DOI: 10.25728/mlsd.2023.1470

ПОДСИСТЕМА АУТЕНТИФИКАЦИИ И ИДЕНТИФИКАЦИИ СУБЪЕКТОВ ДОСТУПА В АСУ ТП НА ОСНОВЕ БРАУЗЕРНЫХ ОТПЕЧАТКОВ[1]

Мещеряков Р.В., Исхаков А.Ю., Мамченко М.В.
Институт проблем управления им. В.А. Трапезникова РАН, Москва, Россия
mrv@ieee.org, iskhakovandrey@gmail.com, markmamcha@gmail.com

Аннотация. В работе предложены программно-аппаратная архитектура и программное решение для оценки эффективности идентификации пользователей с применением браузерных отпечатков субъектов в АСУ ТП. Максимальная средняя точность идентификации пользователей на основе их интегрального браузерного отпечатка составила 93% (в конце эксперимента).

Ключевые слова: АСУ ТП, пользователь, идентификация, аутентификация, fingerprint.js, браузерный отпечаток, цифровой отпечаток пользователя, информационная безопасность.

1. Введение

В настоящее время актуальным вопросом кибербезопасности и защиты данных является защита учетных данных субъектов доступа, а также их идентификация и аутентификация на веб-ориентированных ресурсах (веб-ресурсах). Одним из перспективных решений данной проблемы является использование риск-ориентированных адаптивных комплексов для аутентификации пользователей с динамически изменяемым перечнем факторов проверки и определения легитимности пользователя, в частности, уникального цифрового отпечатка на основе браузерных атрибутов, в том числе данных о браузере (версия, составляющие компонента user-agent, доступные шрифты, шаблоны заголовков запросов, спецификации алгоритмов шифрования и др.), данные об операционной системе и аппаратном обеспечении (разрешение и диагональ экрана, версия операционной системы, параметры видеоадаптера, MAC-адрес устройства, параметры окна, используемый язык и др.), данные запроса (IP-адрес и сведения о провайдере, страна, город и др.), данные о пользователе и т.д. [1-13].

В целом уникальность подобного цифрового отпечатка достаточно высокая: только 1 из 286777 браузеров будут иметь такой же отпечаток. Точность идентификации пользователя на основе цифровых отпечатков может достигать 99,24% [14]. Кроме того, подобный подход также считается быстродействующим. В частности, при использовании атрибутов UserAgent, TimeZone, ScreenResolution, Canvas и WebGL время, затрачиваемое на идентификацию и аутентификацию пользователя, составляет не более 2 мс [15].

1.1. Обзор существующих исследований по тематике цифровых отпечатков пользователя

В работе [14] рассмотрено три сценария использования библиотеки fingerprint.js [16] для формирования браузерных отпечатков на основе 17 атрибутов: «кросс-браузерность», максимальный объем данных и высокая точность. Дополнительно для каждого сценария проводились эксперименты как с использованием виртуальных частных сетей, так и без них. В результате проведения эксперимента точность правильной идентификации пользователя составила 91,12–95,56%.

В статье [1] предложен подход к аутентификации пользователя на веб-ресурсах на основе интегрированного отпечатка браузера. Показано, что основой для определения подлинности пользователя является сравнение степени подобия цифровых отпечатков, в качестве которых выступают наборы отпечатков браузера, с заданным пороговым значением. Для решения данной задачи использовался подход на основе аналитического иерархического процесса с тремя критериями: информативность, скорость получения и сложность получения. Для формирования перечня из 28 атрибутов отпечатках браузера использовалась библиотека fingerprint2.js. В работе представлены результаты эксперимента, в котором рассмотренный подход применялся при сравнении двух цифровых отпечатков, один из которых был сформирован с помощью программного кода и математических вычислений, а второй – на основе изменений значений отпечатков браузера. Точность определения легитимности пользователя составила 81%. Аналогичный с [1] подход использовался в работе [17] с использованием библиотеки fingerprint3.js для формирования цифрового отпечатка на основе 19 браузерных атрибутов. В работе рассматривались сценарии, предполагающие использование пользователями персональных компьютеров и мобильных устройств.

[1] *Исследование выполнено за счет гранта Российского научного фонда 22-21-00846, https://rscf.ru/project/22-21-00846/.*

В работе [18] проведена серия экспериментов по идентификации пользователей на основе цифрового робототехнического полигона ИПУ РАН. Суть экспериментов заключалась в определении легитимности пользователей на основе стандартных логов (журналов) аудита системы безопасности при входе в систему, а также при расширении признакового пространства за счет использования библиотеки fingerprint.js. Эксперимент проводился на основе набора данных из 11631 записей атрибутов авторизации и браузерных отпечатков 57 уникальных пользователей. Данная серия экспериментов подтвердила эффективность использования библиотеки fingerprint.js: наилучшие значения точности при расширении признакового пространства выросли в среднем с 89% до 91%.

В работе предложено использование [19] авторского программного кода на основе библиотеки fingerprint.js и алгоритмов машинного обучения (наивный байесовский классификатор и метод k-ближайших соседей) для идентификации пользователей на веб-ресурсах на основе данных о 49 атрибутах. Было выявлено 1129 уникальных пользователей, 195 из них заходили на сайт повторно в ходе проведения эксперимента. Точность идентификации за период проведения эксперимента составила 80% (для устройств на базе операционной системы Android) и 96% – для браузеров Windows Chrome.

