УДК 004.056.53

ПРИМЕНЕНИЕ ИНТЕГРИРОВАННОГО ПОКАЗАТЕЛЯ ОТПЕЧАТКОВ БРАУЗЕРА В ЗАДАЧЕ АДАПТИВНОЙ АУТЕНТИФИКАЦИИ СУБЪЕКТОВ ДОСТУПА

Саломатин Александр Александрович
Аспирант,
e-mail: karateka30@mail.ru,
Исхаков Андрей Юнусович
К.т.н., старший научный сотрудник,
e-mail: iskhakovandrey@gmail.com,
Институт проблем управления им. В.А. Трапезникова,
117997, г. Москва, ул. Профсоюзная, 65.

Аннотация. В статье предлагается подход к аутентификации пользователя в веб-пространстве с помощью сравнения интегрированных показателей отпечатков браузера. Интегрированный показатель вычисляется дифференцированно в зависимости от класса защищаемой системы, имеющихся факторов аутентификации и аппаратных характеристик вычислительных систем, используемых субъектами доступа. Также в статье рассмотрены группы идентифицирующих пользователя статичных и поведенческих признаков, приведены параметры, рассчитываемые с помощью JavaScript библиотеки Fingerprint.js. Проведенный эксперимент подтвердил успешное применение предложенного подхода для нескольких наборов отпечатков браузера, полученных экспериментально.

Ключевые слова: технология цифровых отпечатков браузера, адаптивная аутентификация, fingerprinting, субъект доступа.

Цитирование: Саломатин А.А., Исхаков А. Ю. Применение интегрированного показателя отпечатков браузера в задаче адаптивной аутентификации субъектов доступа //Информационные и математические технологии в науке и управлении. 2020. № 4(20). С. 84-92. DOI:10.38028/ESI.2020.20.4.008

Введение. В настоящее время особенно актуальными являются вопросы кибербезопасности, связанные с защитой учетных данных субъектов доступа в критически важных инфраструктурах. Низкий уровень защищенности подсистем аутентификации и авторизации способствует повышению рисков компрометации учетных данных и кражи конфиденциальной информации посредством атак повторного воспроизведения, рефлексии с параллельным протоколом, фишинга и других. Для обеспечения защиты разрабатываются новые методы риск-ориентированной аутентификации с динамической конфигурацией факторов проверки [2], позволяющие эффективно определить, является ли пользователь легитимным.

Одним из подходов к реализации подобных систем является определение уникального многокритериального цифрового следа пользователя, основанного на оценке различных статических и поведенческих признаков профиля субъекта доступа. В рамках построения такого профиля зачастую используются следующие группы признаков: данные о браузере (версия, составляющие user-agent, доступные шрифты, шаблоны заголовков запросов,

спецификации алгоритмов шифрования, использующихся в TLS и др.), данные об операционной системе и аппаратном обеспечении (разрешение и диагональ экрана, версия операционной системы, параметры видеоадаптера, MAC-адрес и др.), данные, относящиеся к запросу (IP-адрес и сопутствующие сведения о провайдере, страна/город, RTT запроса и др.), данные о пользователе (домен электронной почты, оператор мобильной связи, профили в социальных сетях и других веб-приложениях, типичное время активности, другая информация, относящаяся к доменной модели веб-приложения) и т.д. [5]. Деление является условным, поэтому один и тот же поведенческий признак может быть вычислен в двух разных группах. Однако, глубинный анализ совокупности разнородных признаков с применением методов машинного обучения позволяет строить робастные персонифицированные модели субъектов доступа [3].

Очевидно, что определение перечня информативных статических характеристик и поведенческих признаков пользователя, а также алгоритмы определения факторов проверки необходимо адаптивно определять, исходя из специфики защищаемого ресурса и выполняемых операций [6]. В данной работе рассматривается метод определения подлинности пользователя на основе первой группы признаков - данных о браузере (отпечатков браузера).

