Сравнительный анализ подходов к обнаружению SQL-инъекций с помощью методов машинного обучения

Е.А. Юдова, О.Р. Лапонина

***Аннотация* – Информационная безопасность не стоит на месте, сейчас появляется большое количество способов защиты от различных видов уязвимостей. Для каждой атаки можно найти много способов ее предотвращения и последующего обнаружения. В данной работе рассмотрены различные подходы к выявлению SQL-инъекций и проведен их сравнительный анализ, с целью выявления оптимального способа, в зависимости от условий работы. Были выделены основные критерии для рассмотрения статьи в ходе работы. В частности, рассматривались только статьи, опубликованные после 2016 года и доступные в полнотекстовом формате.**

Также определены основные характеристики исследований, которые проводились в ходе работ. В проанализированных работах рассматривались способы классификации как статических, так и динамических SQL-запросов. Для классификации использовались различные модели машинного обучения, в том числе: наивный байесовский метод, метод опорных векторов, дерево синтаксического анализа. В ряде работ самыми эффективными методами оказались модели с усилением ансамбля (Ensemble Boosted Trees), упакованные деревья (Ensemble Bagged Trees), линейный дискриминант (Linear Discriminant), кубический метод опорных векторов (Cubic SVM) и точный гауссовский метод опорных векторов (Fine Gaussian SVM). В других работах было обнаружено, что лучшую точность имеют методы итеративный дихотомизатор 3 (ID3) и случайный лес (Random Forest).

***Ключевые слова* – SQL-инъекции, обнаружение атак, машинное обучение, RF, SVM, ID3, Cubic SVM, Fine Gaussian SVM, TC SVM.**

I. Введение

В данный момент вопрос безопасности приложений, сайтов и других веб-ресурсов стоит очень остро. Нужны эффективные и быстрые способы защиты от различных атак. Так как информация может храниться и передаваться по-разному, то и количество атак и способов защиты от них вырастает в разы. В этой статье затрагивается тема SQL-инъекций и их обнаружение с помощью методов машинного обучения.

Статья получена 27 апреля 2023.

Е.А. Юдова – МГУ имени М.В. Ломоносова (email: KateCate.KY@gmail.com).

О.Р. Лапонина – МГУ имени М.В. Ломоносова (email: laponina@oit.cmc.msu.ru).

В ходе работы были проанализированы статьи, которые так или иначе связаны с данной тематикой. Чтобы сократить выборку материала, были введены критерии включения и исключения статьи из рассматриваемой выборки.

После проведения сравнительного анализа, были выделены самые эффективные методы машинного обучения и их комплексное применение для обнаружения атак. Далее методы были сгруппированы исходя из простого или комплексного подхода, учитывая происхождение наборов данных, на которых производились тесты.

II. Постановка задачи

A. Цели работы

1. Определить критерии выбора статей для рассмотрения и дальнейшего анализа
2. Выделить основные характеристики работ для последующего сравнения
3. Произвести сравнительный анализ и выделить основные методы машинного обучения, которые являются эффективными

B. Задачи работы

1. Определить ключевые слова для поиска статей
2. Ввести критерии включения и исключения статьи
3. Выделить основные методы получения данных, с которыми производилась работа и основные методы машинного обучения, которые имели наибольшую точность
4. Сравнить результаты, которые были получены в разных работах

III. Критерии выбора статей, анализирующих обнаружение SQL-инъекций

A. Критерии выбора ключевых слов в статьях

Правильное определение тематики работы очень важно, так как верное определение темы позволит найти нужные статьи в достаточном количестве.

Так как производится анализ SQL-инъекций и их обнаружения, то одним из ключевых слов будет **SQL-инъекции**. Самая главная цель – определить данную атаку, поэтому **обнаружение атак** тоже будет одним из

ключевых слов. Метод, с помощью которого будет производиться обнаружение атаки, тоже является немаловажным фактором. В данном случае рассматриваются только **методы машинного** обучения, поэтому оно тоже будет являться ключевым словом.

