ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ТЕЛЕКОММУНИКАЦИИ
INFORMATION TECHNOLOGY AND TELECOMMUNICATIONS

УДК 004.75
DOI: 10.21822/2073-6185-2023-50-3-132-141 Оригинальная статья /Original article

Разработка клиентских приложений на блокчейн

О.Д. Окладникова, А. В. Буков, Н. Н. Жуков
Национальный исследовательский университет ИТМО,
197101, г. Санкт-Петербург, Кронверкский пр., 49, Россия

Резюме. Цель. В статье рассмотрены вопросы, связанные с процессом разработки клиентских Web-приложений для IT-проектов, в основе которых заложены принципы технологии децентрализованных сетей - блокчейн. Исследуются практические способы реализации процесса авторизации пользователя и его взаимодействия с сетью блокчейн через клиентское приложение. **Метод.** В качестве инструмента исследования было выбрано браузерное расширение МетаМаск, позволяющее разрабатывать, тестировать и запускать собственные программные модули. **Результат.** Проведён анализ инструментов МетаМаск, рассмотрены процессы связанные с началом работы как с точки зрения разработчика, так и с точки зрения пользователя кошелька. Понимание организации данных процессов позволит сформировать у пользователей необходимые знания в области разработки клиентских приложений, построенных на блокчейн. **Вывод.** Разработанные шаблоны программных кодов могут быть использованы в качестве типовых при разработке клиентских приложений в децентрализованной сети Ethereum. Преимущество предложенных решений заключается в использовании простых программных конструкций, позволяющих сформировать базовые принципы.

Ключевые слова: блокчейн, майнер, Ethereum, Web-приложение, авторизация, криптокошелек, криптовалюта, МетаМаск, информационная безопасность.

Для цитирования: О.Д. Окладникова, А.В. Буков, Н.Н. Жуков. Разработка клиентских приложений на блокчейн. Вестник Дагестанского государственного технического университета. Технические науки. 2023; 50(3):132-141. DOI:10.21822/2073-6185-2023-50-3-132-141

Development of client applications on Blockchain

O.D. Okladnikova, A.V. Bukov, N.N. Zhukov
National Research University ITMO,
49 Kronverksky Ave., St. Petersburg 197101, Russia

Abstract. Objective. The article discusses issues related to the process of developing client Web applications for IT projects, which are based on the principles of decentralized network technology - blockchain. The authors explore practical ways to implement the user authorization process and its interaction with the blockchain network through a client application. Method. The MetaMask browser extension was chosen as a research tool, which allows you to develop, test and run your own software modules. **Result.** In the course of the study, the analysis of MetaMask tools was carried out, the processes associated with the start of work were considered both from the point of view of the developer and from the point of view of the wallet user. Understanding the organization of these processes will allow users to form the necessary knowledge in the development of client applications built on blockchain. **Conclusion.** The program code templates developed by the authors can be used as standard ones in the development of client applications in the decentralized Ethereum network. The advantage of the proposed solutions lies in the use of simple software structures that allow the formation of basic principles.

Keywords: blockchain, miner, Ethereum, Web application, authorization, crypto wallet, cryptocurrency, MetaMask, information security.

For citation: O.D. Okladnikova, A.V. Bukov, N.N. Zhukov. Development of client applications on Blockchain. Herald of Daghestan State Technical University. Technical Science. 2023; 50(3):132-141. DOI:10.21822/2073-6185-2023-50-3-132-141

Введение. К одной из наиболее динамичной и постоянно развивающейся сферой в IT–индустрии относится разработка информационных систем и Web – приложений на основе баз данных распределённого типа (децентрализованного реестра данных – блокчейн технологии). Основная концепция блокчейн – это прозрачность распределения данных, реализация которой в цифровых решениях повышает доверие между участниками обмена данными. Согласно исследованию, проведенному Cambridge Centre For Alternative Finance [1], крупнейшим пользователем технологии распределенного реестра является финансовый сектор, т.к. блокчейн технология обладает прорывным инновационным потенциалом в сфере платежей, клиринга, расчетов и других операций. Использование блокчейн изменяет систему обслуживания и хранения активов, урегулирования обязательств, исполнения контрактов и управления рисками.