1. Обзор характеристик браузера для получения отпечатков. Отпечатки браузера представляют собой уникальные значения, отражающие настройки веб-обозревателя субъекта. Ниже представлены примеры собираемых характеристик.

User-Agent – уникальная строка-идентификатор [1, 7]. Ниже приведен пример разбора строки User-Agent: "Mozilla/5.0 (X11; Linux x86_64; en-US; rv:57.0) Gecko/20100101 Firefox/57.0":

“Mozilla/5.0”: оригинальное кодовое имя Navigator;

“X11; Linux x86_64”: операционная система и аппаратная платформа компьютера;

“en-US”: язык локализации;

“rv:57.0”: версия верстки;

“Gecko/20100101”: кодовое наименование программного обеспечения, преобразующего содержимое веб-страниц и информацию о форматировании в интерактивное изображение форматированного содержимого на экране / сборка;

“Firefox/57.0”: имя браузера и версию.

TimeZone – название временной зоны и часовой пояс, определяемый числовым параметром смещения относительно нулевой зоны.

ScreenResolution – разрешение экрана - длина и ширина экрана в пикселях. По данным этой характеристики вычисляется параметр «Диагональ экрана». Стоит отметить, что при переводе пикселей в дюймы часть информации теряется. В связи с тем, что новый идентификатор представляется одним числом, в некоторых случаях такое переопределение может быть эффективным.

Canvas – уникальная строка-идентификатор, которая образуется путем конвертирования градиентного цветного объекта с помощью кодировки base64. Разные вычислители характеризуются различными механизмами обработки изображений, параметров экспорта и уровня сжатия. Эти функции относятся к аппаратным средствам и позволяют проводить идентификацию различных пользователей [10]. Данный признак характеризуется низкой вероятностью повторяемости и коллизий в задаче идентификации.

WebGL – уникальная строка-идентификатор, которую получают конвертацией градиентного объекта с шейдерами с помощью кодировки base64 с учетом всех исключений и возможностей WebGL (например, сглаживания, рендеринга, фильтрации).

DeviceMemory – размер оперативной памяти на устройстве.

HardwareConcurrency – максимально возможно задействованное число потоков на компьютере.

AdBlock – параметр, определяющий факт работоспособности модуля adBlock.

TouchSupport – параметр, определяющий характеристики touchscreen.

Webdriver – параметр, позволяющий определить, управляется ли клиентское приложение, использующее определенный сетевой протокол, автоматически.

Language - язык пользовательского интерфейса браузера.

Colordepth – число бит, определяющих глубину цвета для одного пикселя.

AvailableScreenResolution – доступное разрешение для окна.

SessionStorage – параметр, позволяющий определить возможность сохранения данных в сессионном хранилище.

LocalStorage – параметр, позволяющий определить возможность сохранения данных в локальном хранилище;

IndexedDb – параметр, позволяющий определить возможность сохранения данных в NOSQL хранилище на стороне клиента.

AddBehavior – параметр, позволяющий определить возможность использования поведенческих признаков.

OpenDatabase – параметр, позволяющий определить возможность создания объекта базы данных SQL Lite Database.

CpuClass – класс центрального процессора операционной системы пользователя.

Platform – название платформы браузера, которая представляет прямой способ взаимодействия приложений пользователя с операционной системой Windows.

Plugins – массив сведений о плагинах, установленных в приложении. Плагин представляет собой самостоятельный программный модуль, подключаемый к основной программе и предназначенный для расширения используемых возможностей. Данный параметр включает наименования плагинов, описания плагинов и их типы.

HasLiedLanguages – параметр, проверяющий факт совпадения языка пользовательского интерфейса браузера (отпечаток «Language») с первым языком в списке наиболее предпочитаемых языков пользователя.

HasLiedResolution – параметр, проверяющий факт совпадения разрешение экрана («screenResolution») с доступным разрешением экрана («availablescreenResolution»).

HasLiedOs – параметр, проверяющий факт соответствия между данными об ОС (операционной системе), извлеченными из «userAgent», данными о платформе, классе ЦПУ (central processing unit – центрального процессорного устройства), поддержке touch-screen.