Выделенные ключевые слова являются и ключевыми словами для данной статьи, так как в полной мере характеризуют ее тематику.

B. Критерии включения и исключения статьи

Для начала были определены основные критерии включения статьи в данную работу.

1. Статьи, связанные с атаками с использованием SQL-инъекций.
2. Статьи, которые включают ключевые слова для поиска, определенные ранее.
3. Статьи из научных баз данных ACM, IEEE, SpringerLink и ScienceDirect.
4. Статьи на тему машинного обучения и предметной области безопасности.

Чтобы избежать избыточности рассматриваемой выборки статей, были определены критерии исключения статьи из рассмотрения.

1. Статьи, не охватывающие темы методов машинного обучения и атак с использованием SQL-инъекций.
2. Статьи, опубликованные ранее 2016 года.
3. Статьи, которые недоступны в полнотекстовом формате.

Таким образом, были определены основные критерии, которые позволили собрать и изучить статьи по требуемой тематике.

IV. Анализ рассмотренных статей

A. Рассматриваемые статьи

В большом количестве работ в качестве метрик качества моделей обнаружения SQL-инъекций используются следующие показатели [13].

True Positive (TP) – количество (процент) правильно классифицированного нормального трафика.

$$TP\ rate = \frac{TP}{TP+FN} \quad (1)$$

True Negative (TN) - количество (процент) правильно классифицированного аномального трафика.

$$TN\ rate = \frac{TN}{FP+TN} \quad (2)$$

False Negative (FN) - количество (процент) неправильно классифицированного нормального трафика (нормальный трафик классифицирован как аномальный).

$$FN\ rate = \frac{FN}{TP+FN} \quad (3)$$

False Positive (FP) - количество (процент) неправильно классифицированного аномального трафика (аномальный трафик классифицирован как нормальный).

$$FP\ rate = \frac{FP}{FP+TN} \quad (4)$$

Accuracy (доля верных ответов). Вычисляется по формуле:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (5)$$

Precision (точность). Определяет, на сколько можно доверять работе рассматриваемой модели. Определяется формулой:

$$Precision = \frac{TP}{TP+FP} \quad (6)$$

Recall (полнота). Показывает, как много объектов класса «аномальный трафик» распознает классификатор. Для вычисления используется формула:

$$Recall = \frac{TP}{TP+FN} \quad (7)$$

F1 score (F1) – это среднее гармоническое значение точности и полноты. Это хорошая сбалансированная мера как ложноположительных, так и ложноотрицательных результатов. Чем лучше результаты модели, тем ближе значение данной метрики к единице.

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (8)$$

В статье “Обзор атак с использованием SQL-инъекций” [2] авторы использовали для сравнения 23 разных метода классификации с помощью машинного обучения и собственный набор данных, в котором использовались в общей сложности 616 операторов SQL. Самыми эффективными методами оказались: модели с усилением ансамбля (Ensemble Boosted Trees), упакованные деревья (Ensemble Bagged Trees), линейный дискриминант (Linear Discriminant), кубический метод опорных векторов (Cubic SVM) и точный гауссовский метод опорных векторов (Fine Gaussian SVM).

Classifier	TP rate (%)	TN rate (%)	Accuracy (%)
Ensemble Boosted Trees	>99	64	93.8
Ensemble Bagged Trees	>99	64	93.8
Linear Discriminant	100	62	93.7
Cubic SVM	>99	63	93.7
Fine Gaussian SVM	100	61	93.5

Рис. 1. Результаты статьи [2]

В работе “Обнаружение атак SQL-инъекций с

использованием распознавания грамматических шаблонов и анализа поведения доступа" [3] авторы рассмотрели 5 алгоритмов кластеризации и классификации, в качестве датасета был взят журнал веб-доступа, который содержит перечень всех веб-запросов пользователей сети. Авторы извлекли грамматические и поведенческие особенности SQL-инъекций из логов с помощью предлагаемого распознавателя грамматических паттернов и определения поведения при получении доступа. Соответственно, получилось два тестовых набора данных:

1. Основанный на грамматических характеристиках (GFM)
2. Основанный на поведенческих характеристиках (BFM)

В результате работы было обнаружено, что лучшую точность имеют методы: итеративный дихотомизатор 3 (ID3) и случайный лес (Random Forest).