Например, Ripple – блокчейн-проект, запущенный в 2012 году и работающий на базе сети XRP Ledger является конкурентом централизованной системы Swift. Преимущество Ripple заключается в высокой скорости производства блока (в среднем за 3 - 4 сек.) и низкой комиссии за переводы (около 0.00001 XRP или $0.0000053) [2]. Для сравнения в Swift международные платежи проводятся от 3 до 5 дней, а стоимость за транзакции зависит от банка отправителя, банков посредников и банка получателя [3].

Первыми сетевыми платформами были Bitcoin – платформа для хранения данных о перемещении цифровых денежных ресурсов между пользователями, и Ethereum – платформа для создания «честных» сделок (смарт контрактов) [4]. На основе блокчейн создаются децентрализованные банки, например, Cecabank, Grant Thornton, основной деятельностью которых является поддержка бизнеса, страховых компаний и государственных программ [5]. Одной из проблем цифровизации остается защита данных. Развитие онлайн услуг в государственном и финансовом секторах, телекоммуникационных компаниях, медицинских учреждениях и др., а также активное использование пользователей социальных сетей увеличивает рост утечек конфиденциальной информации.

В аналитическом отчете InfoWatch, опубликованном в апреле 2023 года, объем утечек персональных данных в России в 2022 году составил 667 млн. записей, что почти в 2,7 раза больше, чем в 2021 году. Каждая вторая утечка данных в России происходила в финансовой отрасли. В 2022 г. утечек данных из финансового сектора в России стало больше в 1,7 раза. При этом 90% составляют данные о финансовых операциях, персональные данные клиентов и сотрудников, материалы, классифицируемые как коммерческая тайна (инвестиционные планы, данные маркетинговых исследований, внутренняя закупочная информация и др.) [6]. Одна из причин утечки данных, по мнению экспертов, заключается в недостаточной проработанности архитектурного решения системы с точки зрения обеспечения защиты данных на клиентской стороне приложения.

Использование онлайн сервисов пользователем означает, в первую очередь, доверие своих данных другому лицу. Компании, в которых не заботятся в достаточной степени о безопасности данных, в результате чего происходят утечки, подвергаются как финансовым, так и репутационным потерям. В 2022 году крупные утечки информации произошли в сервисах «Яндекс.Еда» и «Яндекс. Практикум», в онлайн-кинотеатре Start, в сервисе экспресс-доставки СДЭК, сети электроники DNS, компании «Билайн». 15 ноября 2022 г. хакеры выставили на продажу в даркнете за $4200 данные 7,2 млн. пользователей российского сервиса аренды электросамокатов Whoosh. За 1 квартал 2023 года объем утечек данных вырос в 2,3 раза по сравнению с этим же периодом прошлого года. Крупнейшими утечками первого квартала стали: выложенные в открытый доступ данные бонусной программы «СберСпасибо» (суммарно 52,5 млн записей) и сети «Спортмастер» (46 млн записей), выставленная на продажу база интернет-аптеки zdravcity.ru (8,9 млн записей) [7].

Постановка задачи. В качестве решения проблемы по обеспечению безопасности данных в Интернет разработчики IT-продуктов все больше обращаются к технологии децентрализованных систем, построенных на основе блокчейн. Информация, записанная в цепочку блокчейн, имеет более высокий уровень защиты по сравнению централизованными системами за счёт шифрования данных непосредственно владельцем данных, а не сервисом, обеспечивающим обработку и защиту данных. Различные системы используют отличающиеся друг от друга способы и методы шифрования, которые напрямую влияют не только на степень защищённости данных, но и на скорость обработки запросов, так как слишком сложные методы шифрования требуют значительно больших временных и вычислительных ресурсов для совершения операций по кодированию и декодированию данных. В блокчейн системах с быстрым временем подтверждения транзакции в данный момент могут возникнуть ситуации, снижающие безопасность данных.