В работах [20, 21] представлены результаты использования методов машинного обучения для повышения эффективности аутентификации пользователей на основе браузерных отпечатков за счет обнаружения аномалий в поведении пользователей. Обнаружение аномалий осуществлялось на базе набора данных из 48229 записей и 9 стандартных признаках (User, URL, Event, Platform, IP, Browser, Date, Time, and Admin) с использованием трех классификаторов (OneClassSVM, IsolationForest и EllipticEnvelope). При обнаружении подозрительных действий пользователя для дополнительной верификации использовались атрибуты браузерного отпечатка, собираемые библиотекой fingerprint.js. Результаты эксперимента показали, что использование классификатора EllipticEnvelope дает более высокую точность для больших наборов данных (особенно при наличии большого количества записей на одного пользователя), а классификатора IsolationForest – наилучшее значение максимальной средней точности, в том числе при наличии небольшого количества записей на одного пользователя. Средние значения точности идентификации пользователя при использовании различных классификаторов составили: OneClassSVM – 75.3%, IsolationForest – 78.3% и EllipticEnvelope – 76.6%.

1.2. Цель исследования

Целью настоящего исследования является дополнение результатов, полученных в рамках эксплуатации кибернетического робототехнического полигона [18] и в ходе проведения эксперимента по использованию методов машинного обучения для идентификации пользователей с использованием стандартных журналов аудита системы безопасности [20, 21] – за счет оценки точности идентификации пользователей, авторизующихся в системе управления программной платформы диспетчеризации и мониторинга (виртуальная SCADA) в течение продолжительного времени, с использованием интегральных браузерных отпечатков, формируемых библиотекой fingerprint.js.

2. Архитектура предложенного решения

Программно-аппаратная архитектура разрабатываемого решения состоит из трех основных компонентов (рисунок 1):

- веб-ресурса с сервисом авторизации для пользователей на веб-ресурсе и библиотекой fingerprint.js;
- backend-сервера модулем Flask [22] для взаимодействия с сервисом авторизации (через HTTP POST запросы), базой данных SQL [23] и модулем Peewee [24] для записи в базу данных SQL;
- отдельного персонального компьютера для анализа, обработки и визуализации данных (модули, написанные на языке программирования Python с использованием библиотек NumPy [25], Pandas [26], Plotly [27] и MatPlotLib [28]).

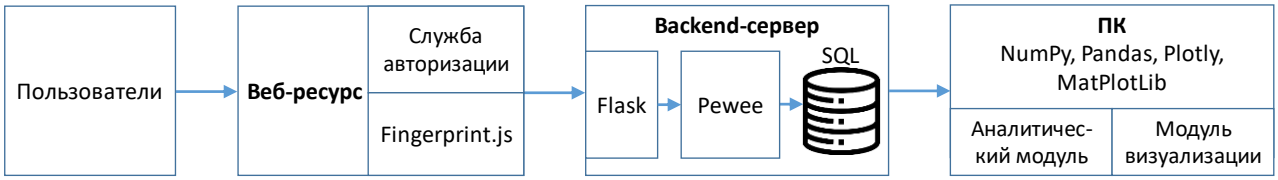

Рис. 1. Схема взаимодействия узлов подсистемы идентификации субъектов доступа

2.1. Структура базы данных

Таблица «Entries» базы данных SQL включает в себя следующие поля:

- Id – уникальный идентификатор пользователя, определенный и передаваемый сервисом авторизации (полагается априори правильным, используется в качестве референсного показателя);
- браузерный цифровой отпечаток пользователя;
- Datetime – дата и время авторизации;
- Role – роль авторизуемого пользователя;
- Data – данные, которые использовались для формирования цифрового отпечатка.
  В свою очередь, поле Data включает в себя следующий набор атрибутов:
- fontPreferences – доступные шрифты;
- timezone – часовой пояс;
- languages – доступные языки;
- screenResolution – разрешение экрана;
- platform – операционная система;
- userAgent – информация о названии браузера и его версии, а также дополнительные данные об операционной системе пользователя;
- cookiesEnabled – логическая переменная, которое показывает, включен ли режим сбора сведений об авторизациях пользователя (так называемые «куки») в браузере;
- plugins – информация о плагинах, установленных в браузере пользователя (название плагина, версия и разработчик);
- osCpu – дополнительная информация об операционной системе и центральном процессоре устройства пользователя;
- deviceMemory – количество оперативной памяти устройства пользователя (в гигабайтах).

3. Описание и результаты эксперимента

В качестве условного объекта защиты выступала программная платформа для создания профессиональных систем автоматизации и мониторинга (рисунок 2). Для защиты интерфейса управления была развернута подсистема идентификации субъектов доступа.