	Test set based on GFM		Test set based on BFM	
Model	**FN rate (%)**	**FP rate (%)**	**FN rate (%)**	**FP rate (%)**
K-means	34.5	10.9	18.2	16.7
Naive Bayes	41.4	4.1	18.2	8.3
SVM	41.4	0.0	18.2	0.0
ID3	37.9	0.68	**9.1**	0.0
RF	37.9	0.68	**9.1**	0.0

Рис. 2. Результаты статьи [3]

В процессе разработки прикладного корпуса, основанного на шаблонах, для прогностической аналитики для смягчения атаки SQL-инъекцией [4] использовались две модели классификации контролируемого обучения с алгоритмами двух классового метода опорных векторов (TC SVM) и двух классовой логистической регрессии (TC LR). Были искусственно созданы данные с использованием символьных конечных автоматов. Оценка с помощью ROC-кривой показала, что алгоритм TC SVM оказался немного точнее, чем TC LR.

Model	TP	TN	FP	FN	Accuracy	Precision
TC SVM	72 359	70 598	1 923	162	0.99	0.97
TC LR	69 421	70 433	2 088	3 100	0.96	0.97

Рис. 3. Результаты классификаторов статьи [4]

Трипати Д. и др. [5] создали набор данных, собрав и объединив большое количество небольших датасетов. Сгенерированный набор данных был размечен, и использовалось обучение с учителем. Они обучили семь моделей машинного обучения и сравнили их с точки зрения производительности и точности. Результаты показали, что классификатор случайного леса (Random Forest) превзошел все другие классификаторы и достиг точности 99,8%.

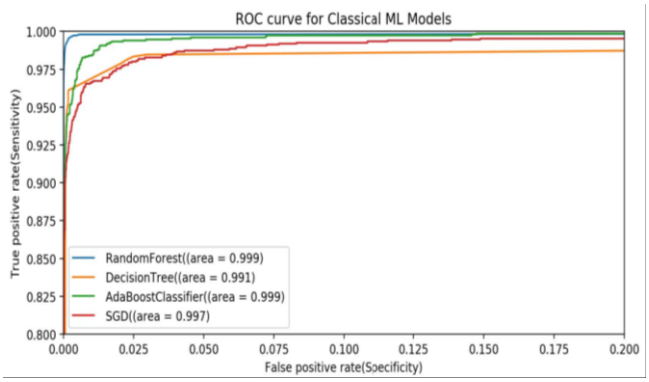

Рис. 4. ROC-кривая результатов статьи [5]

Камтуо К. и Соомлек Ч. [6] предложили структуру для предотвращения SQL-инъекций с помощью сценариев на стороне сервера с использованием платформ машинного обучения и компилятора. Был взят набор данных из 1100 образцов SQL-запросов, и результаты показали, что алгоритм расширяемых деревьев решений (Boosted Decision Tree) достиг наивысшей эффективности прогнозирования среди двух протестированных моделей машинного обучения.

Model	Precision	Accuracy	Processing Time
SVM	1.0	0.99	2.65 seconds
Boosted Decision Tree	1.0	1.0	5.22 seconds

Рис. 5. Результаты статьи [6]

В работе "Обнаружение SQL-инъекций с использованием алгоритма машинного обучения" [7] использовали алгоритм AdaBoost для обнаружения атак с использованием SQL-инъекций. В этом исследовании использовалась весовая классификация. Экспериментальный результат показал, что предложенный алгоритм точно и эффективно обнаружил атаки с использованием инъекций.