HasLiedBrowser – параметр, проверяющий факт соответствия между данными о браузере, извлеченными из «User-Agent», и данными о том, как браузер справляется с искусственно созданной ошибкой.

Fonts – список шрифтов, доступных браузеру.

Audio – число, отражающее сумму буферных значений. Для получения данного параметра веб-сайт отправляет запрос браузеру на моделирование синусоида, основанного

на результатах аудио-стека устройства. Затем результат отправляется серверу и используется аналогично с энтропией для уникальной идентификации.

2. Метод сравнения отпечатков браузера на основе вычисления интегрированного параметра уникальности пользователя.

В качестве интегрированного показателя уникальности пользователя в предлагаемом методе используется параметр Fingerprinting ID. Схема его получения выглядит следующим образом:

1) получение информации по отпечаткам браузера [8];
2) определение весов отпечатков;
3) объединение полученной информации в интегрированную строку;
4) выполнение процедуры хеширования [9, 10].

Для определения весов используется аналитический-иерархический процесс (АИП), в котором используют три критерия: информативность, скорость получения и сложность получения [11].

АИП генерирует вес для каждого критерия в соответствии с попарными сравнениями критерия, а затем присваивает оценку каждому варианту для получения фиксированного критерия. Далее веса критериев и оценки опций объединяются, чтобы определить глобальные оценки для каждого варианта, являющиеся весами соответствующих объектов. На следующем этапе полученные веса в тех же пропорциях переопределяют таким образом, чтобы их сумма была равна 1.

Представим АИП в данном случае с точки зрения математической формализации. Пусть даны N браузерных отпечатков и 3 критерия. Тогда веса критериев представлены вектором $\lambda = [\lambda_1\ \lambda_2\ \lambda_3]^T; \lambda_1 + \lambda_2 + \lambda_3 = 1$, а оценки опций по критериям представлены матрицей $W = [w_{ij}]$, где $w_{1j} + w_{2j} + \ldots + w_{Nj} = 1;\ 1 \le i \le N, 1 \le j \le 3; i, j \epsilon Z$. Для данной матрицы i - номер отпечатка браузера, j - номер критерия. Также обозначим $I = [w_{i1}], F = [w_{i2}], D = [w_{i3}]$. Для получения весов отпечатков браузера используется формула:

$$W^* = W\lambda \quad (1)$$

Затем, с целью обеспечения различимости отпечатков пользователей используется основной алгоритм подобия. Алгоритм подобия основан на сравнении частей цифрового следа пользователя между собой. Если они совпадают, то присваивается значение 1, иначе – 0. В итоге формируется вектор s из нулей и единиц. Далее вычисляется число подобия

$$P = sW', \quad (2)$$

определяющее вероятность того, что пользователь является легитимным. Здесь W' – вектор-столбец весов браузерных отпечатков при их сравнении.

Заданное число подобия сравнивается с пороговым показателем. В случае не достижения показателя данная операция помечается, как подозрение на инцидент информационной безопасности и выполняется активная реакция системы защиты в соответствии с политиками безопасности: адаптивный подбор дополнительного фактора аутентификации, блокировка субъекта и др.

3. Описание эксперимента. На первом этапе для получения информации о отпечатках браузера была использована библиотека fingerprint2.js. С помощью программного кода на языке программирования JavaScript были получены28 параметров (примеры представлены на рисунке 1).

Рис. 1. Вычисление отпечатков браузера в fingerprint2.js

На следующем этапе необходимо определить веса отпечатков браузера с помощью АИП.

Рассмотрим первый критерий – информативность. Значение браузерного отпечатка по данному критерию будем задавать как число попарных сравнений, в которых данный отпечаток браузера является более информативным. Считаем, что строки-идентификаторы уникальнее других параметров, не являющихся строковыми, а результаты других видов сравнений равноценны. Тогда этот набор определяется массивом I, тогда $I = [0.190\ 0.008\ 0.008 \dots 0.206\ 0.214\ 0.198\ 0.008 \dots 0.008]^T$.