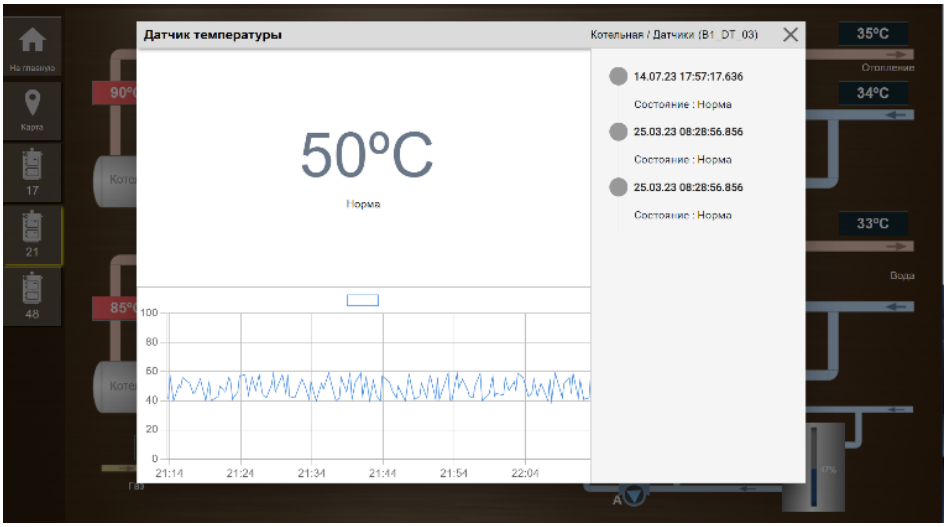

Рис. 2. Виртуальная машина системы диспетчеризации и управления

Сбор данных вёлся в течение 76 дней, за это время было получено 30478 записей об авторизации 1432 уникальных пользователей. Для удобства анализа пользователи были разделены на следующие группы:

- Группа 1: все пользователи, авторизовывавшиеся в системе более одного раза за период сбора данных;
- Группа 2: пользователи, авторизовывавшиеся в системе хотя бы один раз в среднем не реже, чем раз в 5 дней;
- Группа 3: пользователи, авторизовывавшиеся в системе хотя бы один раз в среднем не чаще, чем раз в 5 дней, и не реже, чем раз в 10 дней;

- Группа 4: пользователи, авторизовывавшиеся в системе не чаще, чем раз в 10 дней.

3.1. Анализ активности пользователей (группа 1)

Распределение пользователей на основе их активности за период сбора данных представлено на рисунке 3. Из графика видно, что большинство пользователей проявляют довольно низкую активность, при этом пик количества пользователей приходится на 2 активных дня (192 пользователя).

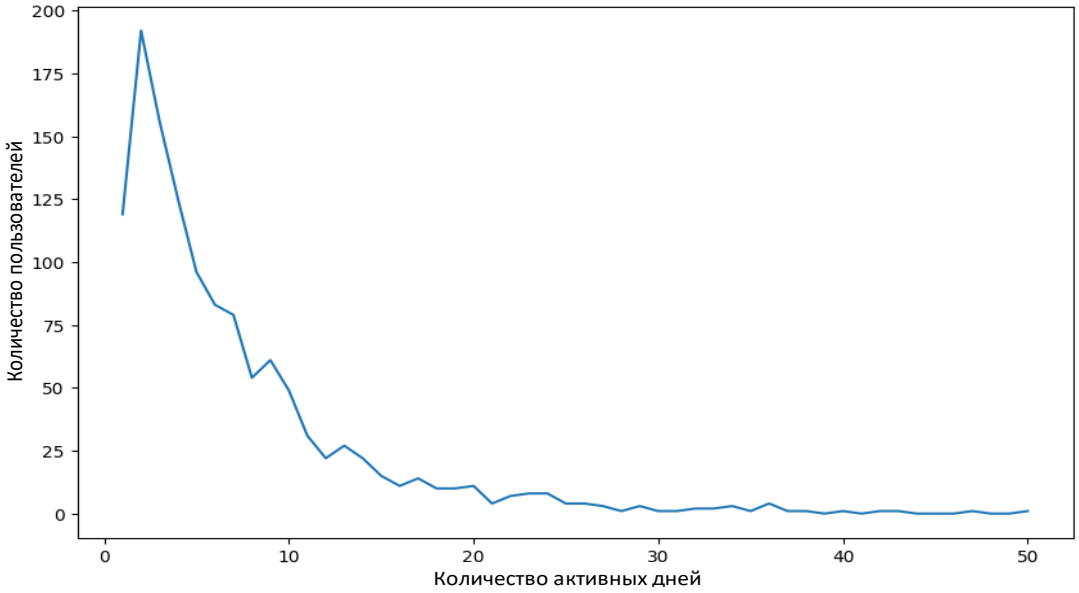

Рис. 3. Распределение пользователей по активности за период сбора данных

3.2. Анализ изменений количества отпечатков пользователя (группа 1)

В среднем на каждого пользователя данной группы приходится 2,27 уникальных отпечатков. На графике (Рисунок 4) точками обозначены средние показатели количества отпечатков для пользователей с соответствующим количеством авторизаций. Размер точек показывает количество пользователей, т.е. чем больше точка, тем больше пользователей имеют количество соответствующих авторизаций. На графике наблюдается явная тенденция к уменьшению скорости роста количества цифровых отпечатков при увеличении количества авторизаций пользователя: при достаточно большом числе авторизаций выбранного пользователя разработанное программное решение может с большой вероятностью идентифицировать его при следующей авторизации на основе собранных о нем данных, и новый цифровой отпечаток для этого пользователя формироваться не будет.

