

УДК 004.85:004.056

Разработка системы автоматизированного анализа конфигурационных файлов с использованием машинного обучения для повышения кибербезопасности

Ивашкин Петр Владимирович

Волосатова Тамара Михайловна (научный руководитель)

МГТУ им. Н.Э. Баумана, Москва, Россия

Разработана система автоматизированного анализа конфигурационных файлов с применением методов машинного обучения, ориентированная на повышение уровня кибербезопасности. Предложен подход, основанный на модели случайного леса, позволяющий классифицировать файлы как корректные или содержащие потенциальные уязвимости и ошибки конфигурации. Для обучения использован синтетический набор данных конфигурационных файлов в формате JSON с различными характеристиками, включая параметры, критичные для безопасности. Программа реализует этапы генерации данных, извлечения признаков, обучения модели и интерпретации результатов. Проведенная оценка показала высокую эффективность предложенного решения в выявлении некорректных конфигураций. Разработанная система может быть применена для автоматизации контроля безопасности конфигураций в процессах разработки и эксплуатации информационных систем.

Ключевые слова: машинное обучение, кибербезопасность, конфигурационные файлы, JSON, автоматизированный анализ, классификация ошибок, выявление уязвимостей

Введение. Современные программные системы и приложения широко используют конфигурационные файлы для управления своим поведением, настройками и параметрами. Эти файлы, зачастую в форматах JavaScript Object Notation (JSON), Extensible Markup Language (XML) или YAML Ain't Markup Language (YAML), содержат критически важную информацию, от которой зависит не только функциональность, но и безопасность системы. Ошибки в конфигурационных файлах, такие как синтаксические неточности, отсутствующие или неверные параметры, а также небезопасные настройки, являются значительной угрозой кибербезопасности. Они могут приводить к сбоям в работе систем, несанкционированному доступу, утечкам данных и другим серьезным инцидентам безопасности.

Ручная проверка конфигурационных файлов на предмет соответствия политикам безопасности и отсутствия ошибок является трудоемким процессом, подверженным человеческому фактору, особенно в условиях большого количества файлов или их сложной динамической структуры. Традиционные анализаторы и линтеры, хотя и полезны для выявления синтаксических ошибок, не всегда способны идентифицировать сложные семантические ошибки или конфигурационные уязвимости, требующие контекстного понимания. В связи с этим актуальной задачей становится разработка интеллектуальных си-

стем для автоматизированного анализа корректности и безопасности конфигурационных файлов, использующих методы искусственного интеллекта.

Целью настоящей работы является разработка системы автоматизированного анализа конфигурационных файлов, использующей методы машинного обучения для классификации файлов на корректные и содержащие ошибки или потенциальные уязвимости. Такой подход позволяет снизить риски, связанные с некорректными конфигурациями, и является важным элементом в обеспечении кибербезопасности современных информационных систем. В статье описывается предлагаемый подход, включающий этапы подготовки данных, обучения модели на основе алгоритма случайного леса и оценки ее эффективности, с акцентом на анализ файлов в формате JSON [1, 2].

Постановка задачи и обзор подходов. Задача исследования заключается в разработке метода и системы для автоматической классификации конфигурационных файлов в формате JSON на два класса: «корректный/безопасный» и «ошибочный/небезопасный». Под ошибками и уязвимостями понимаются синтаксические нарушения, отсутствие обязательных параметров, некорректные типы или значения параметров, а также конфигурации, представляющие угрозу безопасности (например, отключенное шифрование, использование слабых учетных данных, активные отладочные режимы в производственной среде).

Формально, для каждого входного конфигурационного файла, представленного вектором признаков, необходимо построить классификатор, который относит файл к одному из двух классов.

Существующие подходы к анализу конфигурационных файлов включают ручную проверку, использование статических анализаторов (линтеров) и валидаторов схем (например, JSON Schema). Эти методы эффективны для выявления синтаксических ошибок и нарушений структуры, однако они ограничены в обнаружении семантических и контекстно-зависимых уязвимостей [3]. В последние годы активно исследуются методы машинного обучения для решения этой задачи, позволяющие выявлять более сложные закономерности и аномалии в конфигурациях [4, 5].

Предлагаемая система и метод. Архитектура предлагаемой системы автоматизированного анализа конфигурационных файлов представлена на рис. 1. Она включает следующие основные модули: подготовка данных, извлечение признаков, обучение модели и анализ конфигурации.

На этапе подготовки данных формируется набор JSON-файлов, которые размечаются на два класса. В данной работе использовался синтетический датасет для обеспечения контролируемых условий эксперимента. Файлы проходят предобработку и парсинг. На этапе извлечения признаков для каждого файла формируется вектор признаков, который используется для обучения модели. Признаки делятся на три группы. Структурно-синтаксические: оценивают общую структуру файла, его валидность согласно формату JSON, количество элементов, глубину вложенности и т. д. (например, `is_valid_json`, `num_top_level_keys`, `max_nesting_depth`).

Семантические и связанные с безопасностью: анализируют смысловое содержание, наличие или отсутствие критически важных параметров, использование небезопасных настроек, соответствие типов данных ожидаемым и т. п. (например, `missing_critical_keys`, `type_mismatch_count`, `unsafe_values_count`, `uses_weak_credentials`).

Лексические: характеризуют текстовые свойства файла, такие как средняя длина имен ключей, частота определенных токенов (например, `avg_key_name_length`).



Рис. 1. Архитектура системы

Процесс подготовки данных и извлечения признаков более детально показан на рис. 2.

В качестве модели машинного обучения используется алгоритм Случайный лес (Random Forest), представляющий собой ансамбль решающих деревьев. Метод случайного леса был выбран благодаря его высокой точности, устойчивости к переобучению, способности работать с разнородными признаками и возможности оценки их важности. Модель обучается на размеченных векторах признаков и затем используется для классификации новых, неразмеченных файлов.

Результаты вычислительных экспериментов. Для оценки эффективности предложенной системы были проведены вычислительные эксперименты. Использовался синтетический набор данных, состоящий из 2000 JSON-файлов, сбалансированных по классам (1000 корректных/безопасных и 1000 ошибочных/небезопасных). Данные были разделены на обучающую (70 %) и тестовую (30 %) выборки. Модель Случайного леса (100 деревьев) была обучена на обучающей выборке.



Рис. 2. Схема подготовки данных

Качество классификации на тестовой выборке оценивалось с помощью матрицы ошибок (рис. 3) и стандартных метрик (см. таблица).

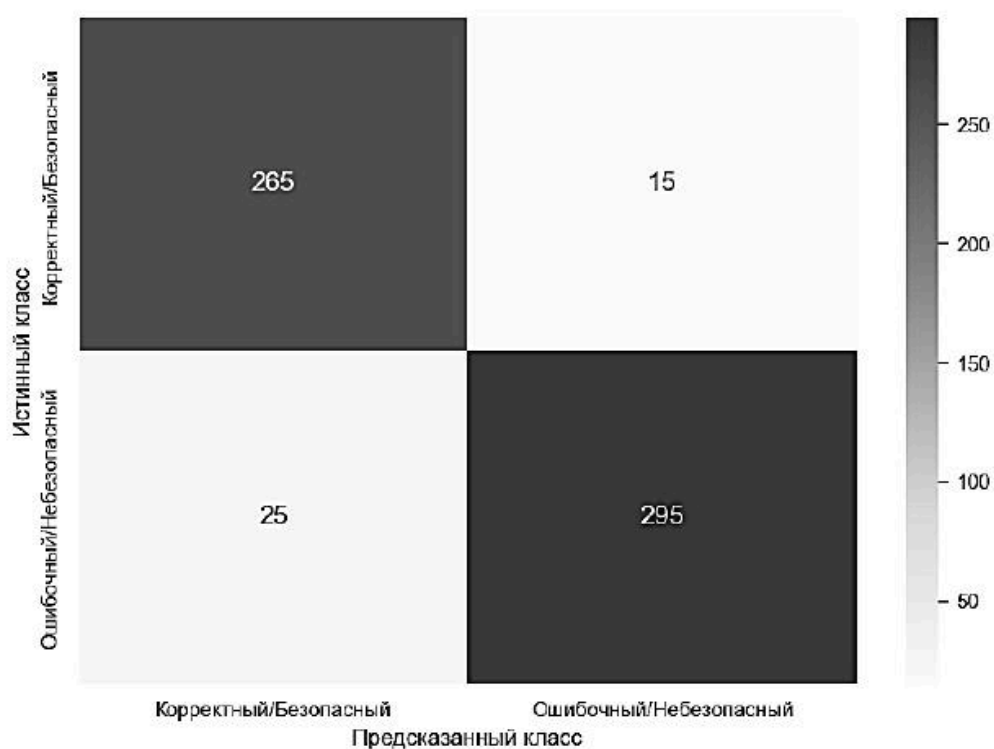


Рис. 3. Матрица ошибок

Высокие значения True Positive (TP = 295, правильно выявленные ошибки) и True Negative (TN = 265, правильно определенные корректные файлы) указывают на хорошую общую способность модели различать классы. Относительно низкие значения False Positive (FP = 15, ложные тревоги) и False Negative (FN = 25, пропущенные ошибки) свидетельствуют о высокой точности и полноте. Особое внимание уделялось минимизации FN, так как пропуск реальной уязвимости более опасен. Полученные значения метрик (табл.) подтверждают высокую эффективность предложенного подхода.

Результаты классификации: метрики качества

Метрика	Значение
Accuracy	0,933
Precision	0,952
Recall	0,922
F1-score	0,937

Анализ важности признаков, выполняемый моделью случайного леса, показал, что наибольший вклад в классификацию вносят признаки, связанные с синтаксической корректностью (`is_valid_json`), наличием критических для безопасности ключей (`missing_critical_keys`), несоответствием типов данных (`type_mismatch_count`) и наличием небезопасных значений (`unsafe_values_count`). Графическое представление важности признаков приведено на рис. 4.

Эти результаты подтверждают, что модель фокусируется на наиболее значимых аспектах конфигурационных файлов для выявления ошибок и уязвимостей.