Созданием блоков занимаются майнеры - программа-пользователь блокчейн сети выполняющая вычисления, необходимые для формирования транзакций в блоки. Для того чтобы сформированный блок был принят в цепь необходимо распространить его между более чем 51% участников. Однако, в результате задержки распространения информации, связанной с интернет-соединением и децентрализованной архитектурой сети, может возникнуть ситуация, когда блок с этим же порядковым номером параллельно начнет обрабатываться другим майнером. В результате в цепочку блокчейн войдёт тот блок, который был обработан ранее пользователем с ресурсами большей вычислительной мощности. Блок, обработанный другим майнером станет «потерянным» (Uncle-Block) [8]. При систематически возникающих подобных «накладках» майнеры с меньшей мощностью вычислительных ресурсов будут многократно «проигрывать» майнерам с большой мощностью вычислительных ресурсов. Вследствие этого, майнеры с меньшей мощностью будут объединятся в крупные пулы (от англ. Pool - объединение, общность) и транзакции в блокчейн сетях с небольшим количеством участников будут обрабатываться на нескольких крупных объединениях, что приведет к централизации данных, и значительно снизит безопасность сети [9]. Так как данные хранятся не в одном месте, а распределяются между множеством устройств-клиентов, может возникнуть ситуация, когда одни устройства будут считать актуальными данные, полученные в результате обработки запроса от одного майнера, а другая часть устройств – от другого майнера. Например, пользователь «А» (покупатель) в рамках одной транзакции пополнил свой электронный кошелёк и проводит оплату в магазине. А пользователь «Б» (продавец) не видит в своей версии блокчейн сети информацию о том, что у пользователя «А» есть нужное количество денег. В данной ситуации сеть не может функционировать, так как между участниками сети отсутствует договорённость о свойствах тех или иных объектов.

Методы исследования. С целью устранения угроз, возникающих при описанной выше ситуации, был разработан алгоритм GHOST («Greedy Heaviest Observed Subtree»), который представляет собой сквозной протокол шифрования с заложенными в нем правилами выбора цепочки на основе ранее потерянных блоков и добавления их в основной блокчейн с частичным вознаграждением майнера. Такой подход повышает сложность атаки на сеть, т.к. майнер – победитель, не единственный, кто владеет вычислительной мощностью. Большее количество узлов сохраняет мощность и устраняет необходимость в централизованных пулах майнинга в более крупных цепочках [10].

Наибольшей уязвимостью с точки зрения информационной безопасности остается процесс авторизации пользователя, особенно в различных WEB-приложениях, например, в социальных сетях, онлайн-магазинах так как проверка аутентификация пользователя производится централизованно сервисом в отношении данных, размещенных на сервере. Например, разработанный функционал для восстановления забытого пароля требует от пользователя ввода личных данных (номера телефона, секретного вопроса или паспортных данных). Мошенники, завладевшие ими, могут получить нелегитимный доступ к системе.

Преимущество блокчейн заключается в том, что данные шифруются с помощью специальных алгоритмов и распределяются между всеми участниками сети, а не хранятся в едином датацентре, что делает невозможным их изменение. Т.к. по факту время расшифровки данных значительно превышает время жизни человека или системы. [11].

Понимание принципов работы блокчейн сети важно как для разработчиков систем, работающих на блокечейн платформе, так и для пользователей, совершающих те или иные действия в этих системах. Выбор правильного инструмента для работы даст пользователю уверенность в надёжности той или иной системы. Для разработчиков, в свою очередь, важно охватить как можно больший спектр клиентских систем, с которыми можно выстроить интеграцию, но также важно понимать как с точки зрения безопасности устроены не или иные клиентские системы.

В централизованных системах и сетях криптография используется для защиты хранилища данных или для их шифрования при передаче по незащищённым каналам связи, а так же направлена на то, чтобы избежать вмешательства третьих лиц к доступу и получению информации, с помощью которой можно идентифицировать личность человека, его электронную почту и др. Технологической основой блокчейн является хранение данных не на общих серверах (как в централизованных сетях), а распределение этих данных по миллионам компьютеров, находящихся в разных точках планеты. Безопасность обеспечивается за счет используемых криптографических методов и алгоритмов [12].

Появление блокчейн расширяет возможности цифровой экосистемы (смарт-контракты, криптовалюта и пр.), при этом сохраняются и угрозы информационной безопасности пользователей (мошенничество, хищение персональных данных, идентификаторов, капитала и пр.). Задача, поставленная авторами статьи – исследовать реализацию процесса авторизации пользователя и его взаимодействия с сетью блокчейн через клиентское приложение с точки зрения информационной безопасности.