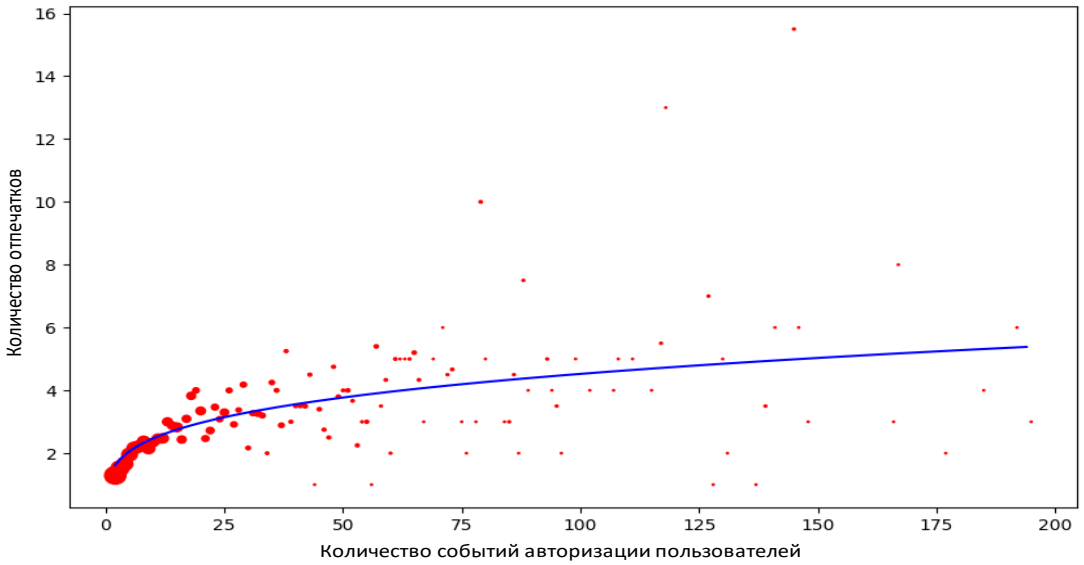

Рис. 4. Зависимость количества цифровых отпечатков от количества авторизаций пользователей

Определим срок «жизни» отдельного цифрового отпечатка пользователя, для этого построим график зависимости количества отпечатков, существовавших N дней (рисунок 5). Временем существования отпечатка будем считать количество дней, прошедшее с первого появления отпечатка в базе данных до момента последней авторизации пользователя с этим отпечатком.

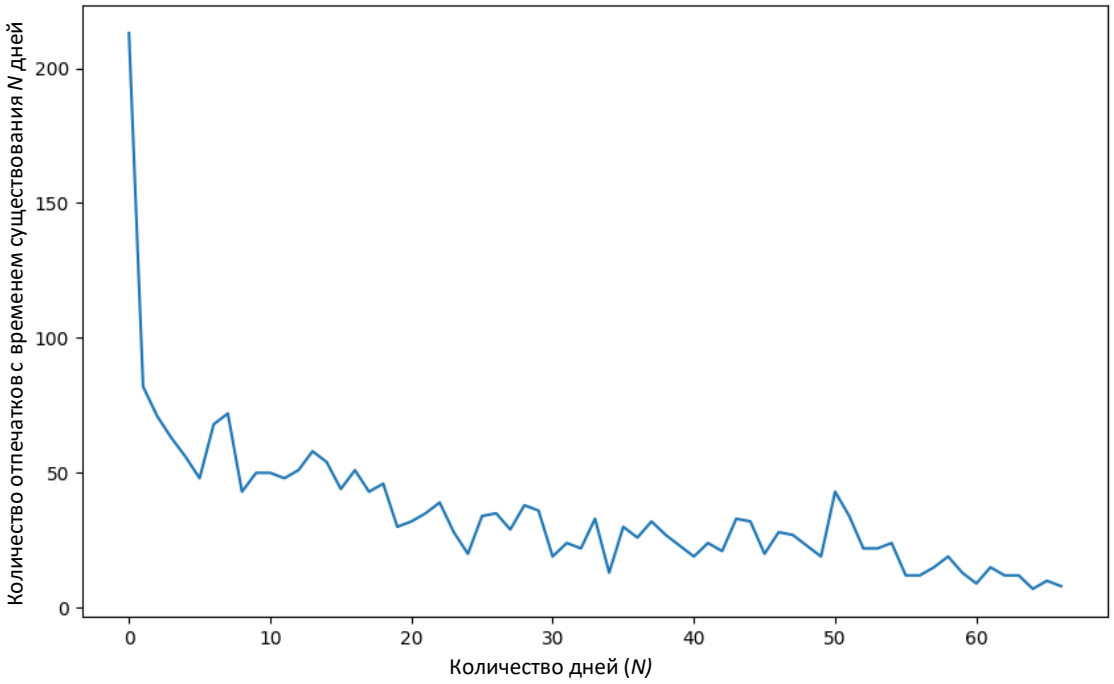

Рис. 5. Зависимость количества отпечатков от продолжительности их существования

Из графика видно, что отпечатки пользователей меняются очень часто. Большое количество отпечатков со сроком жизни в 1 день обусловлено большим количеством пользователей, производящих несколько авторизаций в течении одного дня с нескольких устройств, что приводит к формированию сразу нескольких отпечатков, после чего данные пользователи перестают проявлять активность на длительный период времени, увеличивая вероятность создания для них новых отпечатков при следующей авторизации из-за изменения некоторых браузерных атрибутов (шрифты, аппаратные параметры устройства и т.д.). Среднее время «жизни» одного отпечатка составляет 22,57 дней, что можно полагать удовлетворительным показателем.

