Санкт-Петербургский государственный университет

***МАТЮШИН* Юрий Сергеевич**

Выпускная квалификационная работа на тему

Исследование методов аутентификации в распределённой среде

Уровень образования: магистратура

Направление 02.04.02 «Фундаментальная информатика и информационные технологии»

Основная образовательная программа ВМ.5827.2020 «Распределенные вычислительные технологии»

Научный руководитель:
доцент, кафедра компьютерного моделирования и многопроцессорных систем, к.ф.-м.н.
Корхов Владимир Владиславович

Санкт-Петербург
2022 г.

Saint Petersburg State University

Iurii Sergeevich *MATIUSHIN*

Master's thesis

A Study of Authentication Methods in A Distributed Environment

Level of education: master's degree

Field of study 02.04.02

"Fundamental Informatics and Information Technologies"

Main educational program BM.5827.2020

"Distributed Computational Technologies"

Supervisor:

Associate Professor, Department of Computer Modelling and Multiprocessor Systems

Vladimir Vladislavovich Korkhov, PhD

Saint Petersburg

2022

Оглавление

Введение

В настоящее время проблема идентификации и аутентификации в сети Интернет является актуальной, как никогда раньше. С одной стороны, существует множество Интернет-сервисов – от социальных сетей до банковских сервисов и сайтов государственных организаций – ведущих учёт пользователей и разграничивающих их права доступа к тем или иным ресурсам. Таким сервисам необходимо знать, какой именно пользователь пытается получить доступ к ним и является ли этот пользователь тем, за кого он себя выдаёт.

С другой стороны, в последнее время значительно участились атаки злоумышленников на веб-сервисы. Цели данных атак могут быть различными – от кражи персональных данных до попыток шантажа и вымогательства. Однако во всех случаях слабым местом систем, подвергающихся атакам, является именно система аутентификации. По оценке компании Verizon, ежегодно собирающей статистику по утечкам данных, до 80% успешных хакерских атак (в том числе атаки на крупнейшие сервисы с миллионами пользователей) удались именно из-за слабости системы парольной защиты [1, 2].

Именно поэтому такую важность приобрела задача разработки и применения более надёжных систем аутентификации пользователей. Современные системы аутентификации основываются на различных факторах – от знания секретной информации до биологических свойств организма человека. Более того, многие системы не ограничиваются одним фактором, а используют сразу несколько. Такой подход значительно повышает безопасность системы и снижает вероятность несанкционированного доступа.

В настоящий момент существует множество систем аутентификации, каждая из которых имеет как свои достоинства, так и свои недостатки.

Возникает вопрос выбора наилучшей системы аутентификации для того или иного случая, для чего требуется определить, по каким критериям нужно проводить сравнение и как сравнивать между собой системы, основанные на совершенно разных принципах и факторах.

Особый интерес предоставляют децентрализованные системы аутентификации. Они тесно связаны с понятием суверенной идентичности – идеи о том, что пользователь должен сам контролировать свои личные данные, а не уступать это право другим сторонам. Зачастую для реализации децентрализованных систем аутентификации используется технология блокчейн.

В данной работе рассмотрены методы, применяемые для аутентификации пользователей в современных распределённых системах. Предложен фреймворк для сравнения методов аутентификации, основанных на различных принципах, и проведено сравнение наиболее распространённых методов по нескольким критериям. На практике реализованы системы аутентификации пользователей, в том числе система децентрализованной аутентификации.

Постановка задачи

Целью данной работы является исследование методов аутентификации пользователей, используемых в распределённых системах.

В данной работе были поставлены следующие основные задачи:

- Исследование проблемы аутентификации пользователей в распределённых вычислительных системах.
- Сравнение различных методов, применяемых для аутентификации пользователей.
- Определение достоинств и недостатков рассмотренных методов в различных условиях.
- Практическая реализация системы аутентификации пользователей.

В настоящий момент распределённые системы становятся всё более сложными, а количество их пользователей значительно увеличивается. Существует необходимость в простых и надёжных системах, обеспечивающих аутентификацию пользователей.

Кроме того, участились атаки на распределённые системы. Поэтому к современным распределённым системам предъявляются высокие требования безопасности, которым «традиционные» механизмы аутентификации зачастую не соответствуют.

Наконец, ещё одним важным аспектом является обеспечение удобства использования распределённых систем. Это подразумевает в том числе и наличие удобных в применении систем аутентификации, интуитивно понятных для рядового пользователя.

Всё это делает проблему аутентификации в распределённых системах актуальной темой исследования.

Обзор литературы

Анализу и исследованию методов аутентификации в распределённой среде посвящено достаточное количество работ.

Так, в статье К. Хелкала и Э. Снеккенес «A Method for Ranking Authentication Products» [18] предложен метод сравнения механизмов аутентификации, основанный на критериях совместимости со сценарием использования, безопасности, удобства использования и стоимости. Предполагается последовательно применять данные критерии к рассматриваемым методам, исключая из рассмотрения методы, которые не соответствуют требованиям, выдвигаемым в рамках того или иного критерия. Такой метод во многом схож с методом, применённым в рамках данной работы, хотя есть и отличия – так, в рамках фреймворка UDS (см. ниже) методы ранжируются, но не исключаются, а также введён критерий развёртываемости, включающий в себя как стоимость реализации метода, так и ряд других аспектов.

В статье С. Пуркаяшта, Ш. Гойял, Б. Олуваладе и др. «Usability and Security of Different Authentication Methods for an Electronic Health Records System» [20] рассмотрен пример сравнения систем аутентификации в приложении к конкретному сценарию (системы записей данных о здоровье пациентов). Данная статья рассматривает всего два критерия – безопасность и удобство использования – а основным методом исследования выступает опрос пользователей системы. Рассмотрены аутентификация с использованием логина и пароля, биометрическая аутентификация, а также система единого входа (Single sign-on). Как и в приведённой статье, в рамках данной ВКР также рассматривается система записей о пациентах как один из примеров системы, в которой необходима аутентификация пользователей.

В статье И. Г. Сидоркиной, Р. В. Канаева и О. Ю. Меркушева «Классификация методов аутентификации человека» [6] рассмотрен целый ряд методов аутентификации. В частности, подробно рассмотрены статические и динамические биометрические методы – материалы статьи использованы для написания соответствующего подраздела данной работы. Однако методика сравнения достаточно схематична и основывается только на двух показателях – производительности и стоимости – по каждому из которых авторы присваивают методам буквенную оценку.

В статьях И. Веласкеса, А. Каро и А. Родригеса «Identifying Comparison and Selection Criteria for Authentication Schemes and Methods» [15] и «Multifactor Authentication Methods: A Framework for Their Comparison and Selection» [16] проведены исследования, в ходе которых из множества критерий, применяемых для сравнения методов аутентификации, выделены наиболее важные. Первая статья рассматривает частоту применения критериев в научных статьях, посвящённых проблемам аутентификации пользователей и сравнения различных методов аутентификации. Во второй статье также представлен результат опроса представителей компаний-клиентов – таким образом определяются критерии, которыми руководствуются пользователи систем аутентификации. В частности, в обеих статьях выделены такие критерии, как безопасность, удобство использования, развёртываемость (включающая в себя стоимость имплементации), а также контекст приложения и требования клиента. Данные исследований, представленные в статьях Веласкеса, Каро и Родригеса, использованы в работе для определения критериев аутентификации, по которым будет проводиться сравнение различных методов.

В статье «The Quest to Replace Passwords: A Framework for Comparative Evaluation of Web Authentication Schemes» [17] авторов Дж. Бонно, Ф.Стаджано, П. ван Оршота и К. Хирли рассмотрен метод сравнения различных механизмов аутентификации по ряду критериев, используемый в

данной работе. Выделены три основных критерия – безопасность, удобство использования и развёртываемость – каждый из которых подробно рассмотрен. В каждом критерии выделен ряд аспектов, по наличию или отсутствию которых в той или иной системе можно проводить сравнительный анализ. Также предложен вариант сравнения с использованием весов, зависящих от важности того или иного аспекта в приложении к сценарию аутентификации – таким образом, при сравнении можно также учитывать особенности конкретной системы. Данная статья является одной из наиболее часто цитируемых по теме анализа и сравнения методов аутентификации. Фреймворк UDS, предложенный в ней, использован в данной работе как основной метод сравнения систем аутентификации.

В части работы, связанной с применением децентрализованных идентификаторов, использована рекомендация W3C «Decentralized Identifiers (DIDs) v1.0» [25], в которой подробно рассмотрены понятие децентрализованного идентификатора, основные свойства и методы, связанные с DID, вопросы, связанные с безопасностью DID. В статье М. Сабаделло, К. Ден Хартога, К. Лундквиста и др. «Introduction to DID Auth» [26] рассмотрено использование DID для аутентификации пользователей. В ней перечислены принципы аутентификации с помощью DID, варианты взаимодействия между сторонами аутентификации, возможные варианты архитектуры системы аутентификации.

Также использованы «белые книги» компаний Sovrin [23] и Evernym [24]. Обе эти компании активно занимаются разработкой систем, использующих децентрализованные идентификаторы, и их работы внесли значительный вклад в развитие идеи суверенной идентичности.

Наконец, были использованы материалы статьи А. Мартини «One-click Login with Blockchain» [31]. В ней предлагается вариант практической реализации системы децентрализованной аутентификации с применением системы блокчейн. Подробно описан принцип работы системы, её

достоинства и недостатки. Предложено использование криптокошелька MetaMask для взаимодействия с блокчейном.

Глава 1. Основы аутентификации

1.1 Основные понятия

С каждым зарегистрированным в компьютерной системе субъектом (пользователем или процессом, действующим от имени пользователя) связана некоторая информация, однозначно идентифицирующая его. Эту информацию называют идентификатором субъекта. Если пользователь имеет идентификатор, зарегистрированный в сети, он считается легальным (законным) пользователем; остальные пользователи относятся к нелегальным.

Прежде чем получить доступ к ресурсам компьютерной системы (авторизация), пользователь должен пройти процесс первичного взаимодействия с компьютерной системой, который включает идентификацию и аутентификацию.

Идентификация – это процедура распознавания пользователя по его идентификатору (имени). Это первый шаг в процессе получения доступа к системе. Как правило, идентификаторы зарегистрированных пользователей хранятся централизованно в базе данных. Система проверяет наличие предоставленного пользователем идентификатора в базе и при его наличии переходит к шагу аутентификации.

Аутентификация – процедура проверки подлинности заявленного пользователя, процесса или устройства. Эта проверка позволяет достоверно убедиться, что пользователь (процесс или устройство) является именно тем, кем себя объявляет. Как правило, процедура проверки основана на том, что пользователь передаёт некоторую секретную информацию, известную только ему – чаще всего это пароль – или каким-либо другим способом подтверждает факт владения такой информацией.

Авторизация – процедура предоставления субъекту определенных полномочий и ресурсов в данной системе. Авторизация происходит после

успешной аутентификации – пользователь, не прошедший аутентификацию, не может быть допущен к работе с системой. В рамках авторизации разным пользователям может быть предоставлен разный уровень доступа к системе – например, у администратора будет больше прав, чем у обычного пользователя [3].

Процесс идентификации, аутентификации и авторизации показан на рис. 1.1.

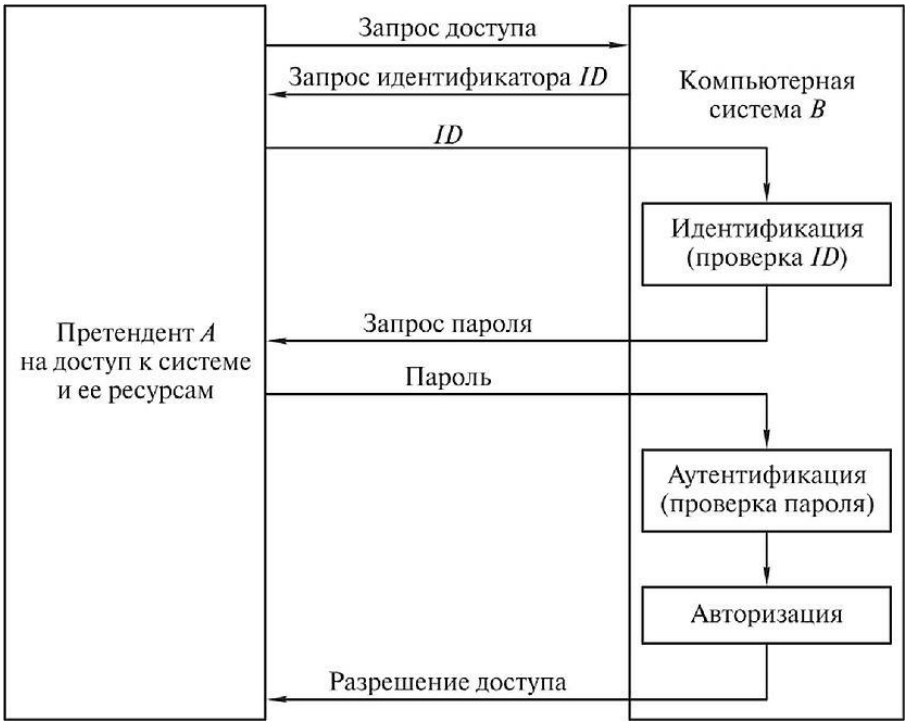

Рисунок 1.1 – Операции идентификации, аутентификации и авторизации

Также в контексте контроля доступа к системе иногда рассматривают такое понятие, как **учёт** (иногда также встречаются термины «аудит» или «администрирование»). Оно связано с ведением записей о действиях пользователя как в процессе входа в систему, так и после его авторизации [4].

1.2 Классификация систем аутентификации

В настоящий момент разработано большое количество систем аутентификации. Их можно классифицировать по многим параметрам, но основным является фактор, на котором основана аутентификация в той или иной системе.

В настоящий момент выделяют три основных фактора аутентификации: фактор знания, фактор владения, фактор свойства.

Фактор знания (what you know). В данном случае аутентификация основана на знании пользователем некоторой секретной информации, которую он должен предоставить по запросу системы. Самый распространённый вариант – использование пароля. Иногда также применяются секретные вопросы, ответ на которые должен знать только пользователь.

Системы аутентификации, основанные на факторе знания, являются наиболее распространёнными и применяются практически повсеместно. Их преимуществами являются лёгкость и низкая стоимость реализации. С другой стороны, такие системы часто бывают уязвимыми к различного рода атакам. Так, недостаточно надёжный пароль можно взломать или подобрать; если же сам пароль надёжный, но не уделено достаточное внимание безопасному хранению данных пользователей, то зачастую пароль удаётся похитить. Секретные вопросы тоже могут иметь достаточно низкую надёжность, поскольку данная информация может быть известна другим лицам, кроме самого пользователя.

Фактор владения (what you have). В данном случае аутентификация основана на факте владения пользователем некоторым уникальным предметом. Для получения доступа пользователь должен предоставить данный предмет или какую-либо информацию, полученную с его помощью.

Для аутентификации может быть использован физический предмет (ключ, токен), файл данных, мобильное устройство.

Для злоумышленника получить такое устройство обычно сложнее, чем взломать или похитить пароль. Кроме того, о факте кражи пользователь может достаточно быстро узнать и сообщить об этом, в результате чего злоумышленник уже не сможет воспользоваться данным устройством. Поэтому данный метод, как правило, обеспечивает более высокую степень безопасности по сравнению с предыдущим. С другой стороны, реализация системы, основанной на этом факторе, обычно является более дорогостоящей.

Фактор свойства (who you are). Системы, основанные на данном факторе, носят название биометрических. Здесь пользователь распознаётся по некой уникальной, присущей только ему физической характеристике [5].

