XLNet, Tensorflow, PyTorch та інші [2, 13, 14]. Ці бібліотеки надають інструменти та готові реалізації для створення та навчання великих мовних моделей, що спрощує процес розробки та дослідження в цій області.

Розглянемо список деяких вразливих функцій та параметрів, які можуть впливати на створення великих мовних моделей в мовах програмування Python, Java та C++.

Python

1) eval() та exec() – ці функції дозволяють виконувати довільний код, що може призвести до ін'єкції шкідливого коду у великі мовні моделі;

2) pickle – модуль pickle використовується для серіалізації об'єктів, але неконтрольоване використання цього модуля може призвести до виконання небезпечного коду під час десеріалізації;

3) subprocess – неконтрольована вставка даних у функції subprocess може призвести до створення процесів з небезпечними командами (наприклад, виконання віддаленого коду - remote code execution (RCE));

4) input() – небезпечна функція для отримання користувацьких вхідних даних без належної валідації;

5) os.system() – функція для виконання команд системної оболонки, яка може бути використана для RCE;

6) urllib.urlopen() – неконтрольований доступ до URL може використовуватися для атак типу SSRF (Server-Side Request Forgery);

7) sqlite3.execute() – неконтрольовані SQL-запити можуть призвести до ін'єкції SQL-коду;

8) import() – небезпечна функція для динамічного імпорту модулів, яка може бути використана для ін'єкції коду;

9) format() – недостатній контроль над відбитками форматування може призвести до RCE та інших проблем безпеки;

10) request.get() - неконтрольоване використання HTTP-запитів може призвести до атак типу CSRF (Cross-Site Request Forgery).

Java

1) java.io.ObjectInputStream – недостатнє фільтрування даних під час десеріалізації може призвести до виконання небезпечного коду;

2) java.net.Socket – неконтрольоване використання функцій для створення сокетів може призвести до можливості DoS атак;

3) Runtime.exec() – неконтрольоване виконання команд системної оболонки може призвести до ін'єкції шкідливого коду;

4) URL.openStream() – неконтрольований доступ до URL може використовуватися для атак типу SSRF (Server-side request forgery);

5) SQL PreparedStatement – неконтрольовані SQL-запити можуть призвести до ін'єкції SQL-коду;

6) JSP Expression Language (EL) – неконтрольований доступ до EL може призвести до XSS-вразливостей;

7) javax.crypto.Cipher – неправильне використання криптографічних методів може призвести до криптографічних недоліків;

8) TrustManager – неконтрольований довірчий менеджер може призвести до ненадійного криптографічного протоколу TLS;

9) URLClassLoader – неконтрольований доступ до класів може призвести до ін'єкції коду;

10) XML Parser – неконтрольована обробка XML-даних може призвести до атак типу XXE (XML External Entity).

C++

1) C++ ін'єкції відбитків форматування – неконтрольоване використання функцій форматування (наприклад, printf()) може призвести до переповнення буфера та вразливостей з відбитками форматування;

2) C++ пам'ять – неправильне управління пам'яттю може призвести до вразливостей, таких як неконтрольовані вказівники та переповнення буфера;

3) scanf() – недостатній контроль вхідних даних може призвести до переповнення буфера;

4) strcpy() – неконтрольоване копіювання рядків може призвести до переповнення буфера;

5) cin – неконтрольоване введення даних користувачем може призвести до переповнення буфера;

6) std::system() – виконання команд системної оболонки без належного контролю може призвести до RCE;

7) C++ бібліотеки шифрування – використання застарілих алгоритмів шифрування може призвести до криптографічних недоліків;

8) atoi() та atof() – недостатній контроль перетворення рядка в число може призвести до некоректних обчислень;

9) fopen() – неконтрольована робота з файлами може призвести до небезпечного зчитування або запису;

10) C++ вказівники – неконтрольована робота з пам'яттю може призвести до неконтрольованого доступу до даних.

На основі поглибленого аналізу вразливих функцій та параметрів, які можуть стати потенційною загрозою для безпеки великих мовних моделей в межах різних мов програмування, можна стверджувати про важливість обережного та уважного ставлення до перевірки і валідації вхідних даних. Вирішальною є необхідність контролю за взаємодією з зовнішніми ресурсами, а також застосування надійних та перевірених бібліотек, які б допомогли забезпечити високий рівень безпеки при використанні великих мовних моделей.

Дослідження підкреслює недостатність поверхового тестування чатботів, що використовують великі мовні моделі, лише зі сторони можливого зловмисника. Часто ця процедура не виявляє всіх потенційних ризиків. Суттєвою є ретельна перевірка коду на наявність вразливостей, аналіз їх характеру, включення таких вразливостей до вже існуючого переліку потенційних проблем, а також надання вичерпних рекомендацій щодо їх усунення.

Ці заходи дозволять виявити та нейтралізувати можливі загрози до того, як вони зможуть завдати шкоди. Даний висновок є результатом нашого дослідження, і ми сподіваємось, що він сприятиме розвитку безпечних практик у сфері використання великих мовних моделей.

Висновки. Стаття охоплює детальне дослідження та порівняння найпопулярніших вразливостей, які зустрічаються у веб-додатках та великих мовних моделях. Цей аналіз виявив цікаві спільні та унікальні характеристики вразливостей в обох контекстах, вказуючи на необхідність універсального підходу до безпеки інформації.

Веб-додатки та великі мовні моделі часто стикаються з різними викликами в області безпеки, проте є велика кількість спільних вразливостей. Наприклад, недостатнє управління сеансами, вбудований небезпечний код та відмова у службі виявились типовими для обох сфер.

