DOI 10.37882/2223-2966.2023.9-2.28

ПРИЛОЖЕНИЯ МАШИННОГО ОБУЧЕНИЯ В РАЗРАБОТКЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ ПРЕДОТВРАЩЕНИЯ УЯЗВИМОСТЕЙ CSRF В ВЕБ-ПРИЛОЖЕНИЯХ

APPLICATIONS OF MACHINE LEARNING IN SOFTWARE DEVELOPMENT TO PREVENT CSRF VULNERABILITIES IN WEB APPLICATIONS

Chavez Quiroz Gabriela Guadalupe

Summary. This article dives into the application of Machine Learning (ML) in detecting Server-Side Request Forgery (SSRF) vulnerabilities in web applications. SSRF vulnerabilities present significant risks by allowing attackers to manipulate requests from the server, which can lead to the exposure of sensitive data or unauthorized intrusions. This article examines how ML has become an essential tool for addressing SSRF, highlighting examples of ML algorithms used to identify patterns and anomalies in web requests. In addition, the successful integration of ML-based solutions into the software development cycle is discussed, enabling early and effective detection of SSRF vulnerabilities. In an increasingly interconnected digital world, this approach is crucial to strengthening security in web applications and online systems.

Keywords: machine learning, vulnerability detection, SSRF, web application security, cybersecurity.

Чавес Кирос Габриэла Гуадалупе
аспирант, Санкт-Петербургский политехнический университет Петра Великого
chaveskiros.g@edu.spbstu.ru

Аннотация. В этой обзорной статье рассказывается о применении методов ML (Машинное обучение) в разработке программного обеспечения для предотвращения уязвимостей SSRF в веб-приложениях. Уязвимости SSRF (Подделка запроса на стороне сервера) представляют серьезную угрозу безопасности, позволяя злоумышленникам манипулировать запросами с сервера. В статье рассматривается, как машинное обучение стало важным инструментом раннего выявления и устранения этих уязвимостей. Описаны примеры алгоритмов машинного обучения, используемых в этом контексте, и способы их интеграции в цикл разработки программного обеспечения. Кроме того, обсуждаются технические проблемы, этические соображения и предлагается призыв к действию для сообщества разработчиков программного обеспечения для активного устранения этих критических уязвимостей.

Ключевые слова: машинное обучение, разработка программного обеспечения, SSRF, безопасность веб-приложений, обнаружение уязвимостей.

Введение

Сегодняшняя растущая зависимость от веб-приложений привела к постоянным опасениям по поводу кибербезопасности. Среди наиболее значимых угроз — уязвимости для SSRF, которые позволяют злоумышленникам отправлять вредоносные запросы с веб-сервера на внутренние или внешние ресурсы. Эти уязвимости могут привести к серьезным атакам, включая утечку конфиденциальных данных и несанкционированный доступ к внутренним системам. Разработка программного обеспечения стала фундаментальной основой, гарантирующей безопасность веб-приложений, и в этом контексте ML стало мощным инструментом. Эта обзорная статья посвящена применению методов ML в разработке программного обеспечения для предотвращения и смягчения уязвимостей SSRF. Мы рассмотрим, как ML стал важным союзником в раннем обнаружении этих угроз, представив примеры применяемых алгоритмов ML и подчеркнув их эффективную интеграцию в цикл разработки программного обеспечения.

По мере того, как мы вступаем в эпоху цифровых технологий, понимание и применение этих методов становится необходимым для обеспечения безопасности веб-приложений и защиты критически важных данных. Кроме того, мы рассмотрим технические проблемы и этические соображения в этом контексте и обратимся с призывом к действию к сообществу разработчиков программного обеспечения для укрепления кибербезопасности во все более взаимосвязанном мире.

SSRF-уязвимости

Уязвимости SSRF возникают, когда злоумышленник может заставить веб-приложение отправлять HTTP-запросы с сервера в пункт назначения, контролируемый злоумышленником, или к внутренним ресурсам, к которым у него обычно не должно быть доступа. Эти запросы могут включать запросы к внутренним службам, облачным сервисам или даже к тому же веб-приложению, которое их создало. Большинство причин уязвимостей SSRF связаны с потребностями бизнеса. Сервер предоставляет возможность получать данные из других сервисов, и пользователи могут участвовать в доступе к URL-адресу. Однако сервер не выполняет строгую фильтрацию и не ограничивает адрес назначения запроса на доступ. Например, SSRF может использовать-

ся для онлайн-транскодирования, перевода, совместного использования содержимого страницы и других функций[1]. Злоумышленник может использовать эту уязвимость для отправки запросов в непреднамеренное местоположение через серверное приложение. В типичной атаке SSRF злоумышленник также может использовать SSRF для подключения сервера к внутренним службам в инфраструктуре организации. Они также могут принудительно подключать сервер к произвольным внешним системам, предоставляя учетные данные и конфиденциальные данные [2][3].