Рассмотрим второй критерий – время получения. Значение отпечатка браузера по данному критерию будем задавать как число попарных сравнений, в которых данный отпечаток браузера был в среднем получен быстрее других. Пусть этот набор определяется массивом F, где время получения отпечатков одинаковое, тогда $F = [0.036\ 0.036\ \dots\ 0.036]^T$.

Рассмотрим третий критерий – сложность получения. Так как отпечатки браузера были получены одним способом комплексно, то зададим сложность получения для каждого отпечатка браузера как равную. Пусть набор определяется массивом D, тогда $D = [0.036\ 0.036\ \dots\ 0.036]^T$.

Предположим, что все три критерия являются равноценными, тогда после перемножения на коэффициенты важности критериев $\lambda = [0.33;\ 0.33;\ 0.33]^T$ можно будет получить показатели, определяющие значимость каждого отпечатка браузера $W^* = [0.08712\ 0.027\ 0.027 \dots 0.092\ 0.095\ 0.089\ 0.027 \dots 0.027]^T$.

С помощью хеш-функции murmur был сформирован показатель fingerprintID (с 28 элементами). Рассмотрим другие условия аутентификации. Допустим, что изменилось значение двух наиболее информативных признаков – canvas и webGL, а остальные значения не подвергались изменению. В результате проведения аналогичных вычислений был получен

показатель fingerprintID, отличающийся по двум параметрам. Проведем сравнение и получим следующие результаты вектора подобия: s = [1,1,...,0,0,1,...1]. Веса для вычисления степени подобия зададим из $W' = W^*$. Вычислим степень подобия:
$P = 1 * 0.027 + 1 * 0.027 + \ldots + 0 * 0.092 + 0 * 0.095 + 1 * 0.089 + 1 * 0.027 + 1 * 0.027 = 1 - 0.092 - 0.095 = 0.81$. Считаем заданным пороговое значение, равное 0,75. Следовательно, пользователь является легитимным с вероятностью 0,81 (81%).

Заключение. В статье предложен подход к аутентификации пользователя в веб-пространстве на основе интегрированного показателя отпечатков браузера.

Первый раздел посвящен описанию технологии отпечатков браузера, рассмотрены группы идентифицирующих пользователя статичных и поведенческих признаков, приведены параметры, рассчитываемые с помощью JavaScript библиотеки fingerprint2.js. Рассмотрены основные типы полученных признаков и приведена интерпретация некоторых значений. Для наиболее информативных признаков были приведены примеры значений и раскрыта практическая значимость оцениваемых параметров. Представлено формализованное описание предложенного метода. Показано, что основой для определения подлинности пользователя является сравнение степени подобия цифровых следов, в качестве которых выступают наборы отпечатков браузера, с заданным пороговым значением. Также описана поэтапная процедура вычисления интегрированных показателей отпечатков для заданных начальных условий. В третьем разделе представлены результаты эксперимента, в котором рассмотренный метод применялся при сравнении двух цифровых следов, один из которых был выявлен практически с помощью программного кода и математических вычислений, а второй – на основе вычислений и предложенных изменений значений нескольких отпечатков браузера. Результат эксперимента подтвердил эффективность предложенного подхода для конкретного практически полученного набора отпечатков браузера.

Исследование выполнено при частичной финансовой поддержке гранта Президента Российской Федерации в рамках научного проекта № МК-2421.2020.9.

СПИСОК ЛИТЕРАТУРЫ

1. Агафонов Ю.М. Деанонимизация пользователей на основе цифровых отпечатков браузера // Безопасность информационного пространства: сборник трудов XVI Всероссийской научно-практической конференции студентов, аспирантов и молодых ученых. Екатеринбург: Уральский федеральный университет имени первого Президента России Б.Н. Ельцина. 2018. С. 3-5

2. Бессонова Е.Е. Метод идентификации пользователей в сети Интернет с использованием компонентного профиля: дис. ... канд. тех. наук: 05.13.19.- Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики. Спб.. 2014. . 115 с.

3. Исхаков А.Ю. Схемы аутентификации пользователя в СКУД с использованием QR кодов и передачи данных по технологии // Информационное противодействие угрозам терроризма. 2014. № 22. С. 11-15.

4. Исхаков А.Ю., Исхакова А.О., Мещеряков Р.В., Бендрау Р., Мелехова О. Использование тепловой карты поведения пользователя в задаче идентификации субъекта инцидента информационной безопасности // Труды СПИИРАН. 2018. № 6 (61). С. 147-171.

5. Платонов Т.С., Оголюк А.А. Межсетевые экраны уровня веб-приложения в современном мире // Программная инженерия и компьютерная техника (майоровские чтения). Спб.. 2019. С. 106-119
6. Решетников С.Ю. Разработка метода идентификации анонимных пользователей сети TOR // Молодежь и современные информационные технологии: сборник трудов XIV Международной научно-практической конференции студентов, аспирантов и молодых ученых. Томск: Национальный исследовательский Томский политехнический университет. 2016. С. 58–60.
7. ElBanna A., Abdelbaki N. Browsers Fingerprinting Motives, Methods, and Countermeasures // 2018 International Conference on Computer, Information and Telecommunication Systems (CITS). Colmar. 2018. Pp. 1-5
8. N. I. Daud, G. R. Haron and S. S. S. Othman, "Adaptive authentication: Implementing random canvas fingerprinting as user attributes factor" 2017 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE). Langkawi. 2017. Pp. 152-156. DOI:10.1109/ISCAIE.2017.8074968.
9. Luangmaneerote S., Zaluska E., Carr L. Inhibiting Browser Fingerprinting and Tracking // IEEE 3rd International Conference on Big Data Security on Cloud. Beijing. 2017. Pp. 63-68
10. Nair K., RoseLalson E. The Unique Id's You Can't Delete: Browser Fingerprints // 2018 International Conference on Emerging Trends and Innovations in Engineering and Technological Research (ICETIETR). Ernakulam. 2018. Pp. 1-5
11. Tracking Your Browser with High-Performance Browser Fingerprint Recognition Model / [Jiang W. and etc.] // China communications. № 17(3). Beijing, 2020. Pp. 168-175

UDK 004.056.53

APPLICATION OF THE INTEGRATED INDICATOR OF BROWSER FINGERPRINTING IN THE PROBLEM OF ADAPTIVE AUTHENTICATION OF ACCESS SUBJECTS

Alexander A. Salomatin
Graduate student,
e-mail: karateka30@mail.ru,
Andrey Yu. Iskhakov
PhD, senior researcher,
e-mail: iay@ipu.ru,
V.A.Trapeznikov Institute of Control Sciences
of Russian Academy of Sciences
117997, Profsoyuznaya Str., 65, Moscow, Russia.

Abstract. This article presents an approach to user authentication in the web space by comparing the integrated metrics of browser fingerprints. The integrated indicator is calculated differentially depending on the class of the protected system, the available

authentication factors and the hardware characteristics of the computing systems used by the access subjects. The experiment carried out confirmed the successful application of the method for a specific practically obtained set of browser fingerprints.

Keywords: digital browser fingerprint technology, fingerprinting, adaptive authentication, access subject

References

1. Agafonov U.M. Deanonimizaciya pol'zovatelej na osnove cifrovyh otpechatkov brauzera [Deanonymization of users based on digital fingerprint] // Bezopasnost' informacionnogo prostranstva: sbornik trudov XVI Vserossijskoj nauchno-prakticheskoj konferencii studentov, aspirantov i molodyh uchenyh = Information Space Security: Collection of Proceedings of the XVI All-Russian Scientific and Practical Conference of Students, Postgraduates and Young Scientists. Ekaterinburg. Ekaterinburg: Ural'skij federal'nyj universitet imeni pervogo Prezidenta Rossii B.N. El'cina = Ural Federal university named after the first President of Russia B.N. Elcin. 2018. Pp. 3-5