В централизованных системах данные хранятся в базе данных или сервисе кэширования (отдельный узел, занимающийся хранением данных), размещенных на одном физическом узле и многократно копируются. В децентрализованных системах данные не копируются, а хранятся в блоках, распространяемых между узлами. Каждый узел системы доверяет только той информации, которая по общепринятому алгоритму считается корректной. Ответственность за процесс проверки информации переносится с одного регулирующего узла на все узлы сети. Отключение одного узла не ведёт к потере данных и при первом (или повторном) подключении узел получает необходимый для работы набор блоков. Такой подход к хранению данных повышает устойчивость системы, однако может оказать влияние на снижение скорости обработки входящих запросов (транзакций), так как для поиска нужных данных необходимо последовательно обработать большое количество блоков, в которых записаны трансформации данных, и сформировать итоговое состояние. Снижение скорости обработки транзакций особенно критично для сферы финансовых услуг, например, при проведении различных платежных операций, таких как онлайн или безналичный расчет.

Обработка денежных операций в блокчейн сетях происходит с помощью токенов. Токены представляют собой запись в регистре, распределенную в блокчейн-цепочке, т.е. по сути это цифровая версия криптовалюты, используемая для совершения онлайн покупок, передачи, а так же оплаты комиссии за проведение тех или иных операций. В каждой блокчейн сети существует своя криптовалюта, как правило название котрой совпадает с названием самой сети (Ethereum, Bitcoin, ZCash и др.). Монета одной сети не может быть использована в другой, но может быть обменена с учётом курса на специальных крипто-биржах. Взаимодействие пользователя с блокчейн сетью осуществляется на основе программного интерфейса – криптокошелька. В настоящее время одним из популярных криптокошельков в мире является МетаМаск, представляющий собой браузерное расширение (или его мобильное приложение для устройств с iOS и Android) с открытым исходным кодом. Кошелек МетаМаск предназначен для хранения и перевода криптовалюты и NFT, созданных в экомистеме Ethereum и других

совместимых с ним сетях (например, Binance Smart Chain, Polygon Network (Matic), Optimism Ethereum и др.). Кроме собственной валюты Ethereum, MetaMask также может работать с токенами, построенными на стандартах ERC20, BEP20 и BEP2. Расширение МетаМаск можно подключить к децентрализованным сервисам (DeFi): обменникам, биржам и пулам ликвидности, а также без регистрации и идентификации работать с такими площадками как Uniswap и PancakeSwap [13]. Браузерное расширение МетаМаск совместимо со всеми распространенными на сегодняшний день браузерами: Chrome, Firefox, Brave, Microsoft Edge, Яндекс, Opera и др. Источник получения расширения, например, для chrome-браузеров (Яндекс, Опера, Edge) – это «интернет-магазин Chrome» [14], а для Firefox – официальный магазин расширений «Firefox Add-Ons» [15].

Отличительная особенность MetaMask заключается в том, что сервис не хранит никакой информации о пользователе: ни адрес электронной почты, ни пароль, ни мастер-ключ (Secret Recovery Phrase – секретная фраза для восстановления), ни приватные ключи. То есть пользователь полностью владеет своей крипто-идентичностью. После установки расширение MetaMask появляется в списке установленных расширений браузера и становится доступным. Для работы с расширением пользователь должен создать кошелек.