Порівняння вразливостей в обох контекстах показало, що хоча великі мовні моделі можуть здаватися безпечнішими на перший погляд через відсутність безпосередньої взаємодії з користувачами або даними, вони все ще можуть бути вразливими до атак.

Автори статті пропонують ряд пропозицій щодо покращення списку найпопулярніших вразливостей великих мовних моделей. Ці пропозиції більше фокусуються на відкритті нових потенційних вразливостей, а також на розробку стратегій їх усунення.

Загалом, ця стаття вносить важливий внесок у розуміння вразливостей, які можуть виникнути у вебдодатках та великих мовних моделях, а також в розробку ефективних стратегій їх усунення. Вона наголо-

шує на важливості постійного вдосконалення безпеки в обох областях, оскільки зловмисники постійно розробляють нові способи атаки.

Список літератури

[1]. "AI and cybersecurity: The future of cyber defence" [Електронний ресурс]. Режим доступу до ресурсу: https: // www.forbes.com / sites / andrewrossow/ 2021 / 06 / 01 / ai-and-cybersecurity-the-future-of-cyber-defense/.

[2]. "How to build your private LLM?" [Електронний ресурс]. Режим доступу до ресурсу: https:// www.leewayhertz.com/build-private-llm/.

[3]. "OWASP Top 10 – 2021" [Електронний ресурс]. Режим доступу до ресурсу: https://owasp. org/ Top10/.

[4]. "OWASP Top 10 for Large Language Model Applications" [Електронний ресурс]. Режим доступу до ресурсу: https: // owasp.org / www-project-top-10-for-large-language-model-applications/.

[5]. "The History of Chatbots" [Електронний ресурс]. Режим доступу до ресурсу: https://onlim.com/ en/the-history-of-chatbots/.

[6]. "The role of AI in cybersecurity" by John Boitnot [Електронний ресурс]. Режим доступу до ресурсу: https: // venturebeat. com / 2018 / 03 / 31 /the-role-ofai-in-cybersecurity/.

[7]. Глорін Себастьян. Конфіденційність і захист даних у ChatGPT та інших чат-ботах зі штучним інтелектом: Стратегії захисту інформації про користувачів. Міжнародний журнал з безпеки та конфіденційності у сфері повсюдних обчислень, 2023, 15(1), с. 14.

[8]. Джин Ю. , Єн-Лінь Ч., Ліп П., Чін Су К. Систематичний огляд літератури про інформаційну безпеку в чат-ботах. Криптографія та інформаційна безпека. 2023, 13(11), с. 6355.

[9]. "Знайдені вразливості в AI Chatbots" [Електронний ресурс]. Режим доступу до ресурсу: https:// fagenwasanni. com / news / vulnerabilities-found-in-ai-chatbots/87954/.

[10]. К'яра Валентина Місіск'я а, Флора Поче б, Крістін Штраус. Чат-боти в клієнтському сервісі: Їх актуальність та вплив на якість обслуговування. Інформатика. 2022, С. 421-428.

[11]. Кларк, Дж., Джейкоб, Дж. (2018). ШІ та кібербезпека: загрози та рішення. Журнал кібербезпеки,4(1), С. 1-14.

[12]. Опірський І.Р., В.А. Сусукайло, С.І. Василишин. Дослідження можливостей використання чатботів зі штучним інтелектом для дослідження журналів подій. Безпека інформації. 2023. С. 177-183.

[13]. Чжан Ю., Чен В., Ян Дж. та Сю В. (2019). Машинне навчання в кібербезпеці. IEEE Access, 7, С. 108700-108707.

[14]. Чжан, Х., Хуан, Х., і Чжан, Ю. (2019). Дослідження прогресу штучного інтелекту в кібербезпеці. Міжнародний журнал систем обчислювального інтелекту, 12 (1), С. 316-326.

УДК 004.056.55

Piskozub A., Zhuravchak D., Tolkachova A. Researching vulnerabilities in chatbots with LLM (Large language model)

***Abstract**. In today's world, artificial intelligence, especially in the field of large language models, is becoming increasingly important, particularly in the form of chatbots. However, along with the rapid development of this technology, the number of potential vulnerabilities is also growing. In this research article, the authors thoroughly investigate the possible vulnerabilities of such chatbots, paying special attention to security aspects, including specific functions, parameters, and interaction with external resources. In addition, the article emphasizes the shortcomings of current testing methods for these applications, which mainly focus on attack scenarios of a potential attacker without considering the full picture of possible threats. Suggestions for improving testing include detailed vulnerability scanning, systematic validation of input data, control of interaction with external resources, and formulation of constructive recommendations for addressing identified vulnerabilities. Given the approaching era of increasingly widespread use of AI, these suggestions are particularly relevant for maintaining a high level of security in chatbots that use large language models and further developing secure practices in this area.*

***Keywords**: chatbot, Artificial Intelligence, cybersecurity, vulnerabilities, ChatGPT, LLM, owasp.*

Піскозуб Андріян Збігнєвич, к.т.н., доцент кафедри захисту інформації Національного університету «Львівська політехніка».
Andriyan Piskozub, PhD., Associate Professor of the Department of Information Protection of the National University "Lviv Polytechnic".

Журавчак Даниїл Юрійович, аспірант, асистент кафедри захисту інформації Національного університету «Львівська політехніка».
Daniil Zhuravchak, graduate student, assistant of the Department of Information Protection of the National University "Lviv Polytechnic".

Толкачова Анастасія Юріївна, старший фахівець з тестування систем захисту інформації кафедри захисту інформації Національного університету «Львівська політехніка».
Anastasia Tolkachova, Senior Specialist in Testing Information Protection Systems of the Department of Information Protection of the National University "Lviv Polytechnic".

Отримано *19 вересня 2023 року,* затверджено редколегією *30 жовтня 2023 року*