Важность устранения проблемы подделки запросов на SSRF при разработке защищенного программного обеспечения имеет решающее значение в современном контексте сильно взаимосвязанных веб-приложений, поскольку это необходимо для защиты данных, предотвращения критических атак, поддержания репутации и доверия пользователей, соблюдения нормативных требований, снижения затрат на устранение неполадок и защиты от возникающих угроз. Эффективная кибербезопасность начинается на этапе проектирования и разработки, и SSRF является важнейшим аспектом этой безопасности.

Машинное обучение и обнаружение SSRF

Машинное обучение стало важным инструментом для выявления закономерностей и аномалий в вебзапросах, что позволяет эффективно обнаруживать SSRF. Алгоритмы машинного обучения, обученные на основе исторических данных и поведенческих моделей, могут оценивать входящие запросы и обнаруживать подозрительные отклонения в режиме реального времени. Это особенно ценно при выявлении атак SSRF, которые часто включают нетипичные запросы, направленные на внутренние или внешние ресурсы.

Примеры примененного ML

Регрессия — это метод, используемый для двух теорий. Во-первых, регрессионный анализ обычно используется для прогнозирования и предсказания, в котором его применение во многом совпадает с областью машинного обучения. Во-вторых, регрессионный анализ может использоваться в некоторых случаях для определения причинно-следственных связей между независимыми и зависимыми переменными [4][5]. В этом контексте он используется для прогнозирования непрерывных переменных, таких как вероятность возникновения кибератаки, количество зараженных устройств или время, необходимое системе для восстановления после инцидента, связанного с безопасностью.

Классификация — это процесс распределения данных по заранее определенным категориям или классам на основе их характеристик. В машинном обучении классификация обычно выполняется с использованием модели, которая была обучена на помеченном наборе данных. Модель учится определять закономерности и особенности в данных, которые указывают на каждый класс, а затем может использовать эти закономерности для прогнозирования новых немаркированных данных [6]. В контексте кибербезопасности классификация используется для выявления и прогнозирования наличия угроз и уязвимостей в компьютерных и сетевых системах, для идентификации различных классов сетевых атак, таких как подмена или сканирование. Кроме того, классификация может быть использована для группировки различных пользователей для группового анализа [4].

Кластеризация — это разделение набора данных на несколько групп таким образом, чтобы эти данные в одних и тех же группах были наиболее похожи на другие в той же группе и отличались от точек данных в других группах. Это в основном совокупность объектов на основе сходства и несходства между ними[7]. Она широко применяется в области кибербезопасности для выявления закономерностей и аномалий в данных, что может помочь обнаруживать киберугрозы и предотвращать атаки. Алгоритмы кластеризации в основном используются для поиска шаблонов данных или кластеров данных в среде больших данных, где маркировка данных становится сложной задачей. Одним из недостатков кластеризации по kсреднему значению является определение значения k в начале. Используя вычисления сходства признаков, кластеризация k-средних широко используется в приложениях безопасности [8][9].

Ассоциативные правила являются одним из основных способов представления структурных паттернов, лежащих в основе необработанных данных. Они представляют зависимости между наборами наблюдений, содержащихся в данных. Ассоциации, установленные этими правилами, очень полезны в области медицины, например, в области прогнозирования здоровья [10]. В контексте кибербезопасности правила партнерства могут помочь выявить подозрительные модели поведения и предсказать потенциальные угрозы. Некоторые используемые алгоритмы: Априорный, Эвклазный, FP-Рост.

Уменьшение размерности — это процесс уменьшения размерности объектов, но данные по-прежнему присутствуют. В уменьшенном наборе данных или наборе данных с низкой размерностью важнейшие признаки сохраняются, даже если какой-то отдельный паттерн исчезает [11][12]. Кроме того, он используется для уменьшения размера входных данных и последующего сохранения значительной дисперсии основных характеристик по сравнению с набором данных большего раз-

мера. В реальных данных это станет легко обнаруживать и использовать для приложений интеллектуального анализа данных и обеспечит высокую точность [13][14] [15][16]. Некоторые используемые методы: Анализ главных компонентов (PCA), Разложение по сингулярным значениям (SVD), Встраивание стохастических соседей, распределенных по T (TSNE), Линейный Дискриминантный анализ (LDA), Скрытый семантический анализ (LSA), Факторный анализ (AF), Анализ независимых компонентов (ICA), Факторизация неотрицательных матриц (NMF). В контексте кибербезопасности уменьшение размерности часто используется для анализа больших наборов данных и выявления закономерностей в поведении пользователей и потенциальных киберугроз. Входные данные сводятся к меньшему количеству переменных, что позволяет быстрее и эффективнее обнаруживать угрозы. Например, уменьшение размерности может использоваться для анализа журналов доступа к серверам и выявления закономерностей подозрительной активности.