С целью обеспечения конфиденциальности и безопасности кошелька и учетных записей используются: пароль и Secret Recovery Phrase, задаваемые пользователем поэтапно при создании кошелька, а также приватные ключи. Рассмотрим особенности перечисленных инструментов. Пароль задается пользователем с учетом ограниченной сервисом минимальной длины – не менее 8 символов. В качестве символов пароля могут быть использованы цифры, буквы верхнего и нижнего регистров, знаки препинания. Эти требования повышают криптостойкость пароля при машинном способе взлома аккаунта. Мастер-ключ представляет собой уникальную фразу, состоящую из 12 слов, которая генерируется при первой настройке MetaMask. MetaMask локально шифрует Secret Recovery Phrase с помощью введенного ранее пароля. В случае, если пользователь заблокирует свой кошелек, никто не сможет использовать находящиеся в нем средства, пока не будет введен пароль. В случае, если пользователь забудет пароль, то для восстановления доступа к аккаунту необходима секретная фраза, которая известна только владельцу. В случае утери мастер-ключа, никто (включая команду MetaMask) не может изменить или восстановить Secret Recovery Phrase. Это связано с особенностью децентрализованного устройства сети – анонимностью ее участников. При регистрации пользователь не вводит свои персональные данные (паспортные данные, номер телефона, фамилию, имя и пр.), следовательно отсутствует необходимость их централизованного хранения в сети. Адрес кошелька, идентифицирующий его владельца, представляет собой публичный код из 40-44 символов, сгенерированный на основе Secret Recovery Phrase.

Особенность генерации публичного кода заключается в том, что шифрование происходит в одну сторону, т.е. даже зная итоговый набор символов невозможно установить из какой последовательности он был получен. Это свойство позволяет использовать публичный код для идентификации пользователей в рамках производимых транзакций. Рассмотрим на примере процесс перевода и покупки токенов в MetaMask (рис. 1).

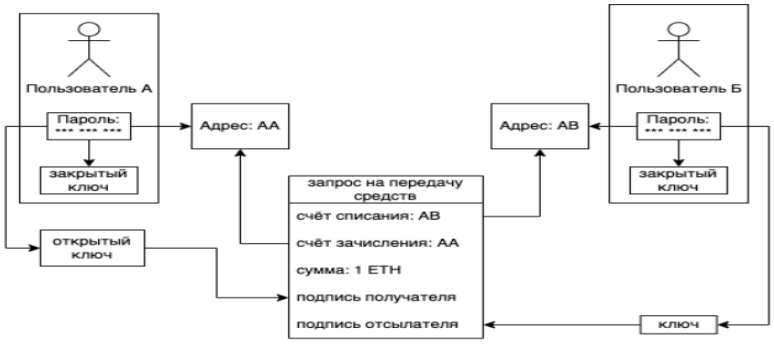

Рис. 1. Схема процесса платёжного оборота в децентрализованной сети
Fig. 1. Scheme of the payment turnover process in a decentralized network

Для проведения операции оплаты пользователь должен иметь на счету достаточное количество токенов и идентификатор получателя – уникальный номер криптокошелька второго участника сети, в сторону которого будет производится списание.

После подтверждения оплаты с помощью одного из вариантов подтверждения (физический ключ – флеш карта с электронной подписью, пароль или фраза из 12 слов) запрос на списание будет поставлен в очередь. Майнер, занимающийся в данный момент обработкой транзакций «возмёт» задачу из очереди и будет тем или иным способом пытаться записать её в блок. Как только сформированный блок с этой и множеством других транзакций будет подписан и принят сетью (более 51% пользователей подтвердят, что блок корректен и начнут использовать данные из него для работы) счёт получателя и отправителя обновятся. Физически валюта не перемещается между пользователями, на самом деле в общий реестр добавляется запись о том, что счёт Пользователя А теперь не имеет N активов, а счёт Пользователя Б был пополнен на сумму N.

Каждый участник сделки (Пользователь А и Пользователь Б) владеет паролем и закрытым ключом, на основе которых были сгенерированы соответствующие адреса кошельков (Адрес АА, Адрес АВ). При оплате Пользователь А должен будет подтвердить факт владения своим кошельком «АА», т.е. подписать транзакцию со своей стороны с помощью закрытого и открытого ключа, и что именно он принял участие в этой транзакции.

Закрытый ключ и открытый ключ представляют собой последовательности символов, генерируемые из пароля. Как и публичный адрес, открытый ключ может свободно распространятся. Закрытый ключ, в свою очередь, должен хранится в секрете и доступ к нему должен иметь только владелец кошелька. В случае использования специализированных кошельков-расширений, например Metamask, закрытый ключ хранится в кэше браузера и подписание происходит в фоновом режиме. Единственное что может запросить кошелёк для подтверждения действий пользователя это установленный им пароль, который так же хранится только на устройстве, где установлен кошелёк.