Биометрические методы принято делить на статические и динамические. Статические методы (их можно назвать «классической» биометрикой) включают в себя аутентификацию по отпечатку пальца, форме лица, радужной оболочке глаза, рисунку вен и т. д. Динамические методы основаны на некоторой особенности поведения человека – голосе и манере речи, клавиатурном почерке, походке и др. Динамические методы изучены менее практических, и в настоящий момент идёт их активное исследование и развитие [6].

Биометрические системы, как правило, просты к использованию и обладают достаточно высокой надёжностью. Тем не менее, в их применении также есть ряд сложностей. Так, важно обеспечить одновременно низкий процент «ложных допусков» (то есть ошибочную аутентификацию нелегальных пользователей) и «ложных отказов» (отказов в аутентификации легальным пользователям). Кроме того, если биометрические данные оказываются украдены, пользователь уже не сможет ими воспользоваться – «сменить» такие данные, в отличие от пароля или устройства, по понятным причинам не представляется возможным. Наконец, реализация

биометрической системы аутентификации может быть достаточно дорогостоящей.

Кроме трёх основных факторов, существуют также некоторые другие – например, фактор времени и фактор пространства (расположения, координат). Однако в настоящий момент системы, основанные на таких факторах, используются относительно редко.

1.3 Примеры систем аутентификации

Рассмотрим наиболее распространённые системы аутентификации, основанные на каждом из трёх перечисленных факторов.

Безусловно, наиболее распространённым решением проблемы аутентификации на настоящий момент является использование логина и пароля – системы аутентификации, основанной на факторе знания. Самый простой пример реализации такой системы – база данных, в которой каждому пользователю, обладающему уникальным идентификатором (логином), соответствует пароль. Аутентификация считается успешной, если пользователь ввёл логин, существующий в базе, и соответствующий ему пароль.

Разумеется, хранение и передача паролей в открытом виде не является безопасной практикой – у злоумышленника есть возможность перехватить передаваемый пароль либо же взломать базу данных, в которой хранятся пароли, и получить доступ к данным всех пользователей. Несмотря на кажущуюся очевидность данных уязвимостей, подобные инциденты продолжают происходить до сих пор, и иногда жертвами становятся огромные компании, хранящие данные миллионов пользователей.

Несколько более безопасным вариантом является хранение пароля в виде хэша. Для этого используется хэш-функция – особая функция, обладающая следующими свойствами: во-первых, она преображает строку произвольной длины в строку фиксированной длины; во-вторых, даже

небольшое изменение (один-два символа) в исходной строке значительно влияет на полученную строку (в идеале изменяется не менее половины символов); в-третьих, по полученной строке практически невозможно восстановить исходную строку (функция является односторонней). В системе пароли не хранятся и не передаются – вместо этого пароль пользователя по определённому алгоритму преобразуется в хэш, который и хранится в базе [7].

В качестве примера системы, основанной на факторе владения, можно рассмотреть физические ключи. Одним из популярных вариантов реализации является YubiKey. Это устройство, внешне напоминающее USB флэш-накопитель. Для аутентификации пользователь должен ввести ключ в USB-порт и набрать на экране выбранный заранее пароль или PIN-код. После этого устройство отправляет в систему строку, состоящую из статического идентификатора (заменяющего собой логин) и одноразового кода (заменяющего собой пароль). Если строка верна, происходит аутентификация.

Существуют и другие распространённые примеры физических ключей – например, RSA SecurID или IronKey. Также популярным решением является использование мобильного телефона – данное устройство является уникальным, обладает достаточной вычислительной мощностью, а также не вынуждает пользователя переносить дополнительное устройство и отдельно следить за его сохранностью [8].

Кроме того, стоит отметить, что в большинстве современных реализаций фактор владения не применяется сам по себе – для защиты от взлома в случае кражи устройства практически всегда используются пароль или PIN-код, то есть методы, основанные на факторе знания.

Самым распространённым вариантом аутентификации с применением фактора свойства является использование отпечатков пальцев. Его применение основано на факте уникальности рисунка папиллярных узоров на кончиках пальцев каждого человека. Суть метода заключается в следующем: пользователь прикладывает палец к специальному сканеру, затем полученные

данные об отпечатке пальца преобразуются в цифровой код и сравниваются с кодами, имеющимися в базе данных системы идентификации. Весь процесс занимает не более 1 секунды.

Процент ложных допусков и ложных отказов будет в значительной степени зависеть от реализации метода. Так, для повышения точности можно использовать отпечатки сразу нескольких пальцев; с другой стороны, это сделает применение метода менее удобным. Также возможны атаки, связанные с копированием рисунка папиллярных узоров и изготовления «копии» пальца – для борьбы с ними система аутентификации должна быть усложнена, что повлияет на её стоимость [9].

В данной главе были рассмотрены основные понятия, связанные с аутентификацией пользователей в распределённых системах. Показано место аутентификации в более общей системе контроля доступа. Рассмотрены основные факторы, на которых может основываться тот или иной метод аутентификации. Приведены примеры систем аутентификации, основанных на данных факторах.

Глава 2. Современные методы аутентификации

2.1 Многофакторная аутентификация

Для обеспечения более высокого уровня защиты могут применяться методы аутентификации, в которых используется два или более фактора (либо же варианты одного фактора). Такие системы называются системами многофакторной аутентификации (multi-factor authentication, MFA).

Хотя MFA объединяет любое количество факторов аутентификации, наиболее распространенным методом является двухфакторная аутентификация (2FA). Необходимость в MFA также может быть вызвана неудачной идентификацией в 2FA или подозрительными действиями предполагаемой личности. Это характерно для систем 2FA, способных переходить в MFA. Это может также потребоваться для обеспечения дополнительной безопасности при доступе к более важным файлам или конфиденциальным данным, таким как медицинские или финансовые записи.

Дополнительные уровни безопасности в процессе входа в систему могут обеспечить уверенность в том, что личная информация пользователя останется защищённой и не попадёт в чужие руки.

В настоящее время распространённой реализацией метода 2FA является использование одноразовых паролей (one-time passwords, OTP). Как правило, для этого пользователь должен привязать к своему аккаунту номер телефона. В ходе процесса аутентификации пользователь должен сначала ввести логин и пароль, а затем получить на телефон одноразовый код (в зависимости от системы он может состоять как из одних цифр, так и из букв, цифр и специальных символов). Введение кода и будет являться вторым шагом двухфакторной аутентификации. Таким образом, совмещаются факторы знания (данные логина и пароля) и владения (мобильный телефон с определённым номером) [10].

Схема двухфакторной аутентификации с использованием одноразового пароля показана на рисунке 2.1.

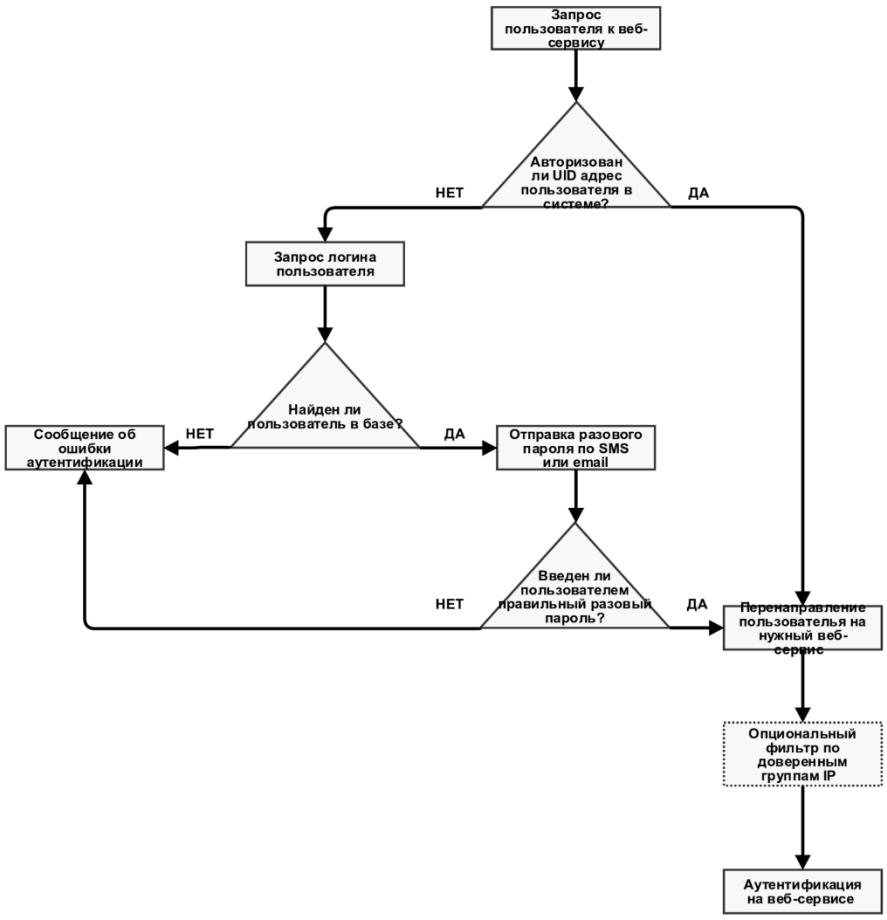

Рисунок 2.1 – Двухфакторная аутентификация с использованием одноразового пароля

Более современные факторы аутентификации учитывают контекст поведения входа в систему. Например, система может распознать, что хакер выполняет вход из необычного или сильно удалённого места, или заметить, что новое устройство пытается получить доступ к учётной записи. Система также учитывает время попытки входа в систему и тип сети, к которой пользователь обращается. Если какой-либо из этих факторов окажется необычным, будет активирована адаптивная аутентификация. Этот способ

идентификации сейчас очень популярен, так как позволяет собрать некоторые привычные факторы пользователя в единый портрет.

В настоящий момент системы многофакторной аутентификации получили широкое распространение. Они применяются, в частности, в сервисах Google, Microsoft, Dropbox, во многих банковских системах и мобильных приложениях банков и т. д. Кроме того, большинство крупных социальных сетей предоставляет пользователям возможность настроить двухфакторную аутентификацию для обеспечения более высокого уровня защищённости своего аккаунта [11].

2.2 Беспарольная аутентификация

В связи с этим всё большее распространение получают методы беспарольной аутентификации. В отличие от двухфакторной аутентификации, в них не используются методы, основанные на факторе знания. Такие системы обладают рядом преимуществ – простота использования, защита от многих распространённых видов атак, отсутствие необходимости создания большого числа паролей. Технологии беспарольной аутентификации находят всё более широкое применение и уже используются рядом крупных компаний – Google, Medium и др.

Одной из самых распространённых реализаций беспарольной аутентификации является технология magic link. При её применении конечному пользователю для регистрации или авторизации в системе нет необходимости использовать пароль – достаточно только ввести адрес электронной почты и перейти по ссылке, отправленной системой аутентификации. Ссылка является уникальной, и авторизация с её помощью возможна только для конкретного пользователя и только в течение ограниченного времени.

Работу системы аутентификации можно описать следующим образом: при входе в систему пользователю предлагается ввести свой адрес email. Если

такого адреса в системе нет, создаётся новый пользователь; в любом случае на данный адрес высылается ссылка, по которой пользователь должен перейти. Когда пользователь переходит по ссылке, происходит проверка подлинности. Если проверка оказывается пройдена успешно, происходит аутентификация, и пользователю предоставляется доступ к ресурсам системы.

Схема регистрации/авторизации с использованием magic link показана на рисунке 2.2.

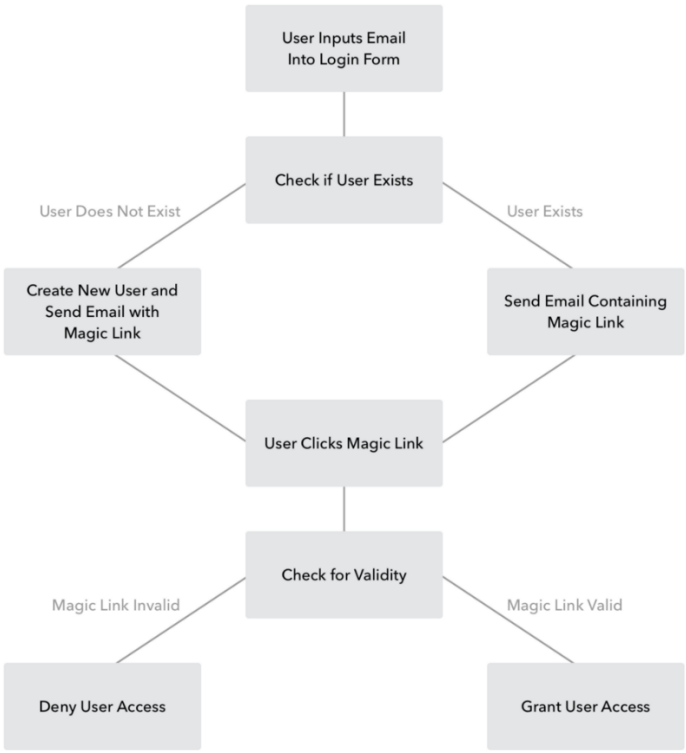

Рисунок 2.2 – Беспарольная аутентификация с использованием magic link

Такой подход не только значительно упрощает процесс регистрации новых пользователей и избавляет их от необходимости запоминания паролей, но и даёт надёжную защиту от ряда атак, связанных с похищением или подбором пароля [12, 13].

2.3 Аутентификация на основе OpenID

OpenID Connect (OIDC) – это протокол аутентификации, основанный на протоколе OAuth2 (который используется для авторизации). Для предоставления служб удостоверений OIDC использует стандартизованные потоки сообщений от OAuth2.

OIDC предоставляет пользователю возможность использовать одну учётную запись для аутентификации на множестве разных не связанных между собой сайтов. Как правило, в качестве поставщика (см. ниже) выбирается социальная сеть или система электронной почты.

OIDC позволяет разработчикам веб-сайтов и приложений выполнять аутентификацию пользователей без необходимости хранения и управления паролями. Таким образом, вместо самостоятельной реализации системы аутентификации разработчик может положиться на проверенную «третью сторону», будучи при этом уверен, что доступ к его системе будут получать только легальные пользователи.

Аутентификация пользователя должна выполняться в поставщике удостоверений, где будут проверяться сеанс или учетные данные пользователя. Для этого понадобится доверенный агент. Собственные приложения для этой цели обычно запускают браузер системы. Внедренные представления считаются ненадежными, так как ничто не помешает приложению перехватить пароль пользователя.

Как правило, у пользователя при аутентификации через OIDC запрашивается согласие. Это означает, что пользователь должен явно разрешить приложению доступ – полный или частичный – к своей учётной записи, предоставленной поставщиком. Как правило, при этом выводится на экран список конкретных действий, которое приложение сможет выполнять. Согласие обычно даётся один раз и остаётся в силе, пока пользователь или

администратор не отменят его. Этим согласие отличается от аутентификации, которая должна проводиться при каждой попытке входа в систему.

В систему OIDC входят следующие компоненты:

1. Пользователь: запрашивает услугу из приложения.
2. Доверенный агент: компонент, с которым взаимодействует пользователь. Этот доверенный агент обычно является веб-браузером.
3. Приложение: приложение или сервер ресурсов, где находится ресурс или данные. Он доверяет поставщику удостоверений безопасную аутентификацию и авторизацию доверенного агента.
4. Поставщик OIDC (также известный как поставщик удостоверений): безопасным способом обрабатывает все, что связано со сведениями о пользователе, с его доступом и отношениями доверия между участниками потока. Он проверяет подлинность пользователя, предоставляет и отзывает доступ к ресурсам, а также выдает маркеры [14].