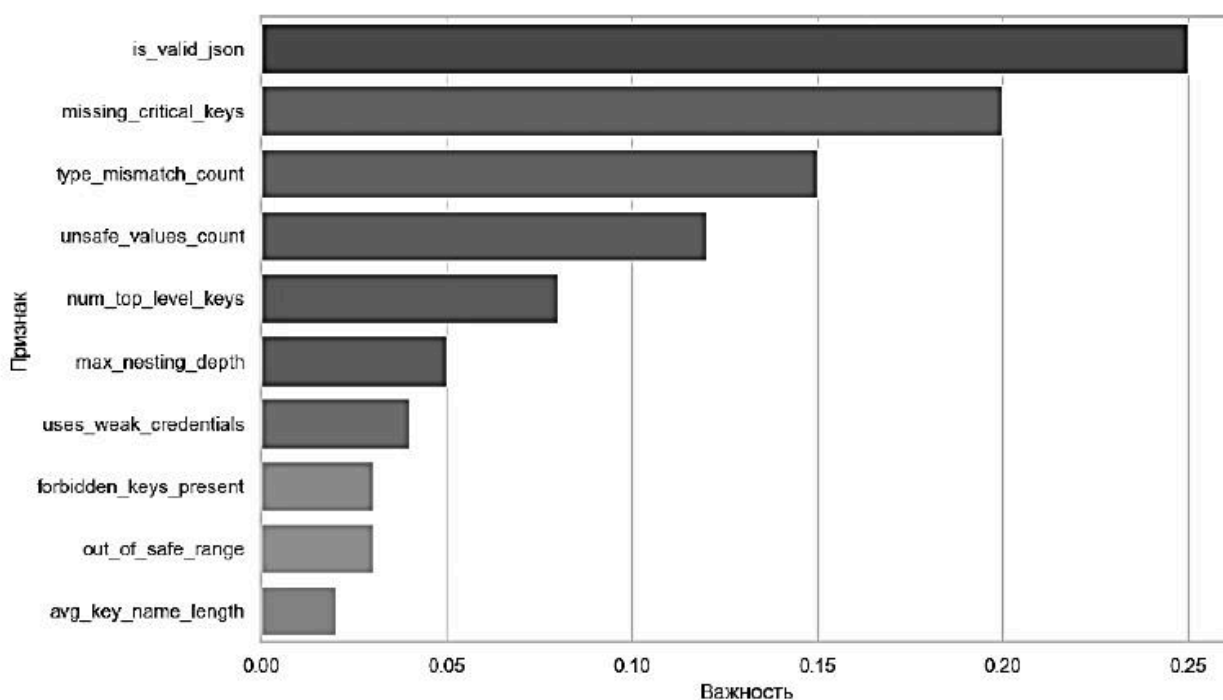


Рис. 4. Важность признаков

Закключение. В ходе исследования была проанализирована актуальность проблемы безопасности конфигурационных файлов и существующие подходы к ее решению. Разработана архитектура системы автоматизированного анализа JSON-конфигураций и предложен метод извлечения информативных признаков, включая специфичные для кибербезопасности. Экспериментально подтверждена высокая эффективность ($F1\text{-score} = 0,937$) обученной модели случайного леса в выявлении ошибок и уязвимостей. Разработанный подход автоматизирует контроль качества и безопасности конфигураций, обладая высокой точностью и способностью выявлять сложные паттерны ошибок. Основные ограничения связаны с качеством обучающих данных и обнаружением новых типов атак. Перспективы включают поддержку новых форматов, исследование более сложных моделей и интеграцию с CI/CD.

Литература

- [1] Турышев А.А., Назаров Д.М. Использование машинного обучения для обнаружения вредоносного трафика. *Актуальные проблемы развития технических наук: сб. ст.* Екатеринбург, Уральский федеральный университет имени первого Президента России Б.Н. Ельцина, 2021, с. 143–145.
- [2] Кондратенко Е.В., Макареня С.Н. Машинное обучение как инструмент анализа данных. *Информационные технологии в образовании, науке и производстве. IX Междунар. науч.-техн. интернет-конф.: материалы.* Минск, БНТУ, 2022, с. 304–309.
- [3] Baset S., Suneja S., Bila N et al. Usable declarative configuration specification and validation for applications, systems, and cloud. *Proceedings of the 18th ACM/IFIP/USENIX Middleware Conference. Industrial Track.* Las Vegas, NV, USA, 2017, pp. 29–35. <https://doi.org/10.1145/3154448.3154453>
- [4] Lian X., Chen Y., Cheng R., Huang J. et al. Configuration Validation with Large Language Models. *Arxiv:2310.09690*, 2023.
- [5] Santolucito M., Zhai E., Dhodapkar R. et al. Synthesizing configuration file specifications with association rule learning. *Proceedings of the ACM on Programming Languages*, 2017, vol. 1, iss. OOPSLA, no. 64, pp. 1–20. <https://doi.org/10.1145/3133888>

Development of an automated configuration file analysis system using machine learning for enhancing cybersecurity

Ivashkin Petr Vladimirovich

Volosatova Tamara Mikhailovna (*scientific advisor*)

BMSTU, Moscow, Russia

This paper presents an automated system for JSON configuration file analysis using machine learning to improve cybersecurity. A random forest model classifies files as correct or containing errors/vulnerabilities. Trained on synthetic data, the system showed high F1-score (0,937) in detecting misconfigurations, aiding automated security control.

Keywords: machine learning, cybersecurity, configuration files, JSON, automated analysis, random forest, vulnerability detection