УДК 004.056

Л. М. Куперштейн
Д. С. Тіщенко

СИСТЕМА ВИЯВЛЕННЯ SQL–ІН'ЄКЦІЙ МЕТОДАМИ МАШИННОГО НАВЧАННЯ

Вінницький національний технічний університет

Анотація
Проведено дослідження присвячене виявленню SQL–ін'єкцій методами машинного навчання з метою підвищення ефективності виявлення SQL–ін'єкцій за допомогою використання методів машинного навчання. Для побудови моделі досліджено ряд класифікаторів.
Ключові слова: *SQL–ін'єкція, виявлення SQL–ін'єкцій, веб–додаток, машинне навчання, збір даних.*

Abstract
The study was devoted to the detection of SQL injections using machine learning methods in order to increase the effectiveness of SQL injection detection using machine learning methods. A number of classifiers were studied to build the model.
Keywords*: SQL injection, detection of SQL injections, web application, machine learning, data collection.*

Вступ

Безпека пристроїв, систем і мереж є життєво важливою, оскільки світ постійно розвивається і залежить від автоматизованих складних систем. Дослідники розробляють різні методи і засоби виявлення аномалій, щоб призупинити та контролювати вплив загроз на системи [1]. Однак існуючі рішення часто не працюють, коли мова йде про пристосування до постійно–змінної архітектури застосунків.

З кожним роком зростає кількість компаній, які використовують веб–технології для підвищення продуктивності та залучення нових клієнтів. Такі організації включають державні та місцеві органи влади, а також комерційні компанії різних форм власності. Інтернет–сервіси несуть з собою безліч переваг, але з ростом числа додатків збільшується і кількість кіберзагроз [2].

Ін'єкції є однією з найдавніших і найнебезпечніших загроз. Вони можуть завдати шкоди цілісності, конфіденційності та доступності даних. Атаки SQL–ін'єкції, також відомі як SQLIA, є однією з найбільш поширених загроз безпеці програм на основі баз даних, згідно з рейтингом OWASP (Open Web Application Security Project), який щороку створює рейтинг кіберзагроз, атаки типу ін'єкції, такі як SQL, NoSQL, OS і LDAP, були найпоширенішими кібератаками в 2017, 2018, 2020 та 2022 роках.

Зловмисник вносить довільний код до програми під час ін'єкційної атаки. Інтерпретатор обробляє цей код як частину команди або запиту. Це може змінити логіку роботи програми та призвести до непоправних наслідків [3].

Результати дослідження

На сьогодні існує багато методів та засобів виявлення та запобігання SQL–ін'єкціям, більшість з яких відрізняються один від одного. Ці методи використовують спеціальні інструменти та механізми, щоб виявити або запобігти зловмисним атакам SQL на цільову базу даних сервера. Розглянемо можливі симптоми SQL–ін'єкції:

– отримання великої кількості запитів за короткий проміжок часу, наприклад, велика кількість електронних листів від форм зворотного зв'язку веб–сайту;
– блоки реклами, які перенаправляють користувачів на підозрілі веб–сайти;
– дивні спливаючі вікна та повідомлення про помилки [4].

Завдання запобігання атакам SQL-injection є дійсно складним питанням. Зловмисники завжди можуть отримати доступ до баз даних через лазівку в коді.

За результатами аналізу існуючих методів захисту було встановлено, що методи машинного навчання є одним із найкращих засобів захисту веб–додатків від SQL–ін'єкцій. Їх успішно використовують при створенні сучасних систем виявлення атак і є перспективним напрямом розвитку в цій галузі, оскільки вирішення схожих завдань поступово підвищує точність рішень [5].

Для реалізації методу машинного навчання важливим кроком є збір і підготовка даних для створення набору даних. Адже тільки завдяки якісному набору даних можна досягти високої точності передбачення [6].

Етап моделювання неможливий без технічної підготовки даних. Така підготовка включає в себе перевірку усіх типів даних та зведення їх до одного, що найкраще підходить для виконання аналізу обраними моделями та методами. Необхідно збалансувати дані, якщо вони цього потребують. Це покращить результат тренування моделей машинного навчання. І ключовий етап це розділення датасету на тренувальну вибірку та тестову. Такий крок дозволить оцінити якість побудованих моделей. Для навчання моделі машинного навчання було використано датасет SQL–ін'єкцій, який складається з 25000 записів Даний датасет є збалансованим, адже 11382 записів містять атаки, а 13618 не містять, що становить 46% та 54% [7].

Для сприйняття даних моделлю машинного навчання їх необхідно перетворити у зрозумілий для неї вигляд. Перетворення символьних даних у векторні дані є звичайним етапом попередньої обробки в завданнях обробки природної мови, де вхідні дані складаються з тексту. Для попередньої обробки було обрано метод Bag–of–Words (BoW) , який представляє кожен документ як вектор, де кожен елемент відповідає частоті символу. Цей підхід відкидає порядок символів [8].

Для виконання задачі виявлення SQL–ін'єкцій було проведено моделювання ряду класифікаторів, таких як:

– Згорточна нейронна мережа (CNN) – це тип алгоритму глибокого навчання. Він складається з кількох шарів, включаючи згорткові шари, шари об'єднання та повністю зв'язані шари.

– Gaussian Naive Bayes – вирішення задачі класифікації, базуючись на незалежності кожної пари ознак (дані для кожного представника обрані з простого розподілу Гаусса);

– Support Vector Machine (SVM) — це контрольований алгоритм машинного навчання, який використовується як для класифікації, так і для регресії.

– K–Nearest Neighbors – встановлення мітки класу, використовуючи схожість ознак» найближчих k сусідів;

– DecisionTree – встановлення мітки класу, використовуючи набір правил у вигляді дерева;

– LogisticRegression – ймовірності описують можливі результати, використовуючи логістичну функцію [9].

Для моделювання також було визначено найкращий набір гіперпараметрів для кожного з класифікаторів за допомогою алгоритму GridSearch, суть якого – перебрати перелік гіперпараметрів моделі, виконати аналіз оцінки і отримати перелік параметрів, при яких модель має найкращі результати.

Для визначення ефективності класифікатора було досліджено помилки першого і другого роду. Помилка першого роду полягає в тому, що буде відхилена правильна гіпотеза. Помилка другого роду полягає в тому, що буде прийнята неправильна гіпотеза. Наслідки цих помилок різноманітні і можуть мати тяжкі наслідки.

На основі результатів було обрано класифікатор з найкращими показниками – дерево рішень.

Спроектовано архітектуру моделі машинного навчання, яка складається з трьох модулів:

1. Модуль моніторингу HTTP–запитів виконує аналіз структури HTTP–запиту та перевірку HTTP–запиту на основі шаблону.
2. Модуль інтелектуального аналізу використовує навчену модель машинного навчання для виявлення SQL–ін'єкцій. Для вибору методу машинного навчання, який буде використано для виявлення SQL–ін'єкцій, необхідно провести моделювання різних методів машинного навчання, виконати аналіз результатів та відповідно до цього сформувати висновок.
3. Модуль збереження даних у базу даних виконує запис у файл для подальшого збільшення об'єму дата сету та покращення в майбутньому результатів передбачення моделі машинного навчання.

Розглянемо алгоритм роботи програмного засобу:

1. Отримання параметрів HTTP–запитів – на цьому процесі виконується збір вхідних параметрів HTTP–запитів, які надсилаються до веб–застосункку.
2. Виконується перевірка на наявність SQL ін’єкції на основі шаблонів.
3. Якщо виявлено ознаки SQL ін’єкції користувач отримує про це повідомлення, а запит записується до бази даних. Якщо ні, то наступним кроком виконується перевірка за допомогою моделі машинного навчання.
4. За допомогою навченої моделі виконується аналіз SQL-коду у параметрах HTTP–запитів, та якщо такі слова були знайдені, то даний запит буде вважатися SQL–ін’єкцією.
5. Формування даних про SQL–для передачі цих даних до процесу сповіщення користувача про загрозу та збереження до бази даних.
6. Якщо не виявлено ознак SQL–ін'єкції запит записується до бази даних і повертається запитувана інформація назад до користувача.

Програмну реалізацію застосунку було успішно виконано мовою Python. Тестування було проведено на спеціально створеному для перевірки web-додатку, вразливому до SQL–ін'єкцій. Розроблений програмний продукт пройшов тестування. Тестування проводилося за двома різними сценаріями роботи: з шкідливими та безпечними запитами. Користувачу надсилаються відповідні повідомлення для усунення проблем, а запити обробляються. Модель машинного навчання має досить високу точність, на тестовій виборці показало 95% передбачення наявності /відсутності SQL–ін'єкції в запитах.

Висновки

В ході дослідження було встановлено, що вразливості та загрози впровадження SQL–ін'єкцій входять до списку найбільш розповсюджених загроз цілісності, конфеденційності та доступності даних. Було спроектовано універсальну архітектуру системи виявлення SQL–ін'єкцій методами машинного навчання. Обраний підхід на основі машинного навчання забезпечує високий рівень достовірності виявлення SQL–ін'єкцій у веб-застосунках.

Основними напрямками подальшого вдосконалення вбачаються розширення існуючуго набору даних

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. SQL ін’єкції в MySQL сервері URL: https://www.securitylab.ru/contest/212083.php (дата звернення: 08.11.2023).
2. SQL Injection Cheat Sheet URL: https://www.netsparker.com/blog/web–security/sql–injection–cheat–sheet/ (дата звернення: 08.11.2023).
3. OWASP CheatSheetSeries URL: https://github.com/OWASP/CheatSheetSeries/blob/master/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.md(дата звернення: 05.11.2023).
4. SQL Injections Top Attack Statistics URL: https://www.darkreading.com/risk/sql–injections–top–attack–statistics/d/d–id/1132988 (дата звернення: 05.11.2023).
5. Best Practices to prevent SQL–injections URL: https://tableplus.io/blog/2018/08/best–practices–to–prevent–sql– injection–attacks.html (дата звернення: 05.11.2023).
6. SQL Injection Attacks Are Rampant: How to Stop Your Next Hack Attack URL: https://goo.gl/RxnbWp. (дата звернення: 05.11.2023).7. Cloudflare Business Plan URL: https://www.cloudflare.com/plans/business/ (дата звернення: 05.11.2023).
8. SQL Injection Prevention Cheat Sheet URL: https://goo.gl/N1NHtW. (дата звернення: 05.11.2023)..
9. Libinjection URL: https://github.com/client9/libinjection(дата звернення: 05.11.2023).

Куперштейн Леонід Михайлович – к.т.н., доцент кафедри захисту інформації, Вінницький національний технічний університет, м. Вінниця, e-mail: kupershtein.lm@gmail.com

Тіщенко Дарина Сергіївна – студентка групи 1БС-22м, факультет інформаційних технологій та комп’ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: daria.tsc@gmail.com

Kupershtein Leonid M. – PhD, Associated Professor of Information Protection Chair, Vinnytsia National Technical University, Vinnytsia, e-mail: kupershtein.lm@gmail.com

Tishchenko Daryna Serhiyivna – *student of Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: daria.tsc@gmail.com*