В данной главе были рассмотрены некоторые системы аутентификации, применяемые в настоящий момент. Исследовано понятие многофакторной аутентификации, а также некоторые аспекты практической реализации систем MFA. Была рассмотрена беспарольная аутентификация, в частности, её разновидность, использующая адрес электронной почты пользователя. Наконец, подвергся рассмотрению протокол аутентификации OpenID Connect, основанный на протоколе авторизации OAuth2.

Глава 3. Сравнительный анализ методов аутентификации

3.1 Критерии сравнения методов аутентификации

Как было установлено в предыдущих частях данной работы, в настоящий момент существует большое количество различных методов аутентификации, основанных на самых разных принципах. Кроме того, аутентификация с использованием пары «логин-пароль», долгое время считавшаяся основным механизмом аутентификации для большинства случаев, становится всё менее и менее актуальной.

Эти факторы обуславливают необходимость проведения сравнительного анализа существующих методов аутентификации с целью выявления наиболее перспективных методов, которые могут использоваться в различных ситуациях, связанных с подтверждением личности пользователя.

Сравнение различных методов аутентификации, особенно относящихся к разным «классам» (основанных на разных факторах или комбинациях факторов) – непростая задача. Можно выделить целый ряд критериев сравнения, как количественных (например, проценты ложных допусков и ложных отказов), так и качественных (безопасность, надёжность, удобство использования и др.) Однако после этого перед нами появляются два важных вопроса: во-первых, как среди множества критериев выделить наиболее значимые, и во-вторых, как проводить сравнение, если критерии сами по себе комплексны и многогранны, а их точное определение зачастую неоднозначно?

В рамках ответа на первый вопрос сотрудники чилийского университета Био-Био Игнасио Веласкес, Анжелика Каро и Альфонсо Родригес провели исследования, в ходе которых они установили, какие именно критерии чаще

всего считаются наиболее важными при выборе того или иного метода аутентификации.

Так, на рис. 3.1 показано сравнение критериев по количеству научных статей на тему аутентификации, принимающих к рассмотрению тот или иной критерий [15].

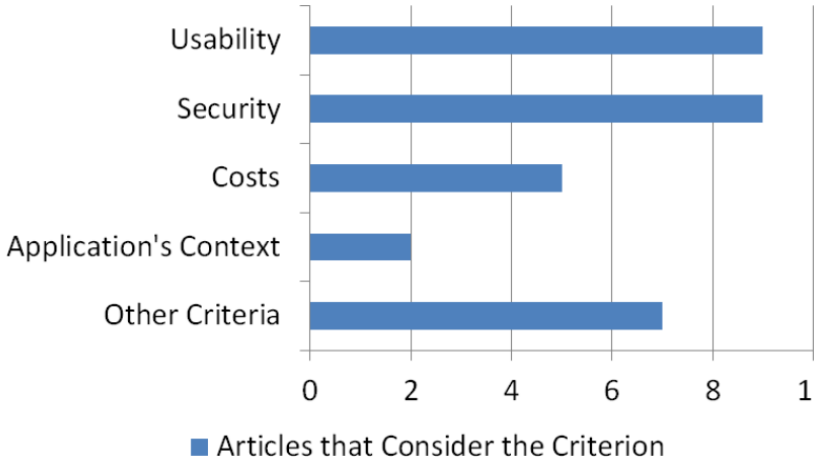

Рисунок 3.1 – Количество статей, учитывающих тот или иной критерий

В таблице 3.1 показаны критерии, которые чаще других называют важными опрошенные учёными представители компаний, оказавшихся перед выбором нового метода аутентификации [16].

Таблица 3.1 – Список критериев по количеству опрошенных, принимающих их во внимание

Criterion	Interviewees that consider the criterion
Client's requirements	11
Application context	11
Usability-related criteria	9
Security-related criteria	11
Cost-related criteria	8
Other criteria	2

Таким образом, из множества критериев, по которым возможно проводить сравнение методов аутентификации, в качестве наиболее важных стоит выделить следующие:

- Безопасность
- Удобство использования
- Развёртываемость
- Контекст приложения и требования клиента

В дальнейшем каждый из перечисленных критериев будет рассмотрен более развёрнуто.

3.2 Фреймворк сравнения механизмов аутентификации

Теперь мы можем сосредоточиться на втором из упомянутых выше вопросов, а именно: каким образом мы можем провести сравнение по любому из этих критериев, если каждый из них является достаточно общим и сам по себе включает в себя целый ряд факторов? Как можно определить – причём на стадии принятия решения о выборе метода, а не после внедрения метода и получения эмпирических данных о его использовании – какой из двух методов «более безопасен» или «менее удобен»?

Один из вариантов методики сравнения различных механизмов аутентификации был предложен в статье «The Quest to Replace Passwords: A Framework for Comparative Evaluation of Web Authentication Schemes», написанной исследователями Кембриджского университета Джозефом Бонно и Фрэнком Стаджано в соавторстве с Полом ван Оршотом из канадского Карлтонского университета и сотрудником фирмы «Майкрософт» Кормаком Хирли. Они предлагают фреймворк для оценки методов по трём критериям, который в статье назван «UDS» (usability-deployability-security) [17].

В рамках данного фреймворка каждый из критериев, в свою очередь, складывается из некоторого числа «преимуществ» (benefits), которые тот или иной механизм аутентификации может предоставлять пользователю. По

количеству преимуществ в каждой из трёх категорий, каждому из которых может быть присвоен свой вес в зависимости от его значимости, возможно провести количественное сравнение нескольких механизмов аутентификации и выявить наиболее подходящего.

Далее рассмотрим перечисленные выше критерии в рамках данного фреймворка. Приведём основные преимущества для каждого критерия, а затем покажем, как на их основе можно дать оценку того или иного метода. Наконец, рассмотрим несколько методов, упомянутых в прошлых частях работы, с точки зрения фреймворка UDS.

3.2.1 Безопасность

Безусловно, для любой системы аутентификации безопасность является одним из важнейших требований. Можно сказать, что само существование аутентификации как части процесса контроля доступа обусловлено требованиями к безопасности – пользователю, не являющемуся истинным обладателем предъявленного им идентификатора (это может быть как злоумышленник, так и совершивший ошибку «добропорядочный» пользователь), не должен быть предоставлен доступ к системе.

Преимущества, из которых складывается критерий безопасности в фреймворке UDS, в основном связаны с устойчивостью к различным атакам на механизм аутентификации. Векторов атаки может быть множество, и далеко не все из них связаны с «взломом» системы в привычном понимании.

Итак, для данного критерия можно выделить следующие преимущества:

1. *Устойчивость к физическому наблюдению.* К этому направлению атаки относятся наблюдение за экраном или получение результатов ввода с клавиатуры.
2. *Устойчивость к имперсонации.* Человек, хорошо знающий пользователя, не сможет получить доступ к системе, зная его

личные данные (например, в качестве ответа на «секретный вопрос»).

3. *Устойчивость к брутфорс-атакам.* Частичным соответствием можно считать устойчивость к атакам, количество которых за промежуток времени ограничено.
4. *Устойчивость ко внутреннему наблюдению.* Злоумышленник не может получить доступ к системе, перехватывая сообщения, идущие по каналу связи.
5. *Устойчивость к утечкам со стороны другого сервиса.* Данные, которые злоумышленник может получить при взломе сервера подтверждающей стороны, либо же при их утечке другим путём, не могут позволить ему пройти аутентификацию под видом того же пользователя в другом сервисе.
6. *Устойчивость к фишинг-атакам.* Злоумышленник не может получить данные пользователя, необходимые для аутентификации, путём симуляции реального сервиса.
7. *Устойчивость к краже физического ключа.* Если злоумышленник получил в свои руки физическое устройство, он не может использовать его для входа в систему под видом пользователя. Частичным соответствием можно считать слабую защиту вроде PIN-кода.
8. *Отсутствие доверенной третьей стороны.* Метод не полагается на третью сторону, которая также могла бы подвергнуться атаке и стать источником утечки данных.
9. *Необходимость явного согласия.* Процесс аутентификации не может начаться без явного согласия со стороны пользователя.
10. *Невозможность связать различные аутентификаторы одного и того же пользователя.* Это преимущество обеспечивает защиту личности пользователя. Здесь не учитываются такие факторы, как, например, один и тот же IP-адрес.

3.2.2 Удобство использования

Важность данного критерия зачастую недооценивается специалистами по безопасности. Тем не менее, в ряде случаев его значимость не подлежит сомнению.

В настоящий момент значительная часть распределённых систем, требующих аутентификации, рассчитана на «обычного» пользователя, не обладающего специальными знаниями в области безопасности и информационных технологий. Механизм аутентификации, сложный в освоении и неудобный в использовании, может серьёзно затруднить работу с системой для рядового пользователя, и вряд ли получит широкое распространение.

Более того, удобство использования может оказывать влияние и на безопасность механизма аутентификации. Хрестоматийный пример – длинные, сложные для запоминания пароли, которые многие пользователи предпочитают записывать на бумажных носителях, что само по себе небезопасно. Необходимость запоминания большого числа паролей ведёт к их повторному использованию, что повышает уровень риска при краже одного из них [18].

К преимуществам, которые системы аутентификации могут предоставлять в области удобства применения, относятся следующие:

1. *Отсутствие необходимости запоминания информации.* Пользователю не нужно запоминать каких-либо секретов (паролей, кодов). Частичным соответствием можно считать случай, при котором пользователь должен запомнить только один пароль вне зависимости от количества аккаунтов.
2. *Масштабируемость с точки зрения пользователя.* Использование метода для создания большого количества аккаунтов не отражается на опыте конкретного пользователя.

3. *Отсутствие необходимости ношения физического объекта*. Частичным соответствием можно считать использование мобильного телефона (так как большинство людей носят его с собой в любом случае).
4. *Физическая лёгкость аутентификации.* Нет необходимости, например, вводить длинные пароли или фразы.
5. *Простота освоения.* Легко понять и запомнить, как использовать метод.
6. *Быстрота использования.* Учитывается как время, необходимое для входа, так и время, необходимое для создания новой учётной записи.
7. *Малое количество ошибок при входе.* Значение уровня ложных отказов (FRR) должно быть достаточно низким.
8. *Простота восстановления.* Если пользователь по тем или иным причинам потерял возможность входа в систему, должна существовать возможность легко и быстро заново получить доступ.

3.2.3 Развёртываемость

Развёртываемость (от англ. deployability – возможность развёртывания) – это достаточно широкий критерий, в общем случае определяющий, насколько просто и удобно можно внедрить тот или иной механизм в реальную распределённую систему.

В рамках фреймворка UDS в критерий развёртываемости входят следующие возможные преимущества:

1. *Малая цена за одного пользователя.* Стоимость подключения пользователя к системе (как со стороны пользователя, так и со стороны владельца системы) должна быть относительно небольшой.

2. *Совместимость с сервером.* Метод совместим с использованием текстовых паролей.
3. *Совместимость с браузерами.* Пользователи могут пользоваться любым современным браузером без необходимости установки дополнительного ПО.
4. *История применения.* Данный метод уже применяется в масштабной системе аутентификации. Также могут учитываться проекты, использующие данный метод, наличие документации, прохождение тестов и т. д.
5. *Открытый доступ.* Исходный код проекта доступен открыто, нет необходимости платить за использование метода.
6. *Доступность.* Метод аутентификации может использоваться пользователями с ограниченными возможностями (по крайней мере, теми, кто может использовать аутентификацию на основе логина и пароля).

3.3 Методика численного сравнения механизмов аутентификации

Итак, мы показали, из чего складываются три критерия, составляющие данный фреймворк. По наличию или отсутствию каждого из преимуществ можно дать оценку соответствия того или иного метода аутентификации требованиям каждого из критериев.

Здесь стоит особо уточнить несколько важных моментов.

Во-первых, преимущества критериев не обязаны быть бинарными («предоставляется/не предоставляется»). Как минимум, в ряде случаев возможно частичное предоставление какого-либо преимущества. Таким образом, в самом простом случае при численной оценке преимуществу можно присваивать значение 0 (не предоставляется), 0,5 (предоставляется частично) или 1 (предоставляется). Теоретически возможна и более точная градация.

Во-вторых, значимость каждого из преимуществ в рамках одного критерия может быть различной. Так, например, совместимость с браузером может быть маловажной в случае, если использование браузера в данном контексте не предполагается. Поэтому для более точной численной оценки каждому из преимуществ можно приписать определённый вес, зависящий от его важности в том или ином контексте.

(Таким образом, хотя мы рассматриваем всего три критерия, контекст приложения и требования пользователя тоже оказываются учтены при окончательной оценке)

Итак, соответствие механизма аутентификации какому-либо критерию можно оценить по следующей формуле:

$$S_i = \sum_j W_j \cdot b_j$$

где

S_i – количественное значение критерия i

W_j – вес, присваиваемый определённому преимуществу данного критерия

b_j – наличие (частичное или полное) или отсутствие данного преимущества в рассматриваемом механизме аутентификации

Поскольку в фреймворке UDS рассматривается три критерия, $i \in [1, 2, 3]$.

Значения весов зависят от конкретного сценария, требующего аутентификации, но «по умолчанию» можно считать $W_j = 1$ для всех j. Наконец, в самом простом случае, как было сказано выше, $b_j \in [0, 0{,}5, 1]$.

3.4 Сравнение распространённых методов аутентификации

В качестве иллюстрации данного подхода проведём численное сравнение рассмотренных ранее методов аутентификации, используя описанный фреймворк.

Перед началом сравнения необходимо упомянуть о нескольких сделанных допущениях. Так, каждое преимущество может принимать значения [0, 0,5, 1], как было показано ранее. Кроме того, для простоты все веса примем равными 1. Разумеется, для сравнения в конкретном контексте значения весов будут различны; полученные далее результаты не следует принимать за полное и универсальное сравнение методов.

Соответствие методов критериям показано в таблицах 3.2–3.4.

Таблица 3.2. Соответствие рассматриваемых методов критерию безопасности

Метод/критерий	Физическое наблюдение	Имперсонация	Брутфорс-атаки	Внутреннее наблюдение	Утечки от др. сервисов	Фишинг-атаки	Кража ключа	Третья сторона	Явное согласие	Связь аутентификаторов
	1	2	3	4	5	6	7	8	9	10
Логин и пароль	0	0,5	0	0	0	0	1	1	1	1
Биометрия (отпечаток пальца)	1	0	1	0	0	0	0	1	1	0
Физический ключ	1	1	1	1	1	1	1	0	1	1
2FA (пароль + OTP)	1	1	1	0,5	1	1	0,5	0	1	1
Magic link	0,5	0,5	0,5	0	1	0	1	0	1	0
OpenID	0,5	0,5	0,5	0	1	0	1	0	1	0

Таблица 3.3. Соответствие рассматриваемых методов критерию удобства использования

Метод/критерий	Запоминание	Масштабируемость	Ношение физ. ключа	Физ. лёгкость	Простота освоения	Быстрота использования	Ошибки при входе	Восстановление
	1	2	3	4	5	6	7	8
Логин и пароль	0	0	1	0	1	1	0,5	1
Биометрия (отпечаток пальца)	1	1	1	0,5	1	0,5	0	0
Физический ключ	0	0	0	0	1	0,5	0,5	0
2FA (пароль + OTP)	1	1	0,5	0	1	0	0,5	0,5
Magic link	0,5	1	1	0,5	0,5	1	1	0,5
OpenID	0,5	1	1	0,5	0,5	1	1	1

Таблица 3.4. Соответствие рассматриваемых методов критерию развёртываемости

Метод/критерий	Малая цена	Сервер	Браузер	История	Открытый доступ	Доступность
	1	2	3	4	5	6
Логин и пароль	1	1	1	1	1	1
Биометрия (отпечаток пальца)	0	0	0	0,5	0	0,5
Физический ключ	0	0	1	1	0	0
2FA (пароль + OTP)	0	0	1	1	1	0,5
Magic link	1	0	1	0,5	1	1
OpenID	1	0	1	1	1	1

Итак, мы можем провести численное сравнение, основываясь на приведённых выше формулах.