Порождающие модели помогают нам лучше представлять или моделировать набор данных, генерируя данные в виде цепочек Маркова или просто используя порождающий итеративный процесс для достижения того же результата. Благодаря недавнему нововведению Generative Adversarial Networks (GAN) теперь стало возможным использовать искусственный интеллект для создания произведений искусства, музыки и т. д. с высокой степенью реализма [17]. Существует несколько методов генеративного моделирования, используемых в кибербезопасности, в том числе Конкурирующие генеративные сети (GAN) и Ограниченные Генеративные модели Больцмана (RBM). Эти модели основаны на идентификации шаблонов и характеристик во входных данных и используют эти шаблоны для генерации новых данных.

Интеграция ML-решений

Успешная интеграция ML-решений в цикл разработки программного обеспечения включает в себя несколько ключевых этапов:

Первоначальная оценка риска: определите области приложения, потенциально уязвимые для CSRF, и установите критерии риска.

Выбор ML-решений: Выберите подходящие ML-алгоритмы для обнаружения SSRF и сконфигурируйте модели.

Непрерывное обучение: Снабжайте модели соответствующими данными и постоянно корректируйте их, чтобы они адаптировались к новым угрозам.

Автоматизация процессов: внедрите автоматизацию для обнаружения и оповещения о SSRF в режиме реального времени.

Интеграция со средствами разработки: Интегрируйте решения ML в инструменты, используемые разработчиками, такие как интегрированные среды разработки (IDE) и системы контроля версий.

Сравнение традиционных методов обнаружения и подходов, основанных на ML, для SSRF

Сравнение традиционных методов обнаружения и подходов, основанных на ML, для обнаружения уязвимостей SSRF важно для понимания того, когда и как использовать каждый из них. Ниже приведены преимущества и недостатки каждого подхода, а также примеры инструментов и техник, связанных с каждым из них (см. табл. 1).

Таким образом, обнаружение уязвимостей SSRF с помощью машинного обучения обеспечивает расширенные возможности обнаружения и адаптивность к возникающим угрозам, но также создает проблемы, связанные с получением адекватных обучающих данных и обработкой ложноположительных и отрицательных результатов. Выбор этого подхода будет зависеть от сложности угроз и ресурсов, доступных для обучения и поддержки моделей ML. Во многих случаях комбинированная стратегия, включающая традиционное обнаружение и ML, может обеспечить надежную защиту от SSRF в веб-приложениях.

Технические и этические проблемы

Внедрение решений на основе ML для устранения уязвимостей, связанных с подделкой запросов на SSRF, сопряжено с техническими трудностями, а также этическими и юридическими соображениями. В этом разделе мы кратко описываем основные технические проблемы и рассматриваем этические и юридические аспекты, уделяя особое внимание конфиденциальности данных.

Технические проблемы:

Адаптация к изменениям: модели ML должны адаптироваться к изменениям в веб-трафике и запросах, что требует постоянного обучения и настройки, чтобы оставаться эффективными.

Ложноположительные и отрицательные результаты: Обнаружение SSRF на основе ML может генерировать ложноположительные результаты (неправильные предупреждения) или ложноотрицательные результаты (неспособность обнаружить реальные угрозы). Настройка моделей для уменьшения этих ошибок является постоянной задачей.

Таблица 1.

Сравнение традиционных методов обнаружения и подходов, основанных на ML, для SSRF

Методы	Преимущества	Недостатки
Методы блокировки IP/URL-адресов	• Простота • Эффективность • Эффективность против известных угроз	• Неэффективны против новых угроз • Потенциальные ложноотрицательные результаты • Потребность в постоянном обслуживании
Способ проверки URL-адреса	• Конкретный фокус. • Низкое потребление ресурсов	• Ограничение по комплексному обнаружению • Неэффективное обнаружение маскировки • Неадаптивный
Правила брандмауэра и метод правил прокси-сервера	• Контроль доступа • Предотвращение несанкционированного доступа	• Ограничение сложного обнаружения • Сложная конфигурация • Требуют обновления вручную
Способ ограничения ресурсов	• Предотвращение злоупотребления ресурсами • Защита от атак типа «отказ в обслуживании» (DoS).	• Не обнаруживает SSRF напрямую • Сложная конфигурация:
Ведение журнала	• Идентификация аномального поведения • Журнал действий • Видимость	• Подразделение человеческого анализа • Ложные срабатывания и отрицательные результаты • Отсутствие предотвращения в режиме реального времени
Проверка подлинности билета	• Проактивный подход • Проверка входных данных • Защита от источников • Уменьшение площади атак	• Ограничивается статической проверкой • Сложность реализации • Потенциальное влияние на удобство использования
Обнаружение с помощью машинного обучения	• Обнаружение новых угроз • Адаптивность • Обучаемость • Уменьшение ложноположительных результатов • Долгосрочная эффективность	• Требуются обучающие данные • Потенциальные ложноположительные и отрицательные результаты • Вычислительные ресурсы и время: • Необходимость постоянного обновления. • Сложность внедрения:
Изучение функций	Позволяет автоматически идентифицировать соответствующие шаблоны вредоносного веб-трафика без необходимости получения подробных предварительных знаний. Это может повысить точность обнаружения SSRF.	Требуется надлежащий набор обучающих данных и алгоритмы обучения функциям. Это может увеличить сложность системы обнаружения.
Модели аномального поведения (модели обнаружения аномалий):	Они эффективны при выявлении аномальных моделей веб-трафика, которые могут указывать на атаку SSRF. Они могут быть полезны для обнаружения неизвестных угроз.	Может генерировать ложные срабатывания при неправильной настройке. Им требуется набор исторических данных, чтобы установить нормальное поведение.
Глубокое обучение	Они могут изучать иерархические представления данных, что полезно для обнаружения сложных шаблонов в веб-запросах. Они эффективны при обнаружении сложных угроз.	Они требуют очень больших наборов обучающих данных и значительных вычислительных ресурсов. Установка и точная настройка могут быть непростыми.

Разнообразие запросов: Веб-приложения могут генерировать значительное разнообразие запросов, что затрудняет создание моделей ML, которые являются достаточно универсальными для обнаружения всех вариантов SSRF.

Этические и юридические соображения:

Конфиденциальность данных: Обнаружение SSRF часто включает анализ веб-трафика, который может включать конфиденциальные или личные данные. Важно обеспечить соблюдение законов о конфиденциальности и безопасную обработку конфиденциальных данных.

Прозрачность и объяснимость: Модели ML, особенно модели с глубоким обучением, могут быть сложными и их трудно интерпретировать. Прозрачность в принятии решений по безопасности важна для доверия пользователей и подотчетности.

Соответствие нормативным требованиям: Различные нормативные акты, такие как Общий регламент

Европейского союза по защите данных (GDPR), устанавливают конкретные требования к обработке персональных данных. Организации должны соответствовать этим требованиям при внедрении решений ML.

Ответственность: Юридическая и этическая ответственность в случае инцидентов безопасности, связанных с SSRF, должна быть четко определена. Это включает в себя ответственность разработчиков, владельцев приложений и поставщиков решений для обеспечения безопасности.

Направления на будущее и призыв к действию

В этом разделе мы исследуем возможные направления на будущее и призываем к действию сообщество разработчиков программного обеспечения.

Совершенствование модели ML: Будущие исследования могут быть направлены на разработку более сложных и точных моделей ML для обнаружения SSRF с учетом эволюции киберугроз.

Непрерывное обучение: Внедряйте системы ML с возможностями непрерывного обучения, которые адаптируются к возникающим угрозам и позволяют избежать устаревания.

Интерпретируемость: Работайте над интерпретируемостью моделей ML для понимания и объяснения решений в области безопасности, которые могут иметь решающее значение при аудите и соблюдении нормативных требований.

Продвинутая автоматизация: изучите передовые решения для автоматизации, которые не только обнаруживают SSRF, но и могут автономно принимать корректирующие меры.

Призыв к действию для сообщества разработчиков программного обеспечения:

Обучение и осведомленность: Продвигайте обучение кибербезопасности и ML в сообществе разработчиков программного обеспечения для создания высококвалифицированных команд в области безопасности.

Междисциплинарное сотрудничество: Поощряйте сотрудничество между специалистами по безопасности, разработчиками и экспертами по ML для совместного решения проблем SSRF.

Соответствие нормативным требованиям: убедитесь, что решения ML соответствуют требованиям конфиденциальности и безопасности, и поощряйте соблюдение стандартов и передовой практики.

Обмен опытом: Создание платформ для обмена опытом и извлеченными уроками при внедрении решений ML для обнаружения SSRF.

Выводы

Таким образом, применение машинного обучения для предотвращения уязвимостей SSRF представляет собой значительный прогресс в области кибербезопасности и разработки программного обеспечения. Раннее выявление SSRF с помощью моделей ML дает существенные преимущества с точки зрения безопасности, стоимости и соответствия нормативным требованиям. Поскольку мы вступаем во все более сложный киберпространственный ландшафт, важно решать технические и этические проблемы, связанные с этими решениями.

Сообщество разработчиков программного обеспечения играет решающую роль в содействии внедрению этих технологий и в продолжении исследований и совершенствовании возможностей ML для предотвращения SSRF и других угроз. Благодаря сотрудничеству и осведомленности мы можем еще больше повысить безопасность наших веб-приложений и защитить критически важные данные в постоянно развивающемся цифровом мире.

ЛИТЕРАТУРА

1. Luo H. Ssrf vulnerability attack and prevention based on php // nternational Conference on Communications, Information System and Computer Engineering (CISCE). IEEE. 2019. С. 469–472.
2. Shahriar, H.; Zulkernine M. Information-theoretic detection of SQL injection attacks // In Proceedings of the 2012 IEEE 14th International Symposium on High-Assurance Systems Engineering. 2012. С. 40–47.
3. Altulaihan, E.A.; Alismail, A.; Frikha M. A Survey on Web Application Penetration Testing // Electronics. 2023. Т. 12, № 5. С. 1229.
4. Wu, J., Liu, C., Cui, W., & Zhang Y. Personalized Collaborative Filtering Recommendation Algorithm based on Linear Regression. // 2019 IEEE International Conference on Power Data Science (ICPDS). 2019. С. 139–142.
5. Maulud, D., & Abdulazeez A.M. A review on linear regression comprehensive in machine learning // J. Appl. Sci. Technol. Trends. 2020. Т. 1, № 4. С. 140–147.
6. Shin Kan A. ¿Qué es la clasificación en el contexto del aprendizaje automático y la ciencia de datos? [Электронный ресурс]. 2023. URL: https://prompt.uno/machine-learning/que-es-la-clasificacion-en-el-contexto-del-aprendizaje-automatico-y-la-ciencia-de-datos/.
7. Priy S. Clustering in Machine Learning [Электронный ресурс]. 2023. URL: https://www.geeksforgeeks.org/clustering-in-machine-learning/ (дата обращения: 26.05.2023).

8. Bhuyan M., Bhattacharyya D., JK K. Network anomaly detection: methods, systems and tools. // IEEE Commun Surv Tutor. 2014. 2014. Т. 16. С. 303–336.
9. Dasgupta, D., Akhtar, Z., & Sen S. Machine learning in cybersecurity: a comprehensive survey // J. Def. Model. Simul. 2022. Т. 19, № 1. С. 57–106.
10. Sánchez-de-Madariaga, R., Martinez-Romo, J., Escribano, J.M.C., & Araujo L. Semi-supervised incremental learning with few examples for discovering medical association rules. // BMC Med. Inform. Decis. Mak. 2022. Т. 22, № 1. С. 1–11.
11. Pandian, A.P., Palanisamy, R., & Ntalianis K. Proceeding of the International Conference on Computer Networks, Big Data and IoT (ICCBI-2019). 2020.
12. Mohammed, M.A., Al-Khateeb, B., Rashid, A.N., Ibrahim, D.A., Abd Ghani, M.K., & Mostafa S.A. Neural network and multi-fractal dimension features for breast cancer classification from ultrasound images. // Comput. Electr. Eng. 2018. Т. 70. С. 871–882.
13. N. Sharma and K. Saroha. Study of dimension reduction methodologies in data mining // International Conference on Computing, Communication & Automation IEEE. 2015. С. 133–137.
14. Saini O., S. Sharma. A review on dimension reduction techniques in data mining // Comput. Eng. Intell. Syst. 2018. Т. 9. С. 7–14.
15. Alamilla Hernández L.M. и др. Arquitectura REST para el desarrollo de aplicaciones web empresariales REST architecture for enterprise web application development // Rev. Electrónica sobre Ciencia, Tecnol. y Soc. 2021. Т. 8. С. 15.
16. Zebari, R., Abdulazeez, A., Zeebaree, D., Zebari, D., & Saeed J. A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction // J. Appl. Sci. Technol. Trends. 2020. Т. 1, № 2. С. 56–70.
17. Harshvardhan, G.M., Gourisaria, M.K., Pandey, M., & Rautaray S.S.(. A comprehensive survey and analysis of generative models in machine learning // Comput. Sci. Rev. 2020. Т. 38. С. 100285.

Журнал «Современная наука: актуальные проблемы теории и практики»