Дас Д. и др. [8] предложили метод классификации динамических SQL-запросов на основе веб-профиля, подготовленного на этапе обучения. Для процесса классификации использовались следующие методы: наивный байесовский метод (Naive Bayes), метод опорных векторов (SVM) и дерево синтаксического анализа (Parse Tree). Общая частота обнаружения с использованием двух наборов данных составила 91% и 90% соответственно. На рис. 6, 7 S1, S2, S3, S4, S5, S6 – различные датасеты. В работе были рассмотрены два сценария защиты:

1. Защита при использовании статических SQL-запросов
2. Защита при использовании динамических SQL-запросов

	S1	S2	S3	S4	S5	S6
Bayes Theorem	92	96	95	95	92	94
SVM	91	95	94	93	93	96
Parse tree	94	94	94	94	92	99
Proposed method (edit-distance)	94	93	96	96	92	93
Proposed method (binary-distance)	86	92	89	89	95	94

Рис. 6. Результаты статьи [8], статические SQL-запросы

	S1	S2	S3	s4	S5	S6
Bayes Theorem	67	66	75	65	62	65
SVM	51	85	74	63	63	66
Parse tree	64	64	74	66	62	68
Proposed method (using edit- distance)	92	93	94	96	91	89
Proposed method (using binary- distance	89	88	92	90	92	88

Рис. 7. Результаты статьи [8], динамические SQL-запросы

Касим О. [9] разработал метод обнаружения вредоносных SQL-запросов. Алгоритмы дерева решений (Decision Tree) с усилением их ансамбля (Ensemble Boosted Trees) использовались для процессов классификации для обнаружения различных SQL-инъекций. Предлагаемая модель обеспечивала точность более 98% при обнаружении атак с использованием SQL-инъекций и превзошла результаты других, рассмотренных в данной статье, моделей.

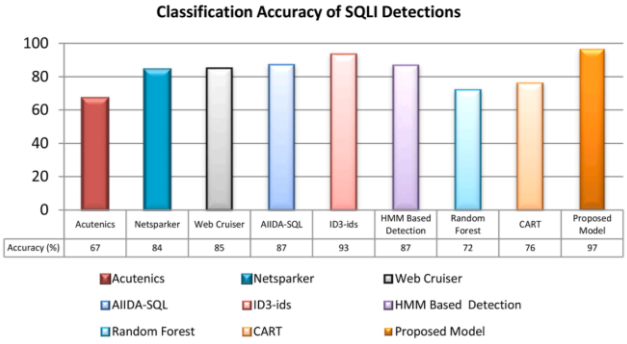

Рис. 8. Результаты статьи [9]

Кар Д. и др. [10] представили метод обнаружения атаки путем моделирования SQL-запросов в виде графа токенов и использовали меру центральности узлов для обучения машины опорных векторов (SVM). Были исследованы различные методы создания графов токенов и предложены альтернативные конструкции системы, состоящей из одного и нескольких SVM. Результаты эксперимента продемонстрировали, что предложенный метод способен эффективно идентифицировать вредоносные SQL-запросы.

Соломон О. и др. [11] представили модель двух классовой машины опорных векторов (TC SVM) для прогнозирования результатов с двоичными метками относительно того, была ли атака SQL-инъекцией успешной или не успешной в веб-запросе. Эта модель перехватывала веб-запросы на уровне прокси-сервера и применяла предиктивную аналитику машинного обучения для прогнозирования атак с использованием SQL-инъекций.

Outcomes	Values		Metrics	Values
TP	72 359		*Accuracy*	0.986
FN	162		*Precision*	0.974
FP	1 923		*Recall*	0.997
TN	70 598		*F1*	0.985

Рис. 9. Результаты статьи [11]

Маквиртер П. и др. [12] представили новый подход к классификации SQL-запросов. Для вычисления показателя сходства между строками запроса использовался алгоритм ядра подпоследовательности строк, взвешенный по пробелам. Алгоритм ядра реализован для идентификации подпоследовательности общих символов между строками запроса для вывода показателя сходства. Затем машина опорных векторов (SVM) была обучена метрикам сходства, чтобы определить, были ли строки запроса обычными или вредоносными. Предложенный подход был оценен с использованием ряда наборов данных SQL-запросов и достиг точности 92,48%.