3.3. Анализ платформ пользователей (группа 1)

Более половины пользователей (55,28%) используют только одну платформу, 38,8% – две платформы, 5,44% – три платформы, 0,48% – четыре платформы. Очевидно резкое снижение количества пользователей, использующих более двух платформ, что обусловлено необходимостью наличия у пользователя нескольких персональных компьютеров/устройств с различными операционными системами для авторизации. В среднем пользователи авторизуются с использованием более одной платформы (1,51).

3.4. Анализ точности распознавания новых авторизаций (группа 1)

Рассмотрим точность распознавания пользователя разработанным программным решением. За ошибку будем считать генерацию нового цифрового отпечатка при авторизации пользователя, уже находящегося в базе данных в момент этой авторизации. Авторизацию пользователя на веб-ресурсе с созданием его первого цифрового отпечатка, не будем считаться ни ошибкой, ни успешной идентификацией. На рисунке 6 наблюдается небольшой рост точности идентификации с увеличением общего количества авторизаций, что можно объяснить тем, что для каждого пользователя в течение некоторого периода времени создаются цифровые отпечатки для каждого из его устройств (или при типовых изменениях значений атрибутов данных устройств). В связи с этим к концу эксперимента для каждого пользователя созданы практически все возможные отпечатки, что повышает вероятность правильной идентификации пользователей при последующих авторизациях.

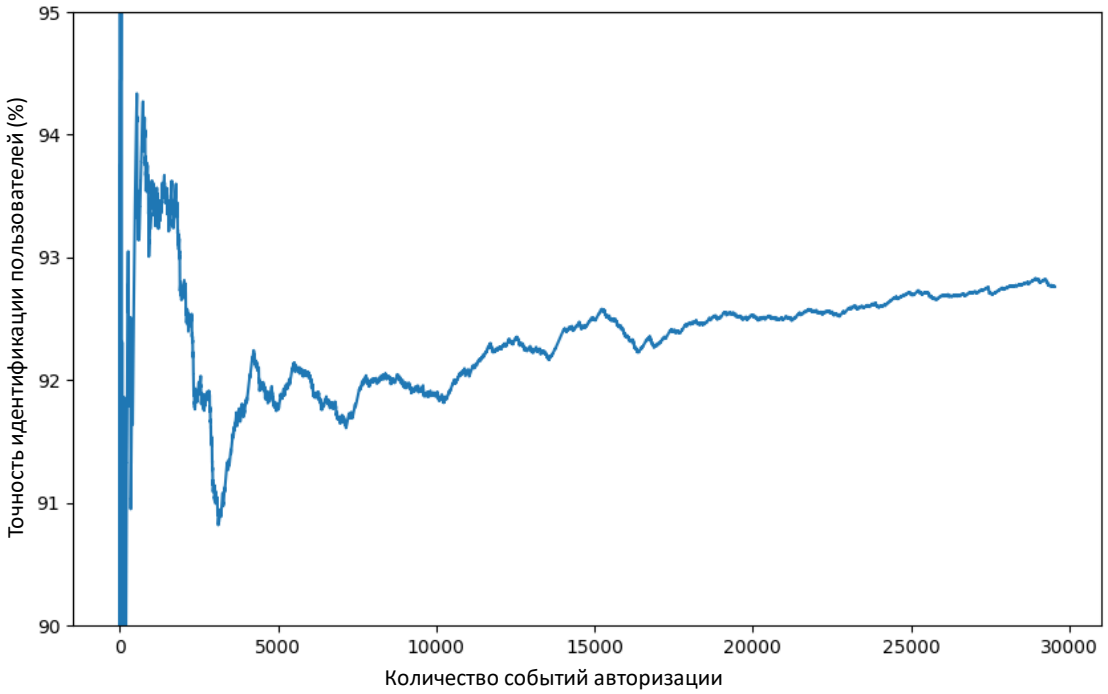

Рис. 6. Зависимость точности идентификации пользователя от общего количества авторизаций

Построим диаграммы всех изменений собираемых значений браузерных атрибутов, в том числе приведших к изменению цифровых отпечатков пользователей (рисунок 7). Существуют очень часто изменяемые параметры пользователей, например, fonts, fontPreferences, canvas и др. На изменение отпечатка влияет не изменение какого-либо одного или нескольких атрибутов, но практически всех параметров одновременно. Однако, некоторые параметры, которые менялись довольно часто, приводили к изменениям отпечатка реже, чем другие. В частности, параметры, изображенные на левой диаграмме (рисунок 7а), влияли на изменение цифрового отпечатка гораздо реже относительно суммарного количества изменений данных атрибутов по сравнению с параметрами, изображенные на правой диаграмме (рисунок 7б). Таким образом, изменение атрибутов screenResolution, platform, hardwareConcurrency, deviceMemory и vendor значительно повышают вероятность формирования нового отпечатка и неверной идентификации пользователя. Данные атрибуты связаны с аппаратной составляющей устройства пользователя и, очевидно, что их изменение будет происходить значительно реже, чем других (программных) атрибутов.