Результаты сравнения по критериям показаны на рисунках 3.2–3.4.

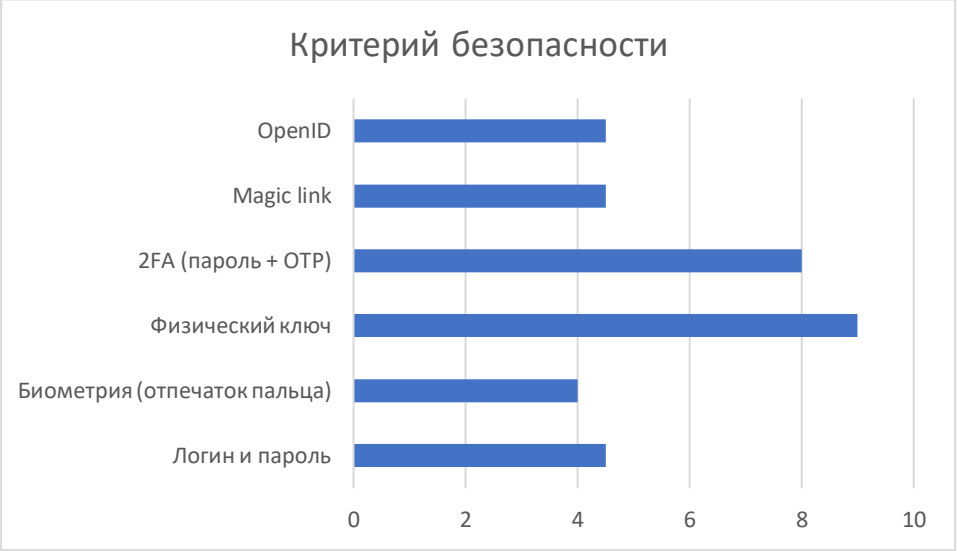

Рисунок 3.2 – Сравнение методов по критерию безопасности

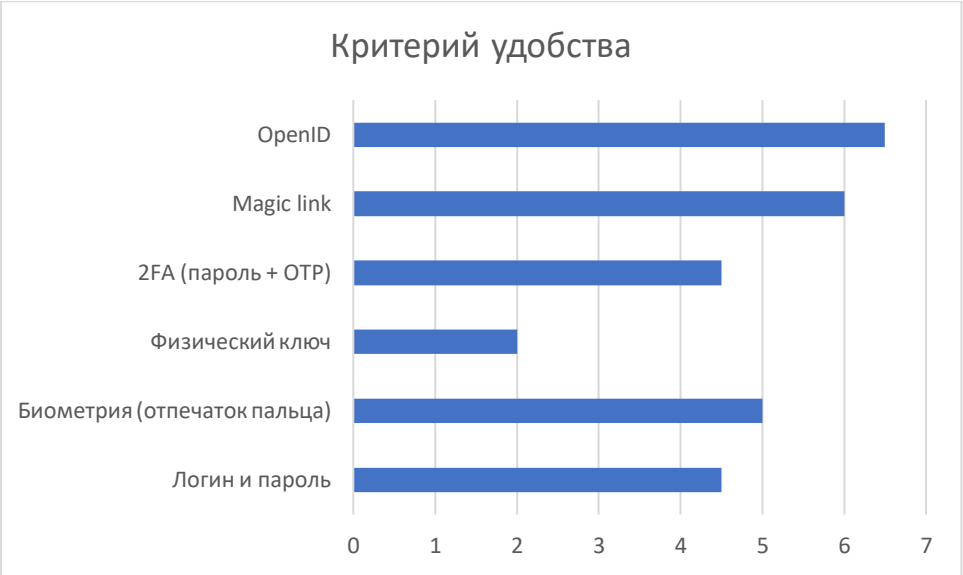

Рисунок 3.3 – Сравнение методов по критерию удобства использования

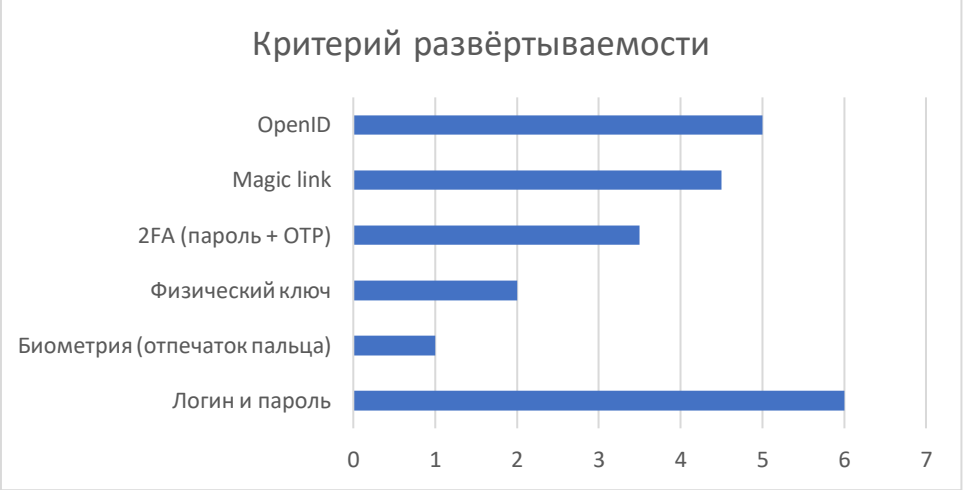

Рисунок 3.4 – Сравнение методов по критерию развёртываемости

На рисунке 3.5 показано сравнение методов аутентификации по трём критериям. По оси x отложена безопасность, по оси y – удобство использования, а размер круга показывает развёртываемость.

Рисунок 3.5 – Сравнение методов по трём критериям

Стоит ещё раз подчеркнуть, что полученные результаты не являются единственным возможным исходом сравнения различных методов. В зависимости от конкретной задачи, стоящей перед нами, мы можем придавать различным преимуществам большее или меньшее значение, что, в свою очередь, отразится на результатах сравнения.

В данной главе была рассмотрена проблема сравнительной оценки методов аутентификации. Были идентифицированы наиболее значимые критерии сравнения, а также предложен метод численной оценки соответствия метода данным критериям. Каждый критерий рассмотрен более подробно. Приведён пример численного сравнения нескольких методов аутентификации.

Глава 4. Сценарии аутентификации пользователей

4.1 Примеры систем, в которых требуется аутентификация пользователей

В предыдущей главе был рассмотрен фреймворк для сравнения различных методов аутентификации, а также приведён пример сравнения наиболее распространённых методов. Однако для простоты сравнения сделано допущение, состоящее в том, что каждый аспект каждого критерия обладает одинаковой важностью. Сами авторы статьи, в которой впервые введён фреймворк UDS, прямо предупреждают, что не стоит просто складывать бинарные результаты и использовать их суммы для оценки системы.

Для учёта контекста конкретного приложения будем использовать такой элемент фреймворка, как веса. Самая простая классификация, использующая их, может выглядеть следующим образом:

1 – аспект незначителен в данном контексте

2 – аспект достаточно значим в данном контексте

3 – аспект является одним из ключевых в данном контексте

Таким образом, мы учитываем не только количество преимуществ, которые та или иная система предоставляет по каждому критерию, но и то, насколько они важны для конкретного приложения. Это позволяет дать более точную оценку за счёт конкретизации задачи.

Распределённые системы можно разделить на несколько групп в зависимости от вариантов взаимодействия между участниками системы. В частности, можно выделить следующие сценарии:

1. Взаимодействие «клиент-сервер» – стороны неравноправны, одна сторона (клиент) производит запросы, которые выполняет другая

сторона (сервер). Как правило, на один сервер приходится множество клиентов.

2. Взаимодействие «peer-to-peer» (P2P) – все стороны равноправны и могут как отправлять, так и обрабатывать запросы.
3. Многоуровневая система – разновидность сценария «клиент-сервер», при котором между клиентом и сервером существуют одна или несколько прослоек [19].

Каждый из данных сценариев можно рассмотреть в рамках конкретного кейса, то есть примера системы, предназначенной для решения конкретной задачи.

Итак, рассмотрим несколько примеров кейсов, в которых требуется аутентификация пользователей. Присвоив различным аспектам веса в соответствии с системой, приведённой выше, посмотрим, как это отразится на сравнительной оценке разных методов аутентификации.

4.2 Хранилище данных медицинского учреждения

Рассмотрим систему, в которой хранятся данные пациентов медицинского учреждения. Попробуем определить основные особенности такой системы с точки зрения требований к аутентификации [20].

В данной системе реализуется сценарий «клиент-сервер» – к базе, в которой хранятся данные, обращаются пользователи системы (врачи и, возможно, пациенты).

В целом, в такой системе наиболее важным стоит считать критерий безопасности. Данные о здоровье человека являются частью его персональных данных, и система должна гарантировать их защищённость. Разумеется, удобство использования также является важным (особенно если системой будут пользоваться не только врачи – которых можно обучить пользованию системой, даже если она не особенно интуитивно понятна – но и пациенты).

Развёртываемость тоже следует учитывать, но можно сказать, что в сравнении с первыми двумя критериями она может быть менее важной.

Рассмотрим аспекты каждого из трёх критериев подробнее.

В условиях больницы – то есть общественного учреждения, в котором может работать большое число людей – важны устойчивость к физическому наблюдению, имперсонации и краже физического ключа, так как вероятность атак с данных направлений повышена по сравнению со случаем, когда доступ к системе пользователь обычно получает из своего дома. С другой стороны, невозможность связать между собой аутентификаторы одного и того же пользователя будет менее важна – напротив, в отношении врачей может применяться контроль их действий в системе.

В плане удобства использования важными аспектами стоит признать отсутствие необходимости ношения физического объекта, простоту освоения метода, простоту восстановления доступа. Менее важной будет масштабируемость с точки зрения пользователя, поскольку не предполагается наличия у одного пользователя аккаунта в множестве подобных систем.

Наконец, в плане развёртываемости важными критериями будут история применения (медицинское учреждение, скорее всего, будет доверять доказавшей себя системе) и доступность (если предполагается использование системы лицами с ограниченными возможностями). С другой стороны, доступность исходного кода и бесплатность программы можно считать менее важными.

В таблице 4.1 показаны веса, присвоенные аспектам каждого критерия в данной системе.

Таблица 4.1 – Веса критериев для сценария системы хранения данных пациентов

Критерий/преимущество	1	2	3	4	5	6	7	8	9	10
Безопасность	3	3	2	2	2	2	3	2	2	1
Удобство использования	2	1	3	2	3	2	2	3	-	-
Развёртываемость	2	2	2	3	1	3	-	-	-	-

Результаты сравнения по критериям с учётом весов показаны на рисунках 4.1–4.4.

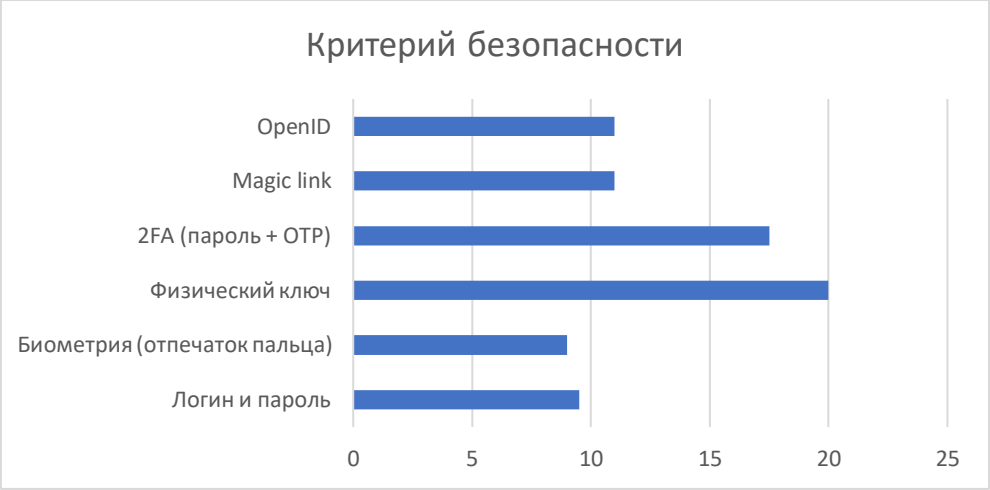

Рисунок 4.1 – Сравнение методов по критерию безопасности

Рисунок 4.2 – Сравнение методов по критерию удобства использования

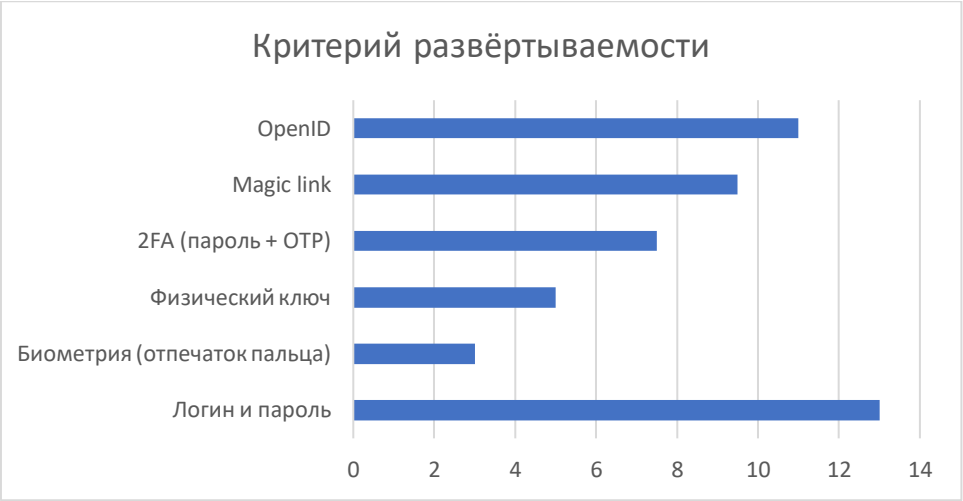

Рисунок 4.3 – Сравнение методов по критерию развёртываемости

Рисунок 4.4 – Сравнение методов по трём критериям

Как можно видеть, сохранились некоторые общие тенденции: так, например, физический ключ всё ещё получает наиболее высокую «оценку» по безопасности, но самую низкую – по удобству использования. С другой

стороны, заметны и изменения: так, относительная оценка системы «логин-пароль» в области безопасности снизилась, как и оценка биометрической системы (отпечатка пальца) в области удобства использования.

4.3 Интернет-мессенджер

Подобная система реализует сценарий «peer-to-peer». Все участники равноправны и могут принимать и отправлять сообщения, загружать контент и т. д [21].

В данной системе безопасность будет несколько менее важна (хотя, разумеется, обеспечить её на достаточном уровне всё ещё является необходимым). Ключевым критерием будет являться удобство использования, так как мессенджер рассчитан на использование широким кругом лиц.

Если рассматривать систему с точки зрения безопасности, то важными преимуществами будут являться устойчивость к имперсонации, к брутфорс-атакам и фишинг-атакам. С другой стороны, указать на какие-либо аспекты как на незначительные в данном случае не представляется возможным.

В плане удобства использования важными аспектами будут масштабируемость с точки зрения пользователя (вполне вероятно, что пользователь будет зарегистрирован в нескольких мессенджерах и социальных сетях), отсутствие физического ключа, простота освоения и восстановления. Менее важным будет малое количество ошибок при входе.

Наконец, с точки зрения развёртываемости важными факторами будут малая цена за одного пользователя и совместимость с браузерами. Менее важной будет открытость исходного кода.

В таблице 4.2 показаны веса, присвоенные аспектам каждого критерия в данной системе.

Таблица 4.2 – Веса критериев для сценария мессенджера

Критерий/преимущество	1	2	3	4	5	6	7	8	9	10
Безопасность	2	3	3	2	2	3	2	2	2	2
Удобство использования	2	3	3	2	3	2	1	3	-	-
Развёртываемость	3	2	3	2	1	2	-	-	-	-

Результаты сравнения по критериям с учётом весов показаны на рисунках 4.5–4.8.

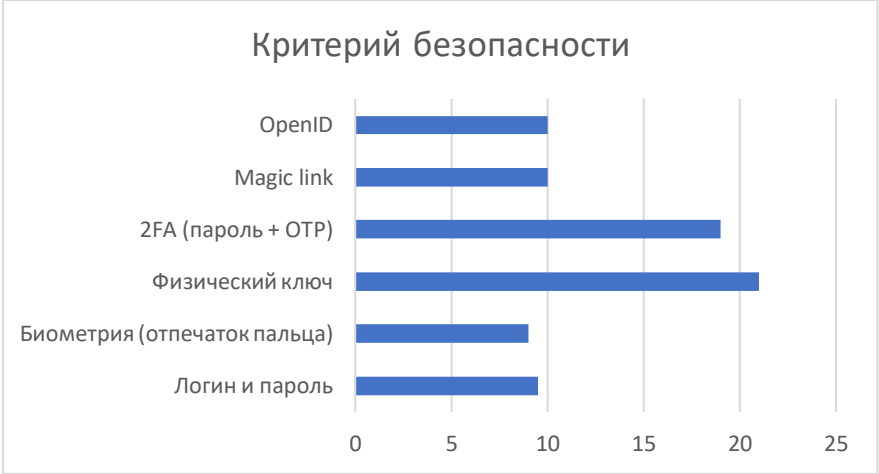

Рисунок 4.5 – Сравнение методов по критерию безопасности

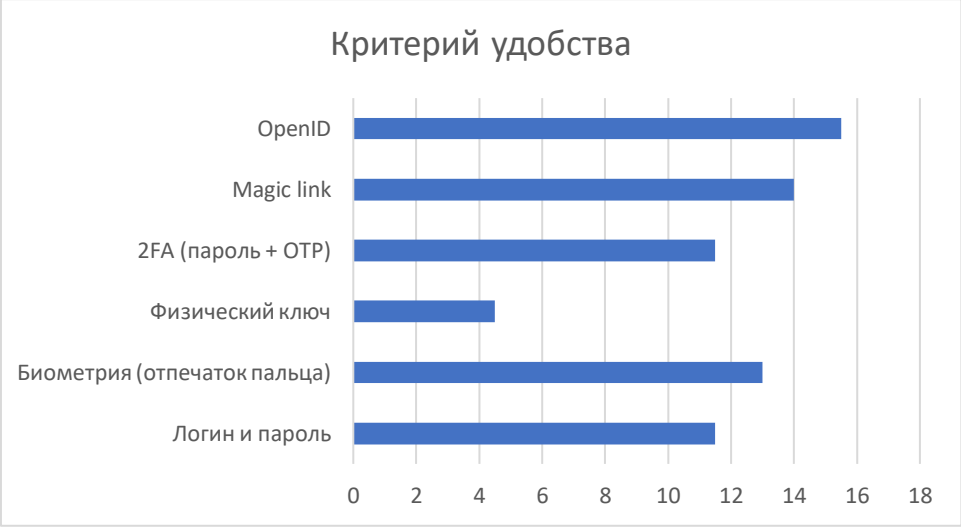

Рисунок 4.6 – Сравнение методов по критерию удобства использования

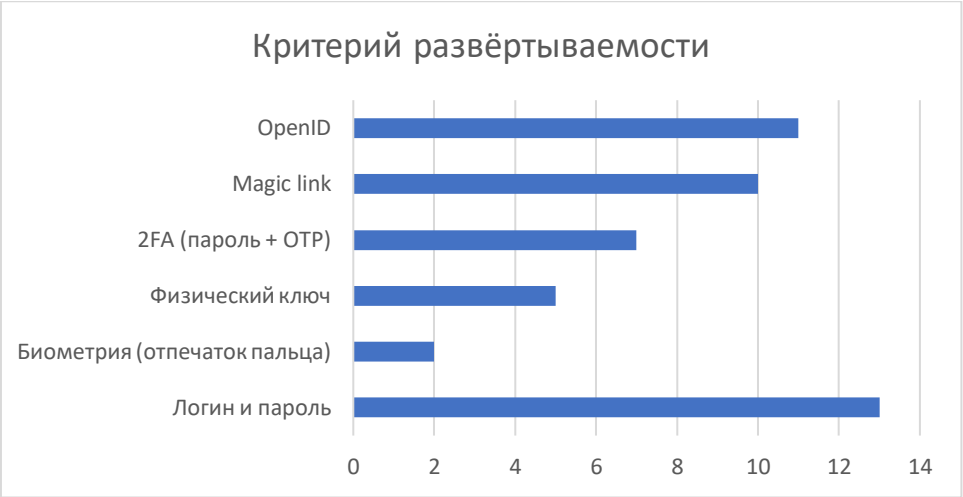

Рисунок 4.7 – Сравнение методов по критерию развёртываемости

Рисунок 4.8 – Сравнение методов по трём критериям

В сравнении с предыдущим кейсом можно заметить, например, что двухфакторная аутентификация получила несколько более высокую оценку: это соответствует реальной практике, где она всё чаще применяется в

мессенджерах и социальных сетях. С другой стороны, физические ключи, хотя и предлагающие высокую безопасность, показывают очень низкую оценку с точки зрения удобства использования, что в связи с рассмотренной ранее сравнительной важностью критериев не позволяет порекомендовать их для использования в данном случае.

4.4 GRID-система

Для доступа к системе GRID обычно используется многоуровневая архитектура. Система является централизованной и в целом напоминает архитектуру «клиент-сервер», но между пользователем и хранилищем данных встраивается дополнительный слой, отвечающий за их взаимодействие [22].

В данной системе стоит уделить внимание безопасности и развёртываемости. Хотя удобство использования тоже будет важным фактором, стоит учитывать, что предполагается использование системы в основном сотрудниками организации, для которых можно организовать специальное обучение.

С точки зрения безопасности будут важны такие факторы, как устойчивость к физическому наблюдению, имперсонации, краже физического ключа, в то время как невозможность связать между собой аутентификаторы менее значительна. В целом требования к безопасности похожи на случай базы данных пациентов.

Если рассматривать удобство использования, то менее важными будут масштабируемость с точки зрения пользователя, отсутствие физического ключа, простота освоения.

В плане развёртываемости важной будет совместимость с сервером. Менее важны малая цена за пользователя, история применения и открытость доступа.

В таблице 4.3 показаны веса, присвоенные аспектам каждого критерия в данной системе.

Таблица 4.3 – Веса критериев для сценария системы интернета вещей

Критерий/преимущество	1	2	3	4	5	6	7	8	9	10
Безопасность	3	3	2	2	2	2	3	2	2	1
Удобство использования	2	1	1	2	1	2	2	2	-	-
Развёртываемость	1	3	2	1	1	2	-	-	-	-

Результаты сравнения по критериям с учётом весов показаны на рисунках 4.9–4.12.

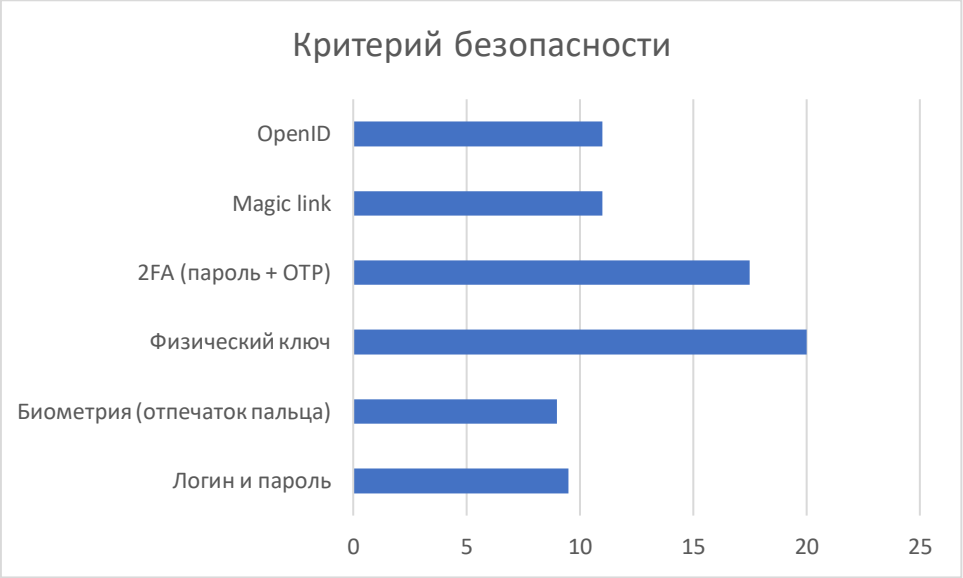

Рисунок 4.9 – Сравнение методов по критерию безопасности

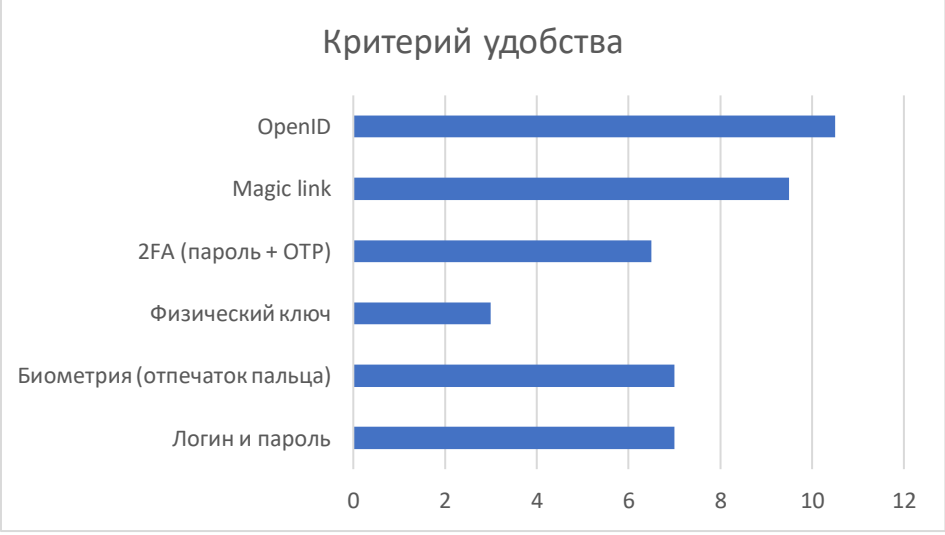

Рисунок 4.10 – Сравнение методов по критерию удобства использования

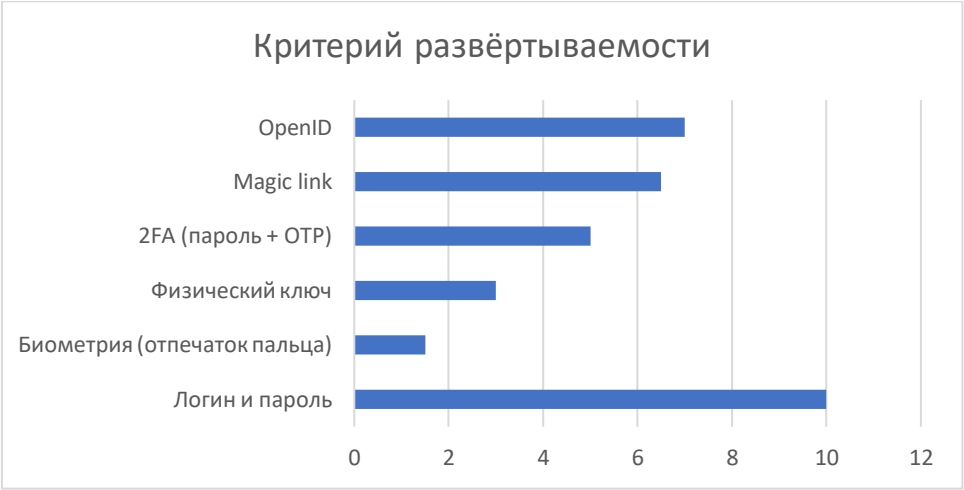

Рисунок 4.11 – Сравнение методов по критерию развёртываемости

Рисунок 4.12 – Сравнение методов по трём критериям

Как и в предыдущих случаях, физический ключ будет наиболее безопасным выбором, обладая при этом не самой малой развёртываемостью. Другие методы – такие, как двухфакторная аутентификация, использование

magic link или OpenID – превосходят его по развёртываемости и удобству, но уступают в области безопасности, хотя их всё ещё можно считать более безопасными, чем стандартные логин и пароль.

4.5 Общее сравнение систем

В предыдущих разделах с помощью фреймворка UDS было рассмотрено три различных сценария аутентификации. Как мы увидели, с помощью весов можно учитывать влияние требований конкретного сценария.

Проведём теперь общее сравнение рассмотренных методов аутентификации в отношении данных сценариев. Оно показано в таблице 4.4.

Таблица 4.4 – Общее сравнение методов в различных сценариях

Метод/критерий	Безопасность	Удобство использования	Развёртываемость
Система данных пациентов			
Логин и пароль	9,5	12	13
Биометрия (отпечаток пальца)	9	11	3
Физический ключ	20	5	5
2FA (пароль + OTP)	17,5	10	7,5
Magic link	11	13	9,5
OpenID	11	14,5	11
Интернет-мессенджер			
Логин и пароль	9,5	11,5	13
Биометрия (отпечаток пальца)	9	13	2
Физический ключ	21	4,5	5
2FA (пароль + OTP)	19	11,5	7
Magic link	10	14	10
OpenID	10	15,5	11
GRID-система			
Логин и пароль	9,5	7	13
Биометрия (отпечаток пальца)	9	7	2
Физический ключ	20	3	5
2FA (пароль + OTP)	17,5	6,5	7
Magic link	11	9,5	10
OpenID	11	10,5	11

В данной главе были рассмотрены примеры сценариев, в которых требуется аутентификация пользователей. В частности, был проведён анализ требований к механизму аутентификации в хранилище данных медицинского учреждения, интернет-мессенджере и системе GRID. Можно сделать вывод о том, что конкретный сценарий будет значительно влиять на сравнительную важность как критериев, так и конкретных преимуществ той или иной системы. Таким образом, не представляется возможным выбор «самого лучшего» метода аутентификации, который был бы оптимальным выбором для любого сценария.

Глава 5. Суверенная идентичность и децентрализованные идентификаторы

5.1 Понятие суверенной идентичности (self-sovereign identity)

Все рассмотренные ранее системы аутентификации пользователей объединяет одно – они являются централизованными. Это значит, что данные пользователей хранятся централизованно на стороне системы. Как правило, существует единая база данных, в которую записываются данные, необходимые для входа. В большинстве рассмотренных методов это сервер системы, требующей аутентификации; в случае magic link и OpenID это будут, соответственно, провайдер электронной почты и социальная сеть, аккаунт которой использован для входа в систему.

Такой подход, несмотря на свою распространённость, обладает несколькими серьёзными недостатками. Один из них связан с безопасностью: взлом системы или утечка данных с её серверов могут привести к компрометации большого количества пользователей и попаданию их личных данных в открытый доступ. Происходили и продолжают происходить инциденты, последствиями которых становится утечка данных сотен тысяч и даже миллионов пользователей банков, социальных сетей, систем электронной почты и др. Таким образом, хранилище данных пользователей само по себе будет являться «узким местом» любой системы, представляя собой желанную цель для злоумышленников.

Кроме того, существуют проблемы с точки зрения удобства использования. Зачастую пользователям приходится запоминать или хранить большое количество паролей от различных систем и сервисов; на практике, многие пользователи используют повторяющиеся пароли, что по очевидным причинам представляет собой риск с точки зрения безопасности. Даже в

системах, не требующих пароля, разные сервисы имеют разные интерфейсы, правила использования и т. д., что неудобно для пользователей, имеющих аккаунты в множестве различных сервисов.

Наконец, централизованность системы означает, что пользователи должны передать контроль над своими данными в руки другой стороны. В последнее время же всё больше распространяется идея о том, что пользователь должен сам контролировать собственные данные и самостоятельно решать, где, когда и в каком объёме раскрывать их при взаимодействии с различными системами и сервисами [23].

Все эти факторы привели к созданию такого понятия, как суверенная идентичность (англ. self-sovereign identity, SSI). Рассмотрим его подробнее.

Технология суверенной идентичности предполагает наличие децентрализованной системы, в которой пользователи могут хранить свои данные. При этом отсутствует единый реестр, и каждый пользователь полностью контролирует свои данные. Вместе с этим обеспечивается верифицируемость данных, то есть запрашивающая их сторона может подтвердить данные и убедиться в их достоверности.

Децентрализованная система, о которой идёт речь – это, как правило, блокчейн. При этом данные пользователей хранятся вне публичного реестра. Вместо этого они находятся в личном цифровом кошельке пользователя, и он может управлять ими через специальное приложение.

В рамках технологии SSI идентичности пользователей являются, с одной стороны, постоянными и портативными: это значит, что пользователь может использовать один и тот же идентификатор в различных сервисах на протяжении длительного времени, и нет необходимости создания большого количества отдельных цифровых идентичностей. С другой стороны, пользователь может создать несколько независимых идентификаторов и

пользоваться ими так, что установить связь между ними будет крайне сложно: это помогает защитить личность пользователя.

К другим требованиям, выдвигаемым к системе суверенной идентичности, относятся прозрачность используемых систем и алгоритмов, необходимость явного подтверждения согласия на использование данных, а также возможность выборочного раскрытия пользователем своих данных [24].

5.2 Децентрализованные идентификаторы (DID)

С технологией суверенной идентичности связано понятие децентрализованного идентификатора (англ. decentralized identifier, DID). Он представляет из себя уникальный идентификатор, позволяющий однозначно определить объект в децентрализованной системе.

DID должен отвечать следующим основным требованиям:

1. Децентрализованность – отсутствие единого центра сертификации.
2. Стойкость – идентификатор должен быть постоянным, его работа не должна зависеть от функционирования какой-либо конкретной организации.
3. Криптографическая подтверждаемость – возможность подтвердить владельца идентификатора с помощью методов криптографии.
4. Разрешаемость – возможность получить метаданные о конкретном идентификаторе.

DID идентифицирует любой субъект (напр., лицо, организацию, вещь, модель данных, абстрактный объект и т. д.), который контроллер DID решает идентифицировать. В отличие от обычных идентификаторов, DID спроектированы таким образом, что их можно отделить от централизованных реестров, поставщиков удостоверений и центров сертификации. В частности, в то время как другие стороны могут использоваться для помощи в

обнаружении информации, связанной с DID, контроллер DID имеет возможность доказать свой контроль над ним, не требуя разрешения от какой-либо другой стороны.

DID – это URI, которые связывают тему DID с документом DID, позволяя надежные взаимодействия, связанные с этим предметом.

Каждый документ DID может содержать в себе криптографические материалы, методы проверки или конечные точки, которые предоставляют набор механизмов, позволяющих контроллеру DID подтверждать контроль над DID. Конечные точки обеспечивают доверительные взаимодействия, связанные с предметом DID. Документ DID может содержать сам предмет DID, если предметом DID является информационный ресурс, такой как модель данных.

DID и документы DID могут быть адаптированы к любой современной системе blockchain, распределенному реестру или другой децентрализованной сети, способной преобразовывать уникальный ключ в уникальное значение. Не имеет значения, является ли блокчейн общедоступным или частным [25].

DID может быть использован для аутентификации пользователя в системе. В частности, механизм подтверждения контроля децентрализованного идентификатора может быть использован для подтверждения личности пользователя в распределённой системе [26].

В данной главе рассмотрено понятие суверенной идентичности, её основания и необходимость. Перечислены факторы, приведшие к появлению SSI, и основные требования, выдвигаемые к системам данного класса. Также исследовано понятие децентрализованного идентификатора, его основные свойства и функции.

Глава 6. Практическая реализация системы аутентификации

6.1 Двухфакторная система аутентификации с использованием OTP

В ходе исследования тематики аутентификации в распределённых системах была реализована на практике система адаптивной двухфакторной аутентификации.

Для реализации системы использован программный продукт с открытым исходным кодом Keycloak. Данный продукт реализует технологию single sign-on – это означает, что пользователь может переходить из одной системы в другую, связанную с первой, без повторной аутентификации.

Также был использован Google Authenticator. Это мобильное приложение, разработанное компанией Google. С его помощью пользователи могут использовать двухэтапную аутентификацию с применением Time-based One-time Password Algorithm (TOTP) и HMAC-based One-time Password Algorithm (HOTP) от Google LLC. Данный сервис реализует алгоритмы, указанные в стандартах RFC 6238 и RFC 4226 [27].

Двухфакторная аутентификация была реализована с применением одноразовых паролей, основанных на времени (TOTP). При первой попытке аутентификации в системе пользователь должен воспользоваться Google Authenticator и отсканировать выводимый программой QR-код. После этого мобильное приложение создаёт одноразовый пароль, который периодически обновляется. Для успешного входа в систему пользователь должен ввести как свой постоянный пароль, так и одноразовый пароль, действующий на данном промежутке времени.

Реализованная схема аутентификации является адаптивной. Это означает, что в зависимости от конкретных условий может быть использован как один, так и два фактора аутентификации. Так, если пользователь подключается к системе из корпоративной сети (считающейся более безопасной), ему не требуется вводить одноразовый пароль. Если же он подключается из любой другой сети, то применяются повышенные меры безопасности, и необходимо использовать оба фактора.

6.2 Беспарольная система аутентификации с использованием magic link

Кроме того, в ходе работы была реализована система беспарольной аутентификации с применением magic link. Для реализации системы использован программный продукт с открытым исходным кодом Keycloak.

Принцип работы magic link был рассмотрен в главе 2. В данном механизме аутентификации пользователь должен предоставить системе свой адрес электронной почты, на который ему будет отправлена одноразовая ссылка для аутентификации.

Был создан и протестирован программный модуль, с помощью которого для конкретного пользователя (группы пользователей) Keycloak можно установить аутентификацию по e-mail. Этому пользователю при входе в систему не нужно будет вводить пароль – только ввести адрес электронной почты в предоставленное поле и перейти по полученной ссылке.

Диаграмма классов созданного модуля показана на рисунке 6.1.

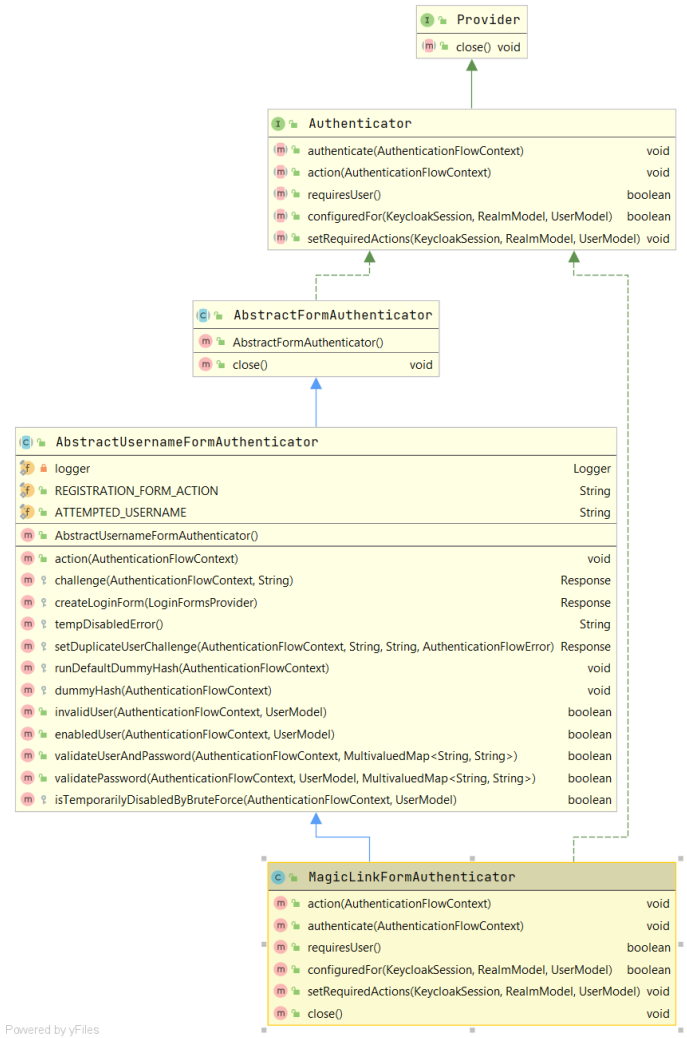

Рисунок 6.1 – Диаграмма классов модуля беспарольной аутентификации

Реализованные системы доступны по ссылкам [28,29].

По результатам проведённой работы была написана статья «Passwordless Authentication Using Magic Link Technology» [30], доклад по материалам которой представлен на 9-й международной конференции «Распределенные вычисления и грид-технологии в науке и образовании» (GRID’2021), проходившей 5–9 июля 2021 года в г. Дубне.

6.3 Децентрализованная система аутентификации

Аутентификация пользователей с помощью децентрализованных идентификаторов – достаточно новая технология. В отличие от

рассмотренных ранее методов (логин и пароль, биометрия, физические ключи и др.) на настоящий момент практически нет широко используемых систем, использующих данный метод аутентификации. Поэтому для более подробного изучения децентрализованной аутентификации выполнена практическая реализация системы, основанной на описанных принципах.

Для реализации системы была выбрана платформа Ethereum. Она является одной из самых распространённых криптовалют, которой уже заинтересовался ряд крупных компаний. Стоит отметить, что, в отличие от большинства аналогов, Ethereum разрабатывалась не только как средство платежей, но и как платформа для разработки различных децентрализованных онлайн-сервисов.

Для взаимодействия с Ethereum был использован криптокошелёк MetaMask. Он реализован как в виде мобильного приложения, так и в виде расширения для браузеров Google Chrome и Mozilla Firefox.

Для разработки системы были использованы язык программирования TypeScript и платформа Node.js.

В ходе разработки использованы материалы статьи А. Мартини «One-click Login with Blockchain» [31] с приведённым кодом, распространяющимся по открытой лицензии MIT.

Пользователь системы должен установить расширение MetaMask. Открыв расширение в первый раз, он должен придумать пароль, с помощью которого он в будущем сможет получить доступ к своему кошельку. Стоит отметить, что пароль является единым для всех сервисов, использующих DID, и хранится на стороне пользователя. Таким образом, отсутствует необходимость хранить и запоминать большое число паролей, а также риск утечки пароля, хранящегося на стороне сервера.

При входе в систему пользователь должен нажать на кнопку «Connect wallet» и при необходимости ввести пароль от кошелька. После одобрения

соответствующей транзакции пользователь войдёт в систему. В случае отказа система выведет сообщение о неудаче входа.

После успешного входа пользователь увидит окно, в котором показаны его имя пользователя и публичный адрес. Имя пользователя можно изменять по желанию, в то время как адрес остаётся неизменным, представляя собой уникальный идентификатор пользователя. Единственное ограничение – в качестве имени пользователя нельзя указать пустую строку.

Диаграмма действий пользователя показана на рисунке 6.2.

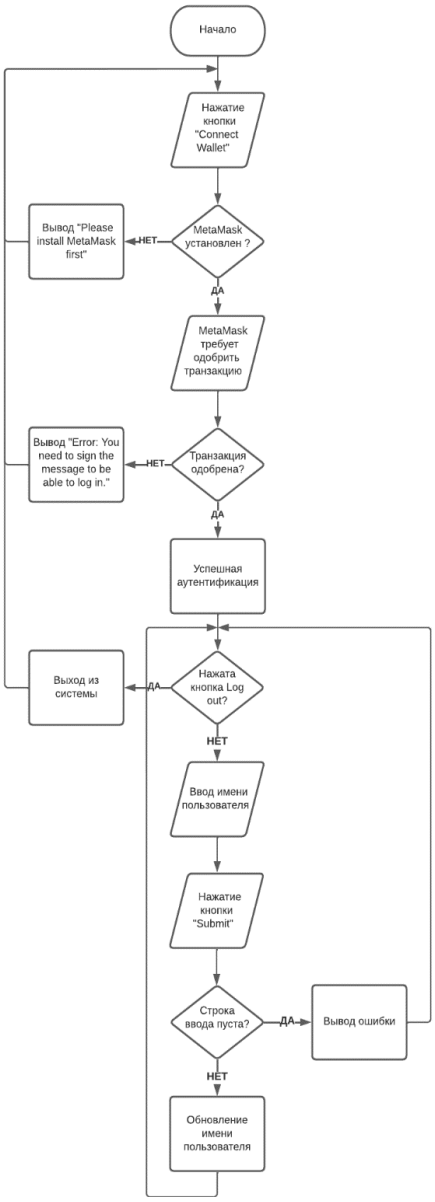

Рисунок 6.2 – Диаграмма действий пользователя системы аутентификации

Реализованный механизм аутентификации предполагается использовать для распределённых систем класса «peer-to-peer». В частности, рассматривается пример использования данной системы для аутентификации в программе-мессенджере.

Как было сказано ранее, в данном случае важным фактором будет обеспечение удобства использования, поскольку мессенджер предполагается для использования широким кругом пользователей. Соответственно, система аутентификации должна быть проста в освоении и использовании, не требуя дополнительных усилий вроде запоминания и частого ввода пароля или, тем более, ношения физических ключей.

Использование блокчейна наряду с криптокошельком MetaMask обеспечивает выполнение многих из этих требований. Так, ввод пароля требуется только при начале новой сессии браузера – в большинстве случаев для аутентификации необходимо только одобрить соответствующую транзакцию. Более того, один и тот же кошелёк может быть использован для аутентификации в разных системах – как и в случае OpenID, это может позволить вместо множества паролей для различных сервисов использовать только один (для криптокошелька). Сам процесс установки и взаимодействия с криптокошельком достаточно прост и понятен – MetaMask представляет из себя обычное расширение для браузера.

Интерфейс приложения сделан простым и минималистичным. Для успешной аутентификации пользователю достаточно нажать всего несколько кнопок (разумеется, при наличии установленного плагина MetaMask).

С другой стороны, использование MetaMask накладывает и некоторые ограничения. В настоящий момент данный криптокошелёк доступен только для браузеров Google Chrome и Mozilla Firefox – хотя оба входят в список наиболее распространённых браузеров, многие пользуются и альтернативными вариантами. Это может негативно повлиять на развёртываемость данного решения, наряду с его новизной.

Более подробный анализ с точки зрения фреймворка UDS представлен в следующей главе данной работы.

Интерфейс системы аутентификации показан на рисунках 6.3–6.4.

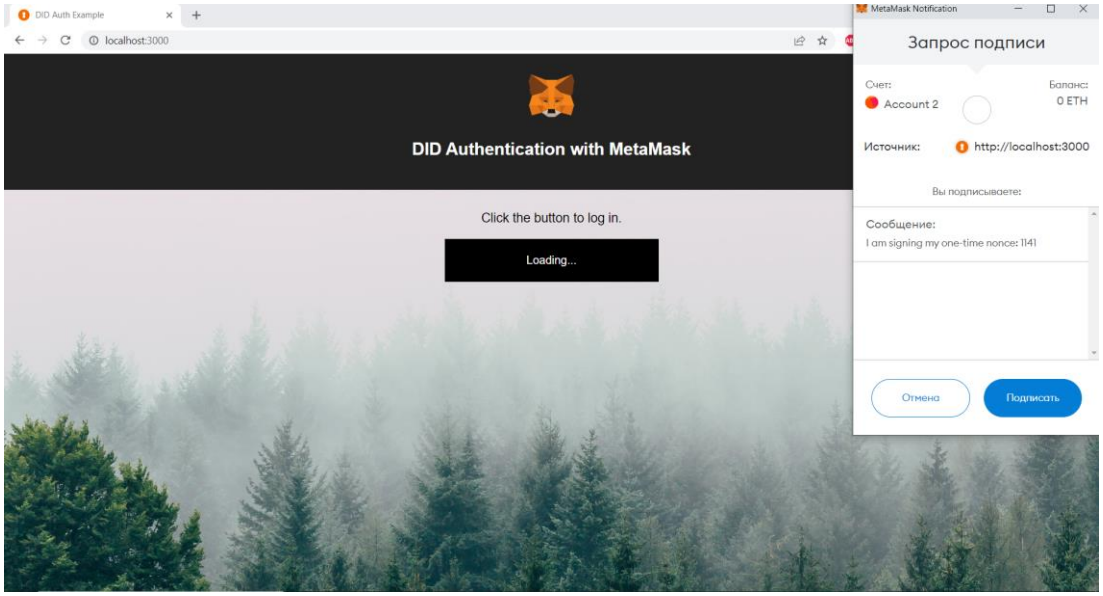

Рисунок 6.3 – Главная страница приложения и подключение криптокошелька

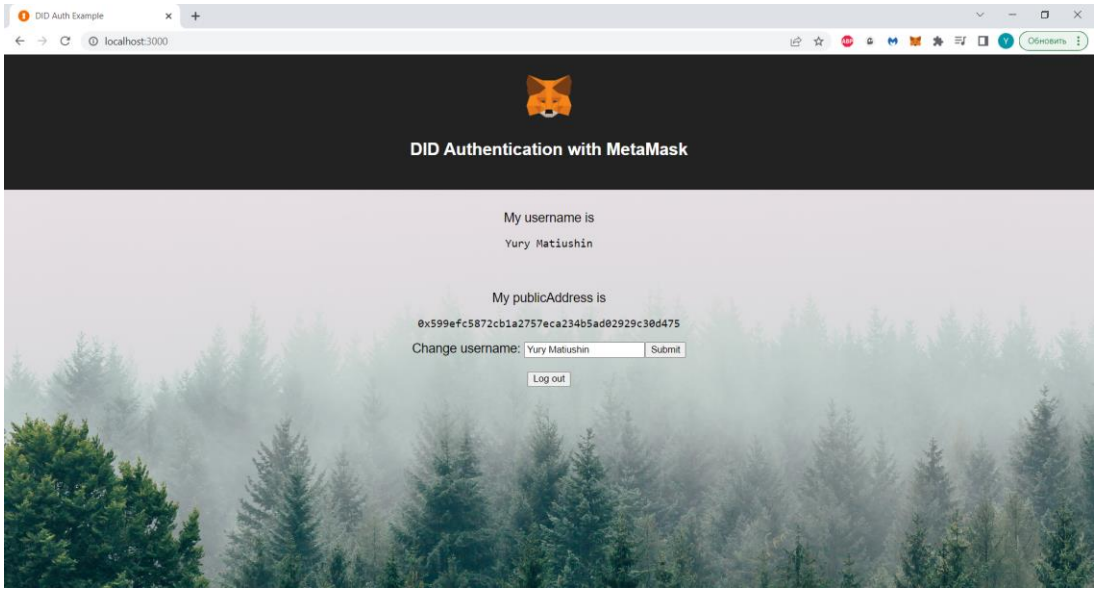

Рисунок 6.4 – Успешная аутентификация и изменение имени пользователя

6.4 Принцип работы системы децентрализованной аутентификации

Рассмотрим более подробно работу механизма аутентификации. В нём можно выделить шесть основных шагов.

Шаг 1 – изменение модели пользователя. В дополнение к имени пользователя также должны использоваться два новых поля – publicAddress и nonce. Первое означает уникальный адрес, ассоциируемый с пользователем в сети (идентификатор пользователя). Во второе записывается периодически изменяемая случайная строка, используемая для криптографического подтверждения личности пользователя. Например, можно использовать большое случайное число.

Шаг 2 – генерация случайной строки. В целях безопасности следует избегать использования функций, генерирующих псевдослучайные значения.

Шаг 3 – получение случайной строки. Когда пользователь нажимает на кнопку «Connect Wallet» и начинает процесс аутентификации, система отправляет запрос к хранилищу, чтобы получить случайную строку, ассоциируемую с данным пользователем. Если такой строки в системе нет, то система создаёт нового пользователя с соответствующим публичным адресом и генерирует новую случайную строку.

Шаг 4 – пользователь подписывает строку. Для этого используется функция web3.personal.sign(nonce, web3.eth.coinbase, callback). Плагин MetaMask выводит окно подтверждения, в котором пользователю предлагается одобрить транзакцию, в ходе которой он использует закрытый ключ своего аккаунта в блокчейне, чтобы подписать соответствующую случайную строку.

Шаг 5 – подтверждение подписи. При наличии случайной строки, публичного адреса пользователя и подписанной строки система может криптографически подтвердить верность подписи. Если подпись

подтверждена, подтверждается и тот факт, что пользователь является обладателем закрытого ключа, соответствующего ассоциированному с ним открытому ключу. Таким образом, личность пользователя подтверждается, и происходит успешная аутентификация.

Шаг 6 – изменение случайной строки. После каждой попытки входа в систему – успешной или нет – для соответствующего пользователя генерируется новая случайная строка. Таким образом, даже если в руки злоумышленника попадут данные, использованные при входе, он не сможет использовать их для получения доступа к системе.

Код программы расположен в репозитории GitHub [32].

В данной главе рассмотрена практическая реализация систем аутентификации. Была реализована система двухфакторной адаптивной аутентификации на основе одноразовых паролей с использованием системы SSO Keycloak и приложения Google Authenticator, а также система беспарольной аутентификации на основе magic link с использованием Keycloak. Реализована на практике децентрализованная система аутентификации.

Глава 7. Анализ результатов

7.1 Анализ децентрализованной системы по методике трёх критериев

Рассмотрим реализованную децентрализованную систему аутентификации с использованием фреймворка UDS по критериям безопасности, удобства применения и развёртываемости.

Система обладает частичной устойчивостью к физическому наблюдению – если в данной сессии браузера не был произведён вход в криптокошелёк MetaMask, пользователь должен ввести пароль. Система устойчива к имперсонации, брутфорс-атакам и внутреннему наблюдению. Также система устойчива к утечкам со стороны других сервисов (единственный пароль хранится на стороне пользователя и не передаётся по сети). Фишинг в системах блокчейн является потенциальной уязвимостью. Кража физического ключа не является актуальной угрозой ввиду отсутствия такового, а отсутствие третьей стороны и невозможность связать различные аутентификаторы одного и того же пользователя являются стандартными чертами блокчейна. Наконец, система запрашивает явное согласие со стороны пользователя [33].

Система частично выполняет условие отсутствия запоминаемой информации (необходимо помнить или хранить единственный пароль). Являясь децентрализованной, она по определению масштабируема со стороны пользователя. Необходимость ношения физического ключа отсутствует, процесс аутентификации достаточно прост (в худшем случае пользователь должен открыть кошелёк и единожды ввести пароль). Систему просто освоить, но на быстроту использования негативно влияет необходимость установки дополнительного плагина и создания криптокошелька. Количество ошибок при входе должно быть малым – нужно запомнить и ввести только один

пароль. Не выполняется условие простоты восстановления – потеря пароля от криптокошелька предотвращает дальнейший доступ к нему, и пользователь должен завести новый [34].

Выполняется условие малой цены за одного пользователя (не требуется использование каких-либо физических устройств). Не выполнены условия совместимости с сервером и совместимости с браузерами (пользователь должен пользоваться конкретным браузером и установить на него плагин). Также не выполнено условие наличия истории применения – метод является достаточно новым, хотя несколько экспериментальных реализаций уже существует. С другой стороны, выполнены условия открытого доступа (исходный код находится в открытом доступе и является бесплатным) и доступности (на уровне использования логина и пароля).

Проведём сравнительный анализ с применением фреймворка UDS, используя для сравнения рассмотренные ране системы аутентификации. Для этого дополним полученные в 3 главе таблицы, добавив к ним реализованный метод.

Стоит отметить, что проведённое сравнение касается в первую очередь реализованной в рамках данной работы системы. Многие из перечисленных характеристик являются свойствами блокчейн-систем в целом и, следовательно, будут относиться и к другим реализациям этого же метода аутентификации; другие особенности (например, ограниченный список поддерживаемых браузеров, что отрицательно сказывается на развёртываемости) могут не распространяться на другие реализации.

Результаты сравнения представлены в таблицах 7.1–7.3.

Таблица 7.1. Соответствие методов критерию безопасности

Метод/критерий	Физическое наблюдение	Имперсонация	Брутфорс-атаки	Внутреннее наблюдение	Утечки от др. сервисов	Фишинг-атаки	Кража ключа	Третья сторона	Явное согласие	Связь аутентификаторов
	1	2	3	4	5	6	7	8	9	10
Логин и пароль	0	0,5	0	0	0	0	1	1	1	1
Биометрия (отпечаток пальца)	1	0	1	0	0	0	0	1	1	0
Физический ключ	1	1	1	1	1	1	1	0	1	1
2FA (пароль + OTP)	1	1	1	0,5	1	1	0,5	0	1	1
Magic link	0,5	0,5	0,5	0	1	0	1	0	1	0
OpenID	0,5	0,5	0,5	0	1	0	1	0	1	0
DID	0,5	1	1	1	1	0	1	1	1	1

Таблица 7.2. Соответствие методов критерию удобства использования

Метод/критерий	Запоминание	Масштабируемость	Ношение физ. ключа	Физ. лёгкость	Простота освоения	Быстрота использования	Ошибки при входе	Восстановление
	1	2	3	4	5	6	7	8
Логин и пароль	0	0	1	0	1	1	0,5	1
Биометрия (отпечаток пальца)	1	1	1	0,5	1	0,5	0	0
Физический ключ	0	0	0	0	1	0,5	0,5	0
2FA (пароль + OTP)	1	1	0,5	0	1	0	0,5	0,5
Magic link	0,5	1	1	0,5	0,5	1	1	0,5
OpenID	0,5	1	1	0,5	0,5	1	1	1
DID	0,5	1	1	1	1	0,5	1	0

Таблица 7.3. Соответствие методов критерию развёртываемости

Метод/критерий	Малая цена	Сервер	Браузер	История	Открытый доступ	Доступность
	1	2	3	4	5	6
Логин и пароль	1	1	1	1	1	1
Биометрия (отпечаток пальца)	0	0	0	0,5	0	0,5
Физический ключ	0	0	1	1	0	0
2FA (пароль + OTP)	0	0	1	1	1	0,5
Magic link	1	0	1	0,5	1	1
OpenID	1	0	1	1	1	1
DID	1	0	0	0	1	1

Как и ранее, можно провести первоначальное сравнение, считая все веса равными 1.

Результаты сравнения по критериям показаны на рисунках 7.1–7.4.

Рисунок 7.1 – Сравнение методов по критерию безопасности

Рисунок 7.2 – Сравнение методов по критерию удобства использования

Рисунок 7.3 – Сравнение методов по критерию развёртываемости

Рисунок 7.4 – Сравнение методов по трём критериям

Как можно видеть, реализованный метод показывает высокие данные в области безопасности и достаточно высокие – в области удобства

использования. В области развёртываемости результаты средние, что объясняется как новизной метода, так и некоторыми техническими особенностями.

7.2 Сравнение системы с существующими аналогами

Попробуем теперь провести анализ децентрализованной системы аутентификации в приложении к конкретному сценарию. В частности, рассмотрим вариант мессенджера, уже описанный ранее.

Расстановка весов останется прежней, а выполнение аспектов того или иного критерия обосновано в предыдущем пункте. Здесь приведём результаты сравнения с другими методами в рамках рассматриваемого сценария.

Результаты сравнения по критериям показаны на рисунках 7.5–7.8.

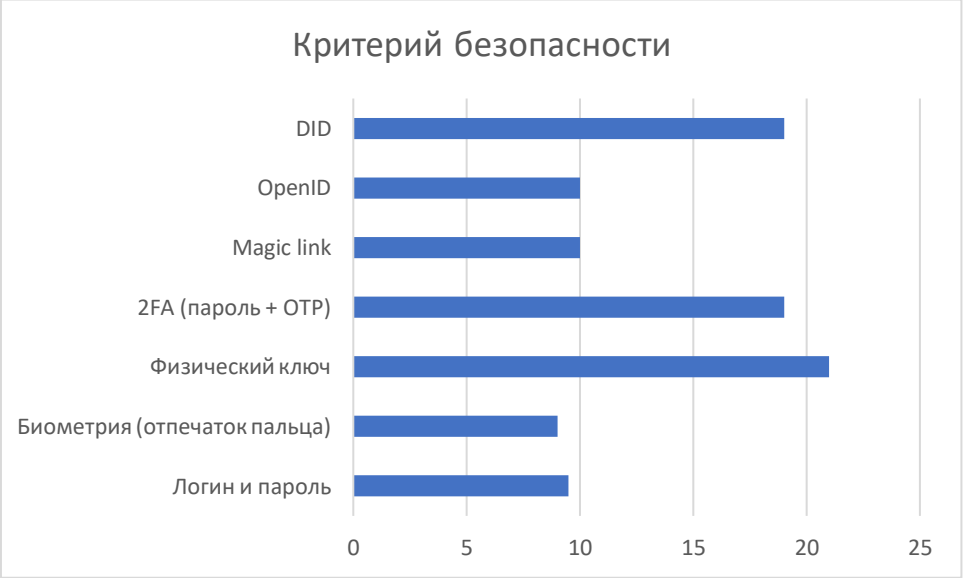

Рисунок 7.5 – Сравнение методов по критерию безопасности

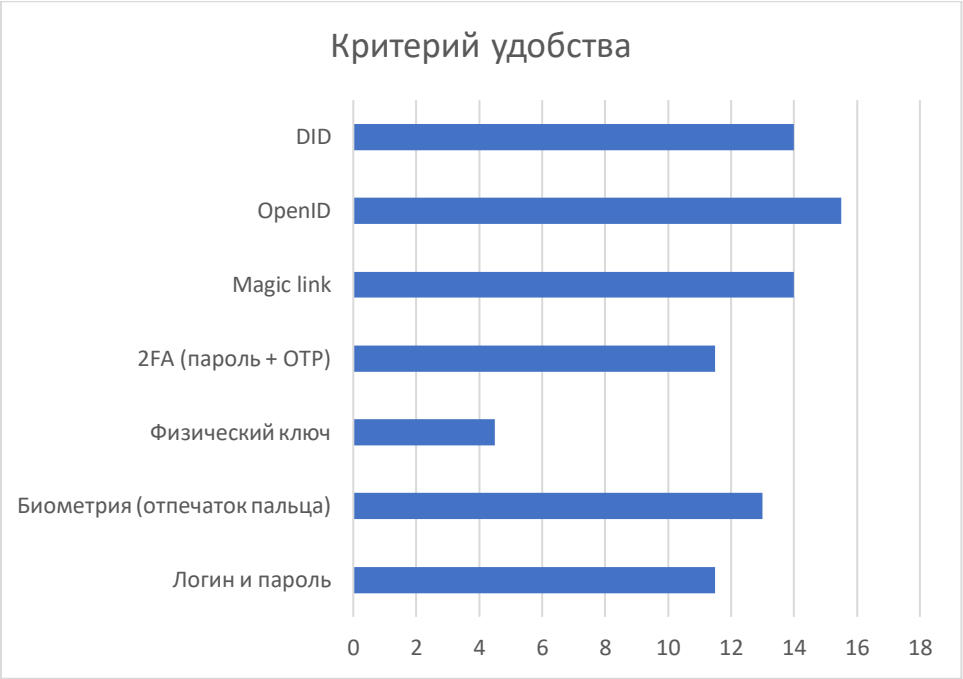

Рисунок 7.6 – Сравнение методов по критерию удобства использования

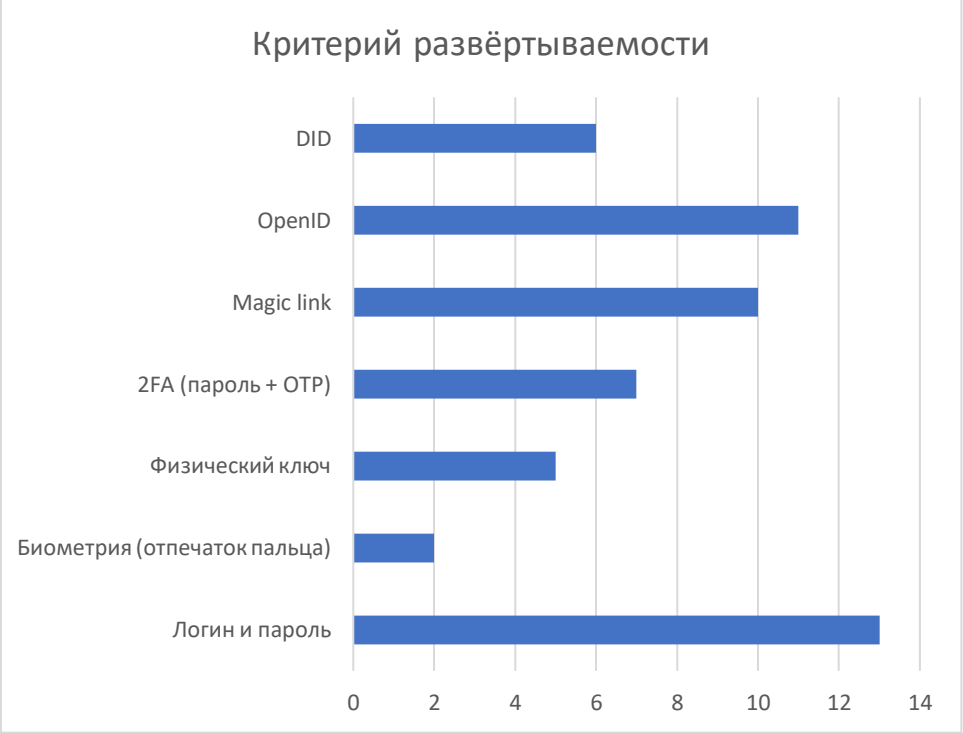

Рисунок 7.7 – Сравнение методов по критерию развёртываемости

Рисунок 7.8 – Сравнение методов по трём критериям

В целом метод сохранил свои позиции и в данном сценарии. Он по-прежнему даёт показывает высокие результаты в областях безопасности и удобства использования, хотя он и не так удобен, как OpenID. Развёртываемость также остаётся на среднем уровне в силу тех же причин, что и в предыдущем пункте.

7.3 Дальнейшие направления работы

Одним из недостатков реализованной децентрализованной системы является её зависимость от конкретного браузера. Криптокошелёк MetaMask, используемый в системе, совместим только с браузерами Google Chrome и Mozilla Firefox, что неудобно для пользователей, предпочитающих другие браузеры (Opera, Edge, Yandex и т. д.) В дальнейшем возможен переход на кошелёк, поддерживающий более широкий круг современных браузеров, что

также обеспечит частичное соответствие одному из преимуществ критерия удобства в использованном нами фреймворке.

Также в качестве недостатка можно рассмотреть саму необходимость устанавливать дополнительное ПО (плагин криптокошелька). Стоит рассмотреть возможность создания системы, в которой взаимодействие с блокчейном целиком ляжет на плечи программы, а пользователю нужно только дать согласие и обеспечить безопасность мастер-ключа, дающего доступ к кошельку.

Наконец, следует отметить, что система требует от пользователя хранить или запомнить текстовый пароль. Это отражается как на удобстве использования, так и на безопасности – в случае утечки пароля злоумышленник может получить доступ к кошельку, а следовательно, и к защищённой системе.

С другой стороны, от пользователя требуется хранение только одного пароля, вне зависимости от количества аккаунтов. Более того, в отличие от многих других систем, пароль хранится только на стороне пользователя – таким образом, утечка со стороны сервиса не приведёт к возможности получения злоумышленником доступа к системе от имени пользователя. В этой области также возможны некоторые усовершенствования – например, использование кодовой фразы из 12–24 слов в случае забытого или утерянного пароля. Подобный механизм уже применяется в некоторых криптокошельках.

В данной главе проведён анализ реализованной децентрализованной системы аутентификации. Рассмотрены её характеристики в рамках фреймворка UDS (безопасность, удобство применения и развёртываемость). Проведён сравнительный анализ с наиболее распространёнными на настоящий момент методами аутентификации. Определены варианты дальнейших направлений работы в области аутентификации с применением децентрализованных идентификаторов.

Выводы

В ходе данной работы проведено исследование различных методов, используемых для аутентификации пользователей в распределённых системах. Были рассмотрены основные понятия, связанные с темой аутентификации, и определены факторы, на которых основаны наиболее распространённые методы. Также был проведён обзор нескольких применяющихся на настоящий момент механизмов аутентификации.

Далее была использована методика для сравнения различных методов, основанная на трёх важнейших критериях – безопасности, удобстве использования и развёртываемости. Был продемонстрирован наиболее простой вариант сравнения методов при предположении о равноценности всех составляющих частей каждого критерия.

После этого были рассмотрены несколько сценариев, в которых требуется система аутентификации пользователей. Были выявлены наиболее важные в каждом случае критерии, а также проведён сравнительный анализ методов аутентификации в приложении к конкретным задачам.

Отдельно подвергся рассмотрению новый метод аутентификации, основанный на использовании децентрализованных идентификаторов. Рассмотрено понятие суверенной идентичности, её истоки и необходимость.

Была проведена практическая реализация систем аутентификации, в том числе двухфакторной, беспарольной и децентрализованной аутентификации. Результаты работы представлены в докладе на конференции GRID, по ним опубликована научная статья.

По результатам реализации метода децентрализованной аутентификации проведён его анализ по рассмотренной ранее методике, выявлены сильные и слабые стороны, определены сценарии, в которых он может показать себя с хорошей стороны.

В целом можно сделать вывод о том, что сравнение различных методов аутентификации, основанных на разных факторах и принципах – сложная и многогранная задача. В данной работе предложен один из вариантов решения данной задачи, предполагающий выделение наиболее важных критериев, разбиение каждого из них на некоторое количество аспектов, и присваивание «оценок», исходя из наличия у конкретного метода того или иного преимущества, а также о важности данного преимущества в рассматриваемом сценарии. Таким образом, хотя не представляется возможным однозначно заявить о наличии «самого лучшего» метода аутентификации на настоящий момент, можно определить, какой метод будет лучше подходить в условиях конкретного сценария и конкретной задачи.

Заключение

В рамках данной работы получены следующие результаты:

1. Проведено исследование проблемы аутентификации пользователей в распределённых вычислительных системах. Рассмотрены основные методы, применяемые в настоящий момент, принципы их работы, достоинства и недостатки.
2. Проведено сравнение методов, применяемых для аутентификации пользователей. Определены основные критерии сравнения, использован фреймворк, учитывающий наиболее важные критерии.
3. Определены достоинства и недостатки рассмотренных методов в различных условиях. Рассмотрено несколько различных сценариев аутентификации пользователей с примерами конкретных кейсов.
4. Реализованы на практике системы аутентификации пользователей. Реализованная децентрализованная система сравнена с уже существующими по основным критериям, в том числе и в приложении к конкретной задаче.

Все задачи, сформулированные перед началом работы, были выполнены в полном объёме. Поставленные цели в ходе работы были достигнуты.

Список литературы

1. Verizon 2022 Data Breach Investigations Report [Электронный ресурс] // URL: https://www.verizon.com/business/resources/reports/dbir/2022/master-guide/ (дата обращения: 25.05.2022)
2. Widup, Suzanne & Pinto, Alex & Hylender, David & Bassett, Gabriel & Langlois, Philippe. (2021). 2021 Verizon Data Breach Investigations Report.
3. Шаньгин В. Ф. Защита компьютерной информации. Эффективные методы и средства / В. Ф. Шаньгин – М: ДМК Пресс, 2010. – 544 с.
4. Гольдштейн Б. С., Елагин В. С., Сенченко Ю. Л. Протоколы AAA: RADIUS и Diameter. Серия «Телекоммуникационные протоколы». Книга 9 / Б. С. Гольдштейн, В. С. Елагин, Ю. Л. Сенченко – СПб.: БХВ-Петербург, 2011. – 352 с.
5. Аутентификация. Теория и практика обеспечения безопасного доступа к информационным ресурсам: учебное пособие для вузов / А. А. Афанасьев, Л. Т. Веденьев, А. А. Воронцов и др. – М.: Горячая линия – Телеком, 2012. – 552 с.
6. Сидоркина И. Г., Канаев Р. В., Меркушев О. Ю. Классификация методов аутентификации человека // Вестник ВУиТ. 2009. №12. URL: https://cyberleninka.ru/article/n/klassifikatsiya-metodov-autentifikatsii-cheloveka (дата обращения: 16.04.2022).
7. Шнайер Брюс. Прикладная криптография. Протоколы, алгоритмы и исходный код на С // Брюс Шнайер – М.: Вильямс, 2016. – 1024 с.
8. Künnemann, Robert & Steel, Graham. (2012). YubiSecure? Formal Security Analysis Results for the Yubikey and YubiHSM. 10.1007/978-3-642-38004-4_17.

9. Барашко, Е. Н. Сравнительный анализ основных подходов к реализации систем идентификации по отпечаткам пальцев / Е. Н. Барашко, М. К. Пижевский // Modern Science. – 2019. – № 11–4. – С. 240–243. – EDN WAALZD.
10. Юрьев Д. Р., Рогова О. С. Сравнительный анализ двухфакторной аутентификации – Владивосток: Технические науки – от теории к практике, 2017, №6 – С. 46–51
11. Подгаев, А. Г. Проблемы безопасности систем двухфакторной аутентификации / А. Г. Подгаев, А. А. Подгаев // Ученые заметки ТОГУ. – 2019. – Т. 10. – № 4. – С. 256–260.
12. Magic Link [Электронный ресурс] // URL: https://magic.link/ (дата обращения: 18.03.2021)
13. Magic Link Authentication – Choose the right path [Электронный ресурс] // URL: https://medium.com/authenticate/magic-link-authentication-choose-the-right-path-f0c351be6ac (дата обращения: 18.03.2021)
14. Аутентификация OpenID Connect с помощью Azure Active Directory [Электронный ресурс] // URL: https://docs.microsoft.com/ru-ru/azure/active-directory/fundamentals/auth-oidc (дата обращения: 24.05.2021)
15. Velásquez, I., Caro, A., & Rodríguez, A. (2017). Identifying Comparison and Selection Criteria for Authentication Schemes and Methods.
16. Velásquez, Ignacio & Caro, Angelica & Rodríguez, Alfonso. (2019). Multifactor Authentication Methods: A Framework for Their Comparison and Selection.
17. Bonneau, J., Herley, C., Oorschot, P. C. van, & Stajano, F. (2012). The Quest to Replace Passwords: A Framework for Comparative Evaluation of Web Authentication Schemes. 2012 IEEE Symposium on Security and Privacy.

18. Helkala, Kirsi & Snekkenes, E. (2008). A method for ranking authentication products. 80–93.
19. Tseng, Thomas & Lee, R. & Lin, Shih-Wei & Han, T. (2006). Mixed Client Server and Peer to Peer System for Internet Content Providers. 3336–3341. 10.1109/ICSMC.2006.384633.
20. Purkayastha, Saptarshi & Goyal, Shreya & Oluwalade, Bolu & Wu, Huanmei & Zou, Xukai. (2021). Usability and Security of Different Authentication Methods for an Electronic Health Records System.
21. Cutting, Daniel & Landfeldt, Björn & Quigley, Aaron. (2006). Implicit Group Messaging over Peer-to-Peer Networks. 125–132. 10.1109/P2P.2006.20.
22. R.Kalai Selvi, and V. Kavitha. "Authentication in Grid Security Infrastructure-Survey" Procedia Engineering, vol. 38, 2012. doi:10.1016/j.proeng.2012.06.461
23. Sovrin™: A Protocol and Token for Self-Sovereign Identity and Decentralized Trust [Электронный ресурс] // URL: https://sovrin.org/wp-content/uploads/Sovrin-Protocol-and-Token-White-Paper.pdf (дата обращения: 16.02.2022)
24. Sovrin: What Goes on the Ledger? [Электронный ресурс] // URL: https://www.evernym.com/wp-content/uploads/2017/07/What-Goes-On-The-Ledger.pdf (дата обращения: 16.02.2022)
25. Decentralized Identifiers (DIDs) v1.0 [Электронный ресурс] // URL: https://www.w3.org/TR/did-core/ (дата обращения: 24.09.2020)
26. Introduction to DID Auth [Электронный ресурс] // URL: https://github.com/WebOfTrustInfo/rwot6-santabarbara/blob/master/final-documents/did-auth.md (дата обращения: 25.05.2022)
27. Willis, Nathan. FreeOTP multi-factor authentication // URL: https://lwn.net/Articles/581086/ (дата обращения: 02.10.2020)

28. Vue-Keycloak [Электронный ресурс] // URL: https://github.com/YuriyM-SPB/Vue-Keycloak (дата обращения: 26.05.2022)
29. Keycloak-magic-link [Электронный ресурс] // URL: https://github.com/YuriyM-SPB/Keycloak-magic-link (дата обращения: 26.05.2022)
30. Matiushin I., Korhkov V. Passwordless Authentication Using Magic Link Technology // Proceedings of the 9th International Conference "Distributed Computing and Grid Technologies in Science and Education"(GRID'2021), Dubna, Russia, 2021. P. 434–438.
31. One-click Login with Blockchain [Электронный ресурс] // URL: https://www.toptal.com/ethereum/one-click-login-flows-a-metamask-tutorial (дата обращения: 10.10.2021)
32. DID-MetaMask [Электронный ресурс] // URL: https://github.com/YuriyM-SPB/DID-MetaMask (дата обращения: 26.05.2022)
33. Yuxin Zhong, Mi Zhou, Jiangnan Li, Jiahui Chen, Yan Liu, Yun Zhao, Muchuang Hu, "Distributed Blockchain-Based Authentication and Authorization Protocol for Smart Grid", Wireless Communications and Mobile Computing, vol. 2021, Article ID 5560621, 15 pages, 2021. https://doi.org/10.1155/2021/5560621
34. Moniruzzaman, Md & Chowdhury, Farida & Ferdous, Md. Sadek. (2020). Examining Usability Issues in Blockchain-Based Cryptocurrency Wallets.