DOI 10.37882/2223-2966.2025.09.16

ДИНАМИЧЕСКИЙ МЕТОД ЗАЩИТЫ КЛИЕНТСКОЙ ЧАСТИ ВЕБ-САЙТОВ ОТ НЕПРАВОМЕРНОГО КОПИРОВАНИЯ ДАННЫХ

DYNAMIC METHOD OF PROTECTING THE CLIENT SIDE OF WEBSITES FROM ILLEGAL DATA PARSING

A. Kochenkov

Summary. The article is devoted to the development of a dynamic method for protecting the client side of websites from both automated data collection (parsing) and manual copying. The author analyzed the existing methods of protection, noting their features and disadvantages. Based on the analysis, a unique method has been developed that functionally provides the same capabilities as translating text into a bitmap image, but requires less computing power. It also allows you to load content dynamically, which makes it possible to output information as needed. This method, combined with other solutions, can increase data security against unauthorized copying.

Keywords: information technology, information security, document protection, management solutions, software development optimization.

Коченков Антон Александрович
Аспирант, ФГБОУВО «Российская академия народного хозяйства и государственной службы при Президенте Российской Федерации»
i.anton.kochenkov@ya.ru

Аннотация. Статья посвящена разработке динамического метода защиты клиентской части веб-сайтов как от автоматизированного сбора данных (парсинга), так и ручного копирования. Автор проанализировал существующие методы защиты, отмечая их особенности недостатки. На основе анализа разработан уникальный метод, который функционально дает те же возможности, что и перевод текста в растровое изображение, но требует меньшей вычислительной мощности. А также позволяет загружать контент динамически, что дает возможность вывода информации по мере необходимости. Данный метод в сочетании с другими решениями способен повысить защищенность данных от неправомерного копирования.

Ключевые слова: информационные технологии, информационная безопасность, защита документов, управленческие решения, оптимизация разработки ПО.

Введение

На данный момент существует множество интернет-ресурсов, которые предоставляют информацию в открытом формате для личного пользования, но при этом требуют приобретение лицензии для коммерческого использования. Зачастую злоумышленник использует средства автоматизированного сбора информации для создания собственной информационной базы. Важность защиты данных возрастает в условиях, когда утечка данных не только приводит к финансовым потерям, но и подрывает доверие и может нанести непоправимый ущерб репутации организации[1-3] Автоматизированные средства сбора информации на данный момент способны собирать данные с открытых профилей социальных сетей, а также картографических сервисов, интернет-магазинов, библиотек и т.д. Методы защиты от автоматизированного сбора данных постоянно совершенствуются, что говорит об актуальности данного вопроса.

Материалы и методы

Автором рассмотрены как основные методы защиты от парсинга (автоматизированного сбора), так и ручного копирования.

Парсеры — программы или скрипты, автоматически собирающие веб-контент со страниц сайта [4, 5].

Выделяют два вида парсеров:

1) Считывающие исходный код html-страницы;
2) Использующие web-браузер для имитации поведения.

Наиболее распространенные с точки зрения защиты данных от автоматического анализа можно считать те методы, которые используют широко известные организации, такие как: Meta (запрещена в РФ), В контакте, Яндекс, Google и т.д. В ходе анализа методом серого ящика исходного кода клиентской части веб-сайтов различных организаций выявлено сходство применимых методов защиты.

Стоит выделить следующие методы защиты информации на сайтах, которые чаще всего применяются:

1) Проверка IP адреса клиента
2) Присвоение уникальных значений CSS-классов, для каждого отдельного элемента с выводом по мере прокрутки страницы
3) Отслеживание перемещения мыши и выявление закономерностей в процессе прокрутки страницы
4) Генерация специальных токенов для обмена информацией между клиентской и серверной частью веб-сайта
5) Сохранение документа в виде изображения
6) Использование тестов Тьюринга (CAPTCHA) при выявлении подозрительной активности

Недостатки теста Тьюринга, применяемого на современных интернет-ресурсах, заключаются в том, что его можно обойти методом перенаправления на зараженные ресурсы, где он будет выводиться реальным пользователям, неподозревающим, что он выполняют верификацию для атаки на другой сайт. Также исследователи Поликарпов Е.С., Анисимов С.Л., Толстых А.А. приводят методы, которые основываются на прохождении CAPTCHA для слабовидящих с инструментами распознавания речи [6].

Наиболее актуальным методом для обнаружения автоматизированного сбора данных является изучение закономерностей перемещения курсора мыши и действий пользователя. Многие исследовательские группы сообщают, что поведенческие паттерны, наблюдаемые при использовании мыши или клавиатуры, могут варьироваться от человека к человеку и зависеть от настроения или уровня внимания [7]. Для поиска имитаций поведения пользователей используют алгоритмы с использованием ИИ, которые выявляют математические закономерности при перемещении курсора мыши. Подобный метод широко применяется в заблокированной на территории РФ по решению Роскомнадзора социальной сети Instagram.