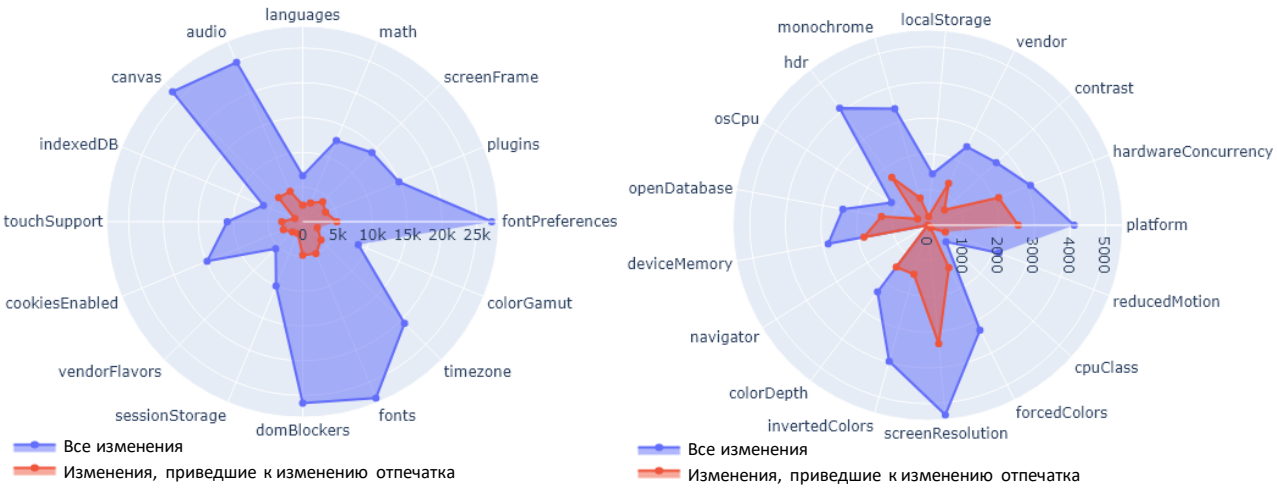

Рис. 7. Суммарные изменения атрибутов пользователя между событиями авторизации (а) и изменения, повлекшие создание нового цифрового отпечатка (б)

3.5. Анализ активности, количества платформ и отпечатков для групп 2-4

Проведем сравнение результатов, полученных для групп пользователей 2-4. Количество авторизаций, пользователей, цифровых отпечатков пользователя, а также среднее количество отпечатков на одну платформу для данных групп представлены в таблице 1.

Таблица 1. Сравнение численных показателей для групп пользователей 2-4

Параметр	**Значение**		
	Группа 2	**Группа 3**	**Группа 4**
Количество авторизаций	14638	8162	6733
Количество пользователей	156	323	771
Среднее количество цифровых отпечатков пользователя	4,2	3,15	1,74
Среднее количество цифровых отпечатков на одну платформу	2,45	1,96	1,43

Из таблицы видно, что пользователи, совершающие авторизации в среднем не реже, чем раз в 5 дней, чаще обладают двумя различными устройствами; пользователи, авторизовывавшиеся в среднем раз в 5–10 дней, имеют 1-2 устройства, а с увеличением интервала входов до 10 дней и более все больше пользователей авторизуются с использованием только одного устройства.

Сравним точность идентификации пользователей в группах 2-4 в зависимости от общего количества авторизаций пользователей (рисунок 8).