2. Bessonova E.E. Metod identifikacii pol'zovatelej v seti Internet s ispol'zovaniem komponentnogo profilya: dis. ... kand. tekh. nauk: 05.13.19 [A method for identifying users on the Internet using a component profile: diss….of the cand. of tech. sc: 05.13.19]. - Sankt-Peterburgskij nacional'nyj issledovatel'skij universitet informacionnyh tekhnologij, mekhaniki i optiki = Saint Petersburg National Research University of Information Technologies, Mechanics and Optics. Spb. 2014. 115 p.

3. Iskhakov A.YU. Skhemy autentifikacii pol'zovatelya v SKUD s ispol'zovaniem QR kodov i peredachi dannyh po tekhnologii NFC [User authentication schemes in ACS using QR codes and data transfer using NFC technology] / A. YU. Iskhakov, R. V. Meshcheryakov. - Tekst: neposredstvennyj // Informacionnoe protivodejstvie ugrozam terrorizma = Information countermeasures against terrorism threats. 2014. № 22. Pp. 11-15.

4. Iskhakov A.YU., Iskhakova A.O., Meshcheryakov R.V., Bendrau R., Melekhova O. Ispol'zovanie teplovoj karty povedeniya pol'zovatelya v zadache identifikacii sub"ekta incidenta informacionnoj bezopasnosti [Using a heatmap of user behavior in the problem of identifying the subject of an information security incident] // Trudy SPIIRAN = Proceedings of the SPIIRAS. 2018. № 6 (61). Pp. 147-171.

5. Platonov T.S., Ogolyuk A.A. Mezhsetevye ekrany urovnya veb-prilozheniya v sovremennom mire [Web Application Layer Firewalls in the Modern World] // Programmnaya inzheneriya i komp'yuternaya tekhnika (majorovskie chteniya): konf = Software engineering and computer technology (major’s readings): conf. Spb. 2019. Pp. 106-119

6. Reshetnikov S.YU. Razrabotka metoda identifikacii anonimnyh pol'zovatelej seti TOR [Development of a method for identifying anonymous users in the TOR network] // Molodezh' i sovremennye informacionnye tekhnologii: sbornik trudov XIV Mezhdunarodnoj nauchno-prakticheskoj konferencii studentov, aspirantov i molodyh uchenyh = Youth and modern information technologies: collection of works of the XIV International scientific-practical conference of students, graduate students and young scientists, Tomsk. - Tomsk: Nacional'nyj issledovatel'skij Tomskij politekhnicheskij universitet = National Research Tomsk Polytechnic University. 2016. Pp. 58–60.

7. ElBanna A., Abdelbaki N. Browsers Fingerprinting Motives, Methods, and Countermeasures // 2018 International Conference on Computer, Information and Telecommunication Systems (CITS). Colmar. 2018. Pp. 1-5

8. N. I. Daud, G. R. Haron and S. S. S. Othman, "Adaptive authentication: Implementing random canvas fingerprinting as user attributes factor," 2017 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE). Langkawi. 2017. Pp. 152-156. DOI:10.1109/ISCAIE.2017.8074968.

9. Luangmaneerote S., Zaluska E., Carr L. Inhibiting Browser Fingerprinting and Tracking // IEEE 3rd International Conference on Big Data Security on Cloud. Beijing, 2017. Pp. 63-68

10. Nair K., RoseLalson E. The Unique Id's You Can't Delete: Browser Fingerprints // 2018 International Conference on Emerging Trends and Innovations in Engineering and Technological Research (ICETIETR). – Ernakulam. 2018. Pp. 1-5

11. Tracking Your Browser with High-Performance Browser Fingerprint Recognition Model / [Jiang W. and etc.] // China communications. - № 17(3). Beijing, 2020. Pp. 168-175

12. This work was partially funded by Russian Federation President Grant for the young scientists [МК-2421.2020.9]