Для имитации «человеческой» траектории курсора чаще всего используют кривую Безье:

$$B(t) = (1-t)^2 P_0 + 2t(1-t)^2 P_1 + t^2 P_2,\ t \in [0,1] \quad (1)$$

Процесс выявления попыток автоматизированного сбора может занимать до нескольких минут, поэтому присутствует необходимость ограничить возможность получения полного документа в момент его открытия. При такой реализации должна выводится только та часть страницы, которая находится в активной области.

Стоит отметить, что информация может быть скопирована не автоматически, а зарегистрированным пользователем. Актуальное на данный момент решение подробно описали Савельева М. Г., Урбанович П. П. [8]. Для защиты от копирования документа со стороны пользователя, а также усложнения работы парсеров используется метод перевода страниц в растровое изображение, однако данное решение является трудозатратным для серверной части веб-сайта.

Автором статьи предложено решение, которое функционально схоже с методом перевода текста в растровое изображение, но при этом менее трудозатратно. Принцип данного решения заключается в создании документа, который будет сложно интерпретировать машине, а также исключает возможность несанкционированного копирования пользователем. При этом для пользователя документ остается полностью читабельным. Для реализации защиты документа предложено использовать алгоритм с применением двух ключей, которые генерируются с каждым запросом. Первый ключ представляет собой таблицу сопоставления символов, а второй ключ представляет собой файл шрифта, внутри которого символы расставлены согласно данным таблицы. Таблица сопоставления символов размещается исключительно на серверной части, что не допускает её получение прямым путем. Текст документа остается читабельным только при условии подключенного файла шрифта, а при

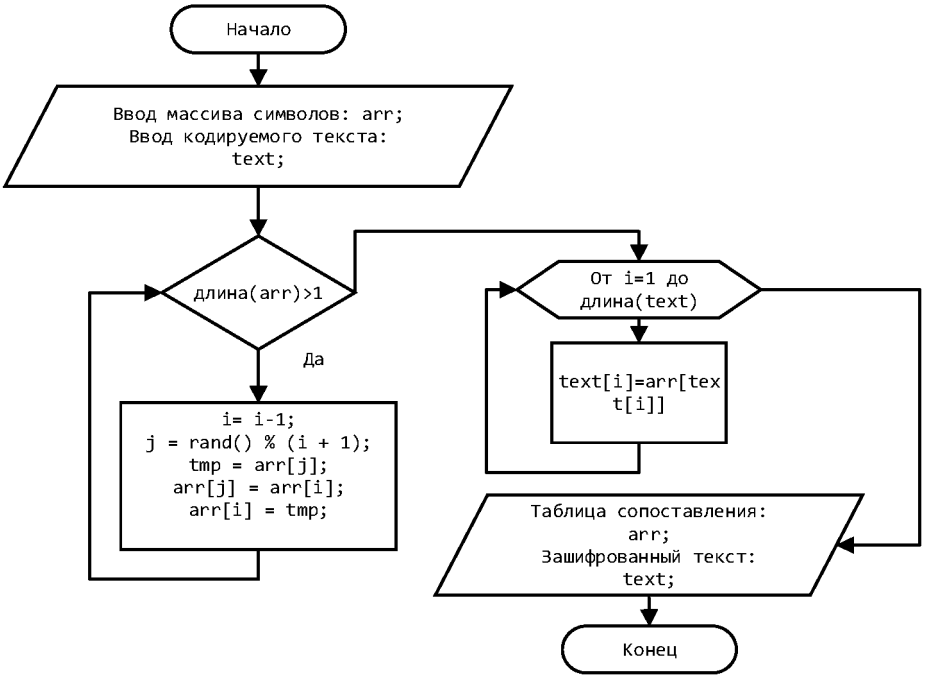

Рис. 1. Упрощенный метод создания защищенного документа

попытке копирования утрачивает данную возможность. Ниже приведена упрощенная блок-схема процесса создания таблицы сопоставления, где для создания таблицы использован алгоритм Фишера-Йетса. В упрощенном варианте представлен на рис. 1.

В качестве примера приведена таблица, содержащая стандартную фразу «Hello World», закодированную предложенным методом (табл. 1).

Таблица 1.

Закодированный текст

Исходный текст	H	E	L	L	O		W	O	R	L	D
Закодированные данные	N	狮	フ	ה	!	/	Б	!	ದ	P	ಲ

Для усложнения получения исходных данных таблицы можно применить символы, которые предназначены для языков, не используемых в документе таким образом, что разные значения внутри закодированного текста, будут означать одну и ту же букву внутри итогового документа. Данный метод схож с предложенным Джозефом Моуборном табличным методом шифрования, что в совокупности с большим количеством значений в таблице дает устойчивость к расшифровке методом Фридриха Касиски [9].

Упрощенная структура документа в виде кода HTML приведена на рис. 2.

```
<style>
  @font-face {
   font-family: Sfont; /* Гарнитура*/
   src: url(fonts/Sfont.ttf); /* Расположение шрифта-ключа */
   }
</style>
<body>
      <p><font face="Sfont">Nxフה!/Б!ದPಲp</font></p>
<body>
```

Рис. 2. Структура документа

Результатом выполнения приведенного кода (при условии подключения ключ-шрифта) будет являться блок текста, содержащий фразу: «Hello World». При попытке копирования текста он будет отображаться как набор символов: «Nxフה!/Б!ದPಲp».

Для поиска методов защиты, схожих по функционалу рассмотрены сайты электронных библиотек, где присутствует ограничение на копирование документов. Наиболее близким методом защиты является решение используемое ЭБС «ЮРАЙТ». Для защиты документа данная библиотека выводит информацию постранично, предварительно преобразуя ее в растровое изображение. Данное изображение можно получить, получить через элемент <canvas>.

Метод используемый ЭБС «ЮРАЙТ» является достаточно эффективным для защиты от неправомерного копирования, со стороны пользователя, а также способствует усложнению парсинга, но имеет ряд существенных недостатков. К недостаткам стоит отнести: большой объем итогового документа, высокая нагрузка на клиентскую и серверную часть веб-приложения. Для сравнения была взята тестовая страница документа из библиотеки «ЮРАЙТ» и переведена в документ с использованием метода, предложенного автором (табл. 2).

Таблица 2.

Сравнение методов

Название метода	Объем исходного документа	Время на вывод документа с учетом кеширования
Перевод текста в изображение	724КБ	0.4 сек
Использование ключ-шрифта	210КБ	0.1 сек

Стоит отметить, что распознавание ключ-шрифта является более трудозатратным процессом, чем распознавание одной страницы текста, поскольку необходимо сопоставить 65536 символов стандарта UCS, что соответствует двадцати страницам формата А4 полностью заполненных текстом 12 pt. Для повышения эффективности рекомендуется использовать шрифты, которые имеют нестандартное начертание символов, а также кешировать файлы, содержащие шрифты и таблицы на серверной части для повышения производительности.

Заключение

Страницы, защищенные методом предложенным автором, не позволяют получить злоумышленнику исходный документ с сохранением форматирования текста и замедляют работу парсеров. Пользователь, не имеющий навыков в разработке специализированного программного обеспечения для парсинга не сможет получить данные из документа. Стоит отметить, что метод, предложенный автором, использует значительно меньше ресурсов серверной части как с точки зрения памяти, так и с точки зрения вычислительной мощности по сравнению со стандартным переводом текста в изображение. Также одним из преимуществ является возможность динамической загрузки текста, поскольку метод не требует загрузки целой страницы с изображением. Данный метод может найти широкое применение в различных сферах, например в электронных библиотеках, где есть ограничение на копирование данных, или для минимизации использования генеративных ИИ при создании создания работ среди студентов. Созданные описанным методом документы содержат маркеры в виде прикрепленного файла шрифта, что позволяет определить первоисточник. При попытке скопировать только текст итоговый документ приобретет нечитаемый формат.

ЛИТЕРАТУРА

1. Edwards, D.J. (2024). Data Protection. In: Critical Security Controls for Effective Cyber Defense. Apress, Berkeley, CA. https://doi.org/10.1007/979-8-8688-0506-6_3
2. Micunocic M., Balkovich, L. Author's rights in the digital age: how Internet and peer-to-peer file sharing technology shape the perception of copyrights and copywrongs // Libellarium Journal for the Research of Writing Books and Cultural Heritage Institutions. 2016. Vol. 8 (2). P. 27–64. DOI: 10.15291/libellarium. v0i0.232. 5. Урбанович П. П.
3. Защита информации методами криптографии, стеганографии и обфускации. Минск: БГТУ, 2016. 220 с
4. Бирюков В.А., Дмитриева О.В., Ливсон М.В. Парсинг аудитории в социальных медиа как инструмент повышения доходов от рекламы электронных средств массовой информации // Известия высших учебных заведений. Проблемы полиграфии и издательского дела. 2021. №2. С. 45–52.
5. Прокопенко В.В. Парсинг как один из инструментов интеллектуальных баз данных // Аллея науки. 2020. №6 (45). Т. 2. С. 68–75.
6. Поликарпов Евгений Сергеевич, Анисимов Сергей Леонидович, Толстых Андрей Андреевич. О защищенности сайта сети интернет от автоматизированного сбора данных // Вестник ВИ МВД России. 2020. №1.
7. A.K. Ghosh, A. Schwartzbard, and M. Schatz. «Learning program behavior profiles for intrusion detection». In Proceedings of the First USENIX Workshop on Intrusion Detection and Network Monitoring, pages 51–62, April 1999.
8. Савельева М.Г., Урбанович П.П. Метод стеганографического преобразования web-документов на основе растровой графики и модели RGB // Труды БГТУ. Сер. 3, Физико-математические науки и информатика. 2022. № 2 (260). С. 99–107. DOI: https://doi.org/10.52065/2520-6141-2022-260-2-99–107.
9. Dooley, J.F. (2023). The Lone Cryptologists: Escape from Riverbank. In: The Gambler and the Scholars. History of Computing. Springer, Cham. https://doi.org/10.1007/978-3-031-28318-5_8

Журнал «Современная наука: актуальные проблемы теории и практики»