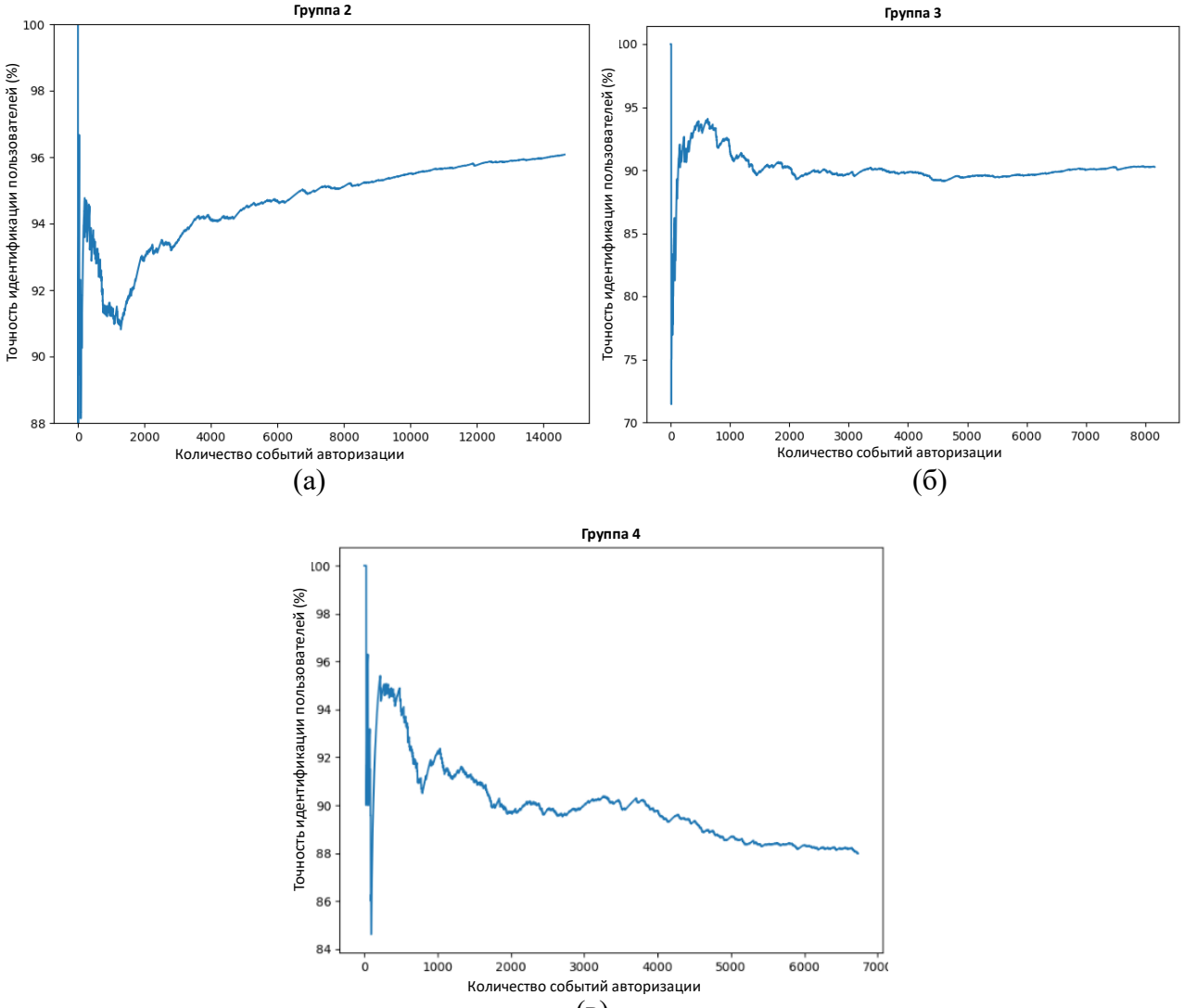

Рис. 8. Зависимость точности идентификации пользователя от общего количества авторизаций в группах 2 (а), 3 (б) и 4 (в)

Несмотря на большее количество отпечатков на пользователей группы 2, процент идентификации с созданием соответствующего отпечатка в этой группе является наибольшим и растет с увеличением

количества авторизаций (до ~96%). В группе 3 этот показатель достигает ~90%, а в группе 4 – снижается до ~88%. Снижение точности идентификации объясняется тем, что с увеличением общего числа авторизаций увеличивается и количество повторных авторизаций для давно не авторизовывавшихся пользователей. У таких пользователей за длительное время с большей вероятностью происходят значительные изменения атрибутов браузера и аппаратных платформ, что приводит к формированию нового отпечатка и снижает точность идентификации.

4. Заключение

В работе предложена и разработана программно-аппаратная архитектура для анализа эффективность идентификации пользователей с использованием их браузерных отпечатков, включающая backend-сервер с базой данной SQL и программными решениями для анализа и визуализации собираемых данных. Максимальная средняя точность идентификации пользователей на основе их интегрального браузерного отпечатка возрастает при увеличении общего количества авторизаций и составляет 92-93% (в конце эксперимента), при этом новые отпечатки практически перестают генерироваться.

Результаты эксперимента позволяют сделать вывод о том, что применение фреймворков и модулей типа fingerprint.js для идентификации пользователей является допустимым. Однако, учитывая специфику систем защиты АСУ ТП, где на первый план выходит задача обеспечения целостности и доступности данных, реализация таких подсистем на реальных объектах возможна исключительно в качестве дополнительного фактора проверки, ограничивая их функционал сигнализацией об аномалиях администраторам безопасности. Основной недостаток используемой программно-аппаратной архитектуры и программного решения для идентификации пользователей по браузерным отпечаткам схожа с другими аналогичными решениями: идентификация пользователя, использующего новое устройство, практически невозможна, несмотря на малый процент подобных авторизаций в проведенном эксперименте от их общего числа.

Для повышения эффективности предложенного программного решения в дальнейшем планируется внедрение методов и алгоритмов машинного обучения с целью повышения точности идентификации пользователей.

Литература

1. *Salomatin A.A., Iskhakov A.Yu.* Application of the integrated indicator of browser fingerprinting in the problem of adaptive authentication of access subjects // Information and mathematical technologies in science and management. – 2020. – Vol. 4, N 20. – P. 84–92.
2. *Fietkau J., Thimmaraju K., Kybranz F., Neef S., Seifert J.-P.* The Elephant in the Background: A Quantitative Approachto Empower Users Against Web Browser Fingerprinting // Proceedings of the 20th Workshop on Workshop on Privacy in the Electronic Society (WPES '21). Association for Computing Machinery. – 2021. – P. 167–180.
3. *Jiang W., Wang X., Song X., Liu Q., Liu X.* Tracking your browser with high-performance browser fingerprint recognition model // China Communications. – 2020. – Vol. 17, N 3. – P. 168–175.
4. *Nair K.V., 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). – 2018. – P. 1–5.
5. *Vastel A., Rudametkin W., Rouvoy R.* FP -TESTER : Automated Testing of Browser Fingerprint Resilience // 2018 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW). – 2018. – P. 103–107.
6. *Cheng D.* Using Function Call Sequence for Browser Fingerprinting Detection // 2022 3rd International Conference on Computer Science and Management Technology (ICCSMT). – 2022. – P. 104–109.
7. *Antonio E., Fajardo A., Medina R.* Tracking Browser Fingerprint using Rule Based Algorithm // 2020 16th IEEE International Colloquium on Signal Processing & Its Applications (CSPA). – 2020. – P. 225–229.
8. *ElBanna A., Abdelbaki N.* Browsers Fingerprinting Motives, Methods, and Countermeasures // 2018 International Conference on Computer, Information and Telecommunication Systems (CITS). – 2018. – P. 1–5.
9. *Zou F., Zhai H.* Browser Fingerprinting Identification Using Incremental Clustering Algorithm Based on Autoencoder // 2021 IEEE 23rd Int Conf on High Performance Computing & Communications; 7th Int Conf on Data Science & Systems; 19th Int Conf on Smart City; 7th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys). – 2021. – P. 525–532.
10. *Iqbal U., Englehardt S., Shafiq Z.* Fingerprinting the Fingerprinters: Learning to Detect Browser Fingerprinting Behaviors // 2021 IEEE Symposium on Security and Privacy (SP). – 2021. – P. 1143–1161.
11. *Sjösten A., Hedin D., Sabelfeld A.* EssentialFP: Exposing the Essence of Browser Fingerprinting // 2021 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW). – 2021. – P. 32–48.
12. *Wu T., Song Y., Zhang F., Gao S., Chen B.* My Site Knows Where You Are: A Novel Browser Fingerprint to Track User Position // ICC 2021 - IEEE International Conference on Communications. – 2021. – P. 1–6.

13. *ElBanna A., Abdelbaki N.* NONYM!ZER: Mitigation Framework for Browser Fingerprinting // 2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C). – 2019. – P. 158–163.
14. *Karpukhin E., Sharmaev V., Propp A.* De-anonymization of the user of web resource with browser fingerprint technology // Journal of Theoretical and Applied Information Technology. – 2022. – Vol. 100, N 14. – P. 5401–5408.
15. *Iskhakov A.Y., Salomatin A.A.* Estimation of the time for calculating the attributes of browser fingerprints in the user authentication task // Topical Problems of Agriculture, Civil and Environmental Engineering (TPACEE 2020). E3S Web of Conferences. – 2020. – Vol. 224, N 01030. – P. 1–8.
16. FingerprintJS documentation [Электронный ресурс]. URL: https://dev.fingerprint.com/docs (дата обращения: 20.07.2023).
17. *Salomatin A.A., Iskhakov A.Y., Meshcheryakov R.V.* Application of the User's Digital Footprint in the Adaptive Authentication Problem // 2021 International Siberian Conference on Control and Communications (SIBCON). – 2021. – P. 1–5.
18. *Iskhakov A.Y., Iskhakova A.O., Meshcheryakov R.V.* Algorithm for building a cyberphysical system operator profile for adaptive authentication // IFAC-PapersOnLine. – 2021. – Vol. 54, N 13. – P. 493–498.
19. *Flood E.* Browser Fingerprinting. Master of Science Thesis in the Program Computer Science: Algorithms, Languages and Logic / E. Flood, J. Karlsson. – Gothenburg: Chalmers University of Technology, 2012. – 99 c.
20. *Iskhakov A.Y., Mamchenko M.V., Khripunov S.P.* Enhanced User Authentication Algorithm Based on Behavioral Analytics in Web-Based Cyberphysical Systems // 2023 International Russian Smart Industry Conference (SmartIndustryCon). – 2023. – P. 253–258.
21. *Iskhakov A.Y., Khazanova Y.Y., Mamchenko M.V., Meshcheryakov R.V., Iskhakova A.O., Khripunov S.P.* Adaptive Authentication System Based on Unsupervised Learning for Web-Oriented Platforms // Mobile Computing and Sustainable Informatics. Lecture Notes on Data Engineering and Communications Technologies. – 2023. – Vol. 166. – P. 507–522.
22. Flask documentation [Электронный ресурс]. URL: https://flask.palletsprojects.com/en/2.3.x/ (дата обращения: 20.07.2023).
23. MySql documentation [Электронный ресурс]. URL: https://dev.fingerprint.com/docs (дата обращения: 20.07.2023).
24. Peewee documentation [Электронный ресурс]. URL: https://docs.peewee-orm.com/en/latest/ (дата обращения: 20.07.2023).
25. NumPy documentation [Электронный ресурс]. URL: https://dev.fingerprint.com/docs (дата обращения: 20.07.2023).
26. Pandas documentation [Электронный ресурс]. URL: https://pandas.pydata.org/docs/ (дата обращения: 20.07.2023).
27. Plotly documentation [Электронный ресурс]. URL: https://plotly.com/python/reference/ (дата обращения: 20.07.2023).
28. Matplotlib documentation [Электронный ресурс]. URL: https://matplotlib.org/stable/index.html (дата обращения: 20.07.2023).