Ван Ю. и др. [14] предложили гибридный подход с использованием древовидно-векторных ядер в методе опорных векторов (SVM) для изучения операторов SQL. Авторы использовали как древовидную структуру синтаксического анализа запросов SQL, так и характеристику сходства значений запросов, чтобы различать вредоносные и безопасные запросы. Результаты подтвердили преимущество интеграции для эффективного и точного выявления аномальных запросов. Здесь TPR и FPR это TP rate и FP rate, соответственно.

Kernel Type	TPR	FPR	Time(ms)
3-gram	0.640	0.000	0.233
4-gram	0.453	0.002	0.193
Vector	0.560	0.000	0.450
Tree-vector	0.982	0.000	3.862

Рис. 10. Результаты оценки качества алгоритмов, основанных на SVM, в статье [14]

Прияа Б. и др. [15] предложили структуру, которая объединила алгоритм эффективного дерева адаптивных решений по данным (EDADT) и алгоритм классификации SVM для обнаружения атак с использованием SQL-инъекций. Используемый набор данных был создан с использованием системы набора данных MovieLens [17], которая включала логин пользователя и сведения о фильме. Результаты эксперимента показали, что предложенный подход достиг точности 99,87%.

Techniques	Accuracy
Our Hybrid Approach	99.87%
Composite kernel in SVM[2]	99.6%
idMAS – SQL[3]	99.01%
SVM[4]	96.47%
C4.5 and SVM[5]	99.87%
EDADT[6]	98.12%

Рис. 11. Результаты статьи [15]

Джоши А. и др. [16] предложили метод обнаружения SQL-инъекции с использованием наивного байесовского алгоритма машинного обучения (Naive Bayes). Авторы применили процесс токенизации, чтобы разбить запрос на значимые элементы, называемые токенами. Затем список токенов стал исходным материалом для дальнейших процессов классификации. Результат наивного байесовского подхода был проанализирован с использованием точности, полноты и доли верных ответов.

Model	Accuracy	Precision	Recall
Naive Bayes	0.933	1.0	0.89

Рис. 12. Результаты статьи [16]

B. Обобщение по рассмотренным работам

Можно заметить, что авторы большей части статей рассматривали метод машинного обучения или комплексное использование нескольких методов в качестве собственной разработки. При этом наборы данных были разного вида, что в некоторой степени может повлиять на качество результатов. Рассмотрение различных методов машинного обучения для обнаружения SQL-инъекций позволит понять, какие методы являются эффективными, чтобы применить их в совокупности с другими методами и создать усовершенствованный подход к выявлению атак данного типа.

V. Основные характеристики статей

Ниже, на Рис. 13 представлена сравнительная таблица, в которой выделены основные параметры работ. Были выделены:

- Номер статьи в списке литературы. Для удобства вместо названия статьи был взят ее номер в списке литературы;
- Используемые наборы данных, которые использовались для тестирования методов;
- Методы машинного обучения, которые показали лучшие результаты;
- Является ли рассматриваемый метод комплексным или упомянутые методы рассматривались отдельно.

Данная таблица позволяет быстро определить основное направление статьи и эффективные методы для тестовых данных, которые использовались в представленной работе. На основе Рис. 13 была создана итоговая таблица (Рис. 14), которая объединила результаты всех рассмотренных статей.

Стоит отметить, что SQL-запросы, извлеченные из веб-запросов [3] были причислены к существующим датасетам, так как это данные, которые возникли в результате реального взаимодействия.

Номер статьи	Датасет	Методы ML	Комплексный метод (да/нет)
[2]	SQL запросы, созданные искусственно	Ensemble Boosted Trees Ensemble Bagged Trees Linear Discriminant Cubic SVM Gaussian SVM	нет
[3]	SQL запросы из веб-запросов	ID3 Random Forest	нет
[4]	SQL запросы, созданные искусственно	TC SVM	нет
[5]	объединение уже существующих датасетов	Random Forest	нет
[6]	SQL запросы, созданные искусственно	Boosted Decision Tree	нет
[7]	SQL запросы, созданные искусственно	AdaBoost	нет
[8]	динамические SQL запросы	Naive Bayes SVM Parse Tree	да
[9]	объединение уже существующих датасетов	Decision Tree Ensemble Boosted Trees	да
[10]	объединение уже существующих датасетов	SVM с весовой классификацией	да
[11]	SQL запросы, созданные искусственно	TC SVM	да
[12]	объединение уже существующих датасетов	алгоритм ядра подпоследовательности строк, взвешенный по пробелам SVM	да
[14]	SQL запросы, созданные искусственно	древовидно-векторные ядра и характеристика сходства значений SVM	да
[15]	SQL запросы, созданные искусственно	SVM EDADT	да
[16]	объединение уже существующих датасетов	Naive Bayes	да

Рис. 13. Характеристики всех работ

VI. Выводы

Данный анализ статей по теме выявления SQL-инъекций с помощью различных алгоритмов машинного обучения позволит исследователям в дальнейшем иметь представление о том, какие модели могут быть эффективны в их исследованиях, какие модели уже использовались для сравнения и оценки качества.

Благодаря итоговой таблице выделены важные факты:

1. Модели с комплексным подходом чаще используются и показывают хорошие результаты на реальных данных;
2. В комплексных методах часто используются весовые алгоритмы или характеристики сходства значений.

Из этого следует вывод, что для качественной работы собственного подхода к обнаружению SQL-инъекций, следует использовать комплексный метод, который будет включать в себя весовой алгоритм или текстовый анализатор.

Также, стоит заметить, что на искусственных данных результаты алгоритмов часто имеют хорошие показатели, что в дальнейшем может сказаться отрицательно, если данный алгоритм будет применяться в практике. Поэтому стоит тестировать модели не на

искусственно созданных данных, а на данных, собранных при реальном взаимодействии и из различных источников.

VII. Итоги

В ходе работы были установлены критерии рассмотрения статей. Изучено и проанализировано множество статей, которые отвечали установленным требованиям. В итоге, получен обобщенный результат (Рис. 14) по информации, которая была представлена в работах. Данную таблицу можно будет использовать при разработке собственных подходов к выявлению SQL-инъекций с помощью машинного обучения.

Датасет	Методы ML	Комплексный метод (да/нет)
SQL запросы, созданные искусственно	Ensemble Boosted Trees Ensemble Bagged Trees Linear Discriminant Cubic SVM Gaussian SVM TC SVM Boosted Decision Tree AdaBoost	нет
объединение уже существующих датасетов	Random Forest ID3	нет
динамические SQL запросы	Naive Bayes SVM Parse Tree	да
объединение уже существующих датасетов	Decision Tree Ensemble Boosted Trees Naive Bayes SVM в совокупности с: 1) весовой классификацией 2) алгоритмом ядра подпоследовательности строк, взвешенным по пробелам	да
SQL запросы, созданные искусственно	EDADT TC SVM SVM в совокупности с: 1) древовидно-векторными ядрами 2) характеристикой сходства значений	да

Рис. 14. Итоговый результат

Она поможет будущим авторам статей и разработчикам определить какие методы подходят лучше и уже были протестированы. Основные выводы, которые были сделаны в итоге исследования:

1. Тестирование на реальных данных помогает создать условия максимально приближенные к реальным, по сравнению с искусственно созданными запросами SQL;
2. Алгоритмы, которые состоят из нескольких методов машинного обучения и имеют в своей структуре синтаксический анализ или взвешенный алгоритм, показывают хорошие результаты на реальных данных.

VIII. Заключение

Что было сделано в ходе работы:

- Определены критерии, согласно которым статья рассматривалась или не рассматривалась в данной работе;
- Определена основная информация по всем рассмотренным работам;
- Выделены основные характеристики работы (датасет, методы машинного обучения, комплексность подхода);
- Выделены основные методы, которые показывают хорошее качество, в зависимости от типа данных и сложности подхода;
- Подведены итоги о лучших вариантах компоновки методов машинного обучения для выявления SQL-инъекций.

Библиография

[1] Marashdeh Z., Suwais K., Alia M. A survey on sql injection attack: Detection and challenges //2021 International Conference on Information Technology (ICIT). – IEEE, 2021. – С. 957-962.

[2] Hasan M., Balbahaith Z., Tarique M. Detection of SQL injection attacks: a machine learning approach //2019 International Conference on Electrical and Computing Technologies and Applications (ICECTA). – IEEE, 2019. – С. 1-6.

[3] Gao H. et al. Detecting SQL injection attacks using grammar pattern recognition and access behavior mining //2019 IEEE International Conference on Energy Internet (ICEI). – IEEE, 2019. – С. 493-498.

[4] Uwagbole S. O., Buchanan W. J., Fan L. An applied pattern-driven corpus to predictive analytics in mitigating SQL injection attack //2017 Seventh International Conference on Emerging Security Technologies (EST). – IEEE, 2017. – С. 12-17.

[5] Tripathy D., Gohil R., Halabi T. Detecting SQL injection attacks in cloud SaaS using machine learning //2020 IEEE 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing,(HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS). – IEEE, 2020. – С. 145-150.

[6] Kamtuo K., Soomlek C. Machine Learning for SQL injection prevention on server-side scripting //2016 International Computer Science and Engineering Conference (ICSEC). – IEEE, 2016. – С. 1-6.

[7] Sivasangari A., Jyotsna J., Pravalika K. SQL injection attack detection using machine learning algorithm //2021 5th International Conference on Trends in Electronics and Informatics (ICOEI). – IEEE, 2021. – С. 1166-1169.

[8] Das D., Sharma U., Bhattacharyya D. K. Defeating SQL injection attack in authentication security: an experimental study //International Journal of Information Security. – 2019. – Т. 18. – С. 1-22.

[9] Kasim Ö. An ensemble classification-based approach to detect attack level of SQL injections //Journal of Information Security and Applications. – 2021. – Т. 59. – С. 102852.

[10] Kar D., Panigrahi S., Sundararajan S. SQLiGoT: Detecting SQL injection attacks using graph of tokens and SVM //Computers & Security. – 2016. – Т. 60. – С. 206-225.

[11] Uwagbole S. O., Buchanan W. J., Fan L. Applied machine learning predictive analytics to SQL injection attack detection and prevention //2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM). – IEEE, 2017. – С. 1087-1090.

[12] McWhirter P. R. et al. SQL Injection Attack classification through the feature extraction of SQL query strings using a Gap-Weighted String Subsequence Kernel //Journal of information security and applications. – 2018. – Т. 40. – С. 199-216.

[13] Vinogradova E., Golovin E. Metriki kachestva algoritmov mashinnogo obuchenija v zadachah klassifikacii // Nauchnaja sessija GUAP. — 2017. — s. 202—206.

[14] Wang Y., Li Z. SQL Injection Detection via Program Tracing and Machine Learning //Internet and Distributed Computing Systems. – 2017. – С. 264-274.

[15] Priyaa, B.D.; Student, P.G.; Devi, M.I. Hybrid SQL Injection Detection System. In Proceedings of the 2016 3rd International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 22–23 January 2016.
[16] Joshi A., Geetha V. SQL Injection detection using machine learning //2014 international conference on control, instrumentation, communication and computational technologies (ICCICCT). – IEEE, 2016. – C. 1111-1115.
[17] Movie lens datasets https://grouplens.org/datasets/movielens/ Retrieved: Apr 2023

Comparative Analysis of SQL Injection Detection Approaches Using Machine Learning Methods

E.A. Yudova, O.R. Laponina

***Abstract* - Information security does not stand still, now there are a large number of ways to protect against various types of vulnerabilities. For every attack, you can find many ways to prevent and then detect it. In this paper, various approaches for identifying SQL injections are considered and their comparative analysis is carried out in order to identify the optimal method, depending on the working conditions. The main criteria for consideration of the article in the course of work were identified. In particular, only articles published after 2016 and available in full-text format were considered.**

Also, the main characteristics of the studies that were carried out in the course of the work were determined. In the analyzed works, the methods of classification both of static and dynamic SQL queries were considered. Various machine learning models were used for classification, including: Naive Bayes, Support Vector Machine, Decision Tree. In a number of studies, the most effective methods were Ensemble Boosted Trees, Ensemble Bagged Trees, Linear Discriminant, Cubic SVM, and Fine Gaussian Support Vector Machines. In other works, Iterative Dichotomizer 3 (ID3) and Random Forest methods were found to have better accuracy.

***Keywords* – SQL injection, attack detection, machine learning, RF, SVM, ID3, Cubic SVM, Fine Gaussian SVM, TC SVM.**

REFERENCES

[1] Marashdeh Z., Suwais K., Alia M. A survey on sql injection attack: Detection and challenges //2021 International Conference on Information Technology (ICIT). – IEEE, 2021. – C. 957-962.
[2] Hasan M., Balbahaith Z., Tarique M. Detection of SQL injection attacks: a machine learning approach //2019 International Conference on Electrical and Computing Technologies and Applications (ICECTA). – IEEE, 2019. – C. 1-6.
[3] Gao H. et al. Detecting SQL injection attacks using grammar pattern recognition and access behavior mining //2019 IEEE International Conference on Energy Internet (ICEI). – IEEE, 2019. – C. 493-498.
[4] Uwagbole S. O., Buchanan W. J., Fan L. An applied pattern-driven corpus to predictive analytics in mitigating SQL injection attack //2017 Seventh International Conference on Emerging Security Technologies (EST). – IEEE, 2017. – C. 12-17.
[5] Tripathy D., Gohil R., Halabi T. Detecting SQL injection attacks in cloud SaaS using machine learning //2020 IEEE 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing,(HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS). – IEEE, 2020. – C. 145-150.
[6] Kamtuo K., Soomlek C. Machine Learning for SQL injection prevention on server-side scripting //2016 International Computer Science and Engineering Conference (ICSEC). – IEEE, 2016. – C. 1-6.
[7] Sivasangari A., Jyotsna J., Pravalika K. SQL injection attack detection using machine learning algorithm //2021 5th International Conference on Trends in Electronics and Informatics (ICOEI). – IEEE, 2021. – C. 1166-1169.
[8] Das D., Sharma U., Bhattacharyya D. K. Defeating SQL injection attack in authentication security: an experimental study //International Journal of Information Security. – 2019. – T. 18. – C. 1-22.
[9] Kasim Ö. An ensemble classification-based approach to detect attack level of SQL injections //Journal of Information Security and Applications. – 2021. – T. 59. – C. 102852.
[10] Kar D., Panigrahi S., Sundararajan S. SQLiGoT: Detecting SQL injection attacks using graph of tokens and SVM //Computers & Security. – 2016. – T. 60. – C. 206-225.
[11] Uwagbole S. O., Buchanan W. J., Fan L. Applied machine learning predictive analytics to SQL injection attack detection and prevention //2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM). – IEEE, 2017. – C. 1087-1090.
[12] McWhirter P. R. et al. SQL Injection Attack classification through the feature extraction of SQL query strings using a Gap-Weighted String Subsequence Kernel //Journal of information security and applications. – 2018. – T. 40. – C. 199-216.
[13] Vinogradova E., Golovin E. Metriki kachestva algoritmov mashinnogo obuchenija v zadachah klassifikacii // Nauchnaja sessija GUAP. — 2017. — s. 202—206.
[14] Wang Y., Li Z. SQL Injection Detection via Program Tracing and Machine Learning //Internet and Distributed Computing Systems. – 2017. – C. 264-274.
[15] Priyaa, B.D.; Student, P.G.; Devi, M.I. Hybrid SQL Injection Detection System. In Proceedings of the 2016 3rd International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 22–23 January 2016.
[16] Joshi A., Geetha V. SQL Injection detection using machine learning //2014 international conference on control, instrumentation, communication and computational technologies (ICCICCT). – IEEE, 2016. – C. 1111-1115.
[17] Movie lens datasets https://grouplens.org/datasets/movielens/ Retrieved: Apr 2023