Для дальнейшего проведения и подтверждения транзакции, система проходит по истории кошелька и убеждается, что кошелёк, используемый для оплаты, владеет достаточным количеством токенов (в истории блокчейн, с этим кошельком связаны транзакции пополнения в сумме на нужный объём), иначе операция отменяется.

После того как система подтвердила баланс пользователя А, создаётся запрос на перевод, помещаемый в очередь для майнера. Майнеры настроены таким образом, что наиболее приоритетными для них являются те транзакции, у которых установлена высокая комиссия за проведение. Для ускорения процесса перевода пользователь А может повысить комиссию (gas в сети ethereum).

В MetaMask предусмотрены следующие типы сетей: основные и тестовые. Основная сеть (Ethereum Mainnet) – это основной блокчейн сети криптовалют. Проекты, достигающие фазы основной сети проходят тщательный процесс оценки перед запуском. Запуск основной сети выполняется только после проверки проекта на безопасность. Любой проект, запущенный в основной сети, доказывает, что его технология криптовалюты жизнеспособна. Mainnet строго придерживается правил, поскольку имеет дело с токенами, которые имеют реальную экономическую ценность [16]. Тестовые сети (Sepolia, Goerly и Linea Goerly) используются разработчиками для создания, модификации и тестирования функциональных возможностей проектов блокчейн, контроля их производительности перед запуском в основной сети. Токены тестовой сети представляют собой «поддельные» криптоактивы (или токены без стоимости), используемые для запуска протокола.

Тестовая сеть предоставляет разработчику набор функций и свойств, список которых может быть выведен в консоли при обращении к свойству «ethereum» (рис. 2).

Проверка доступности инструментов в консоли разработчика осуществляется с помощью свойства «window.ethereum» (рис. 3). В случае, если результатом проверки будет значение «undefiend», значит установка MetaMask не прошла и кошелёк не доступен для разработки и отладки. В случае успешной установки кошелек становится доступным.

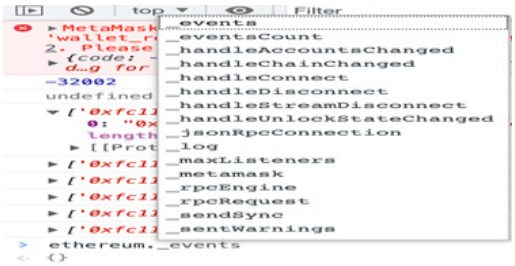

Рис. 2. Список функций разработчика в MetaMask
Fig. 2. List of developer functions in MetaMask

Рис. 3. Вывод свойства window.ethereum в консоль разработчика
Fig. 3. Output of the window.ethereum property to the developer console

Разработка любого Web-приложения предусматривает разработку подсистемы авторизации пользователя в системе. Рассмотрим особенности создания тестовой Web-страницы клиентской части приложения с использованием инструментов сервиса MetaMask. Процесс авторизации пользователя на сайте и вывод данных аккаунта пользователя в консоль может быть описан с помощью функции «connectEthWallet», написанной на языке программирования JavaScript (рис. 4). Программный код сохраняется в файле main.js.

Рис. 4. Пример программного кода авторизации пользователя и вывода данных его аккаунта
Fig. 4. Example of the user authorization code and the output of his account data

Запрос на получение уникального идентификатора кошелька пользователя реализуется с помощью функции «request» (создание запроса). Результат обработки запроса будет выведен на консоль (рис. 5). Для тестирования программного интерфейса Metamask необходимо создать переменную для функции «connectEthWallet», в которой будет содержаться асинхронный вызов анонимной функции. Функция «request» использует реализацию JSON-RPC интерфейса, которая хранится в переменной «ethereum» - «request».

Рис. 5. Вывод на консоль уникального идентификатора кошелька
Fig. 5. Output of the unique wallet identifier to the console

В качестве аргумента в функцию необходимо передать документ в формате JSON, описывающий запрос. В рассматриваемом примере для получения идентификатора кошелька активного пользователя необходимо вызвать метод «eth_requestAccounts». Полный список методов можно найти в официальной документации API ethereum [17]. Для тестирования разработанного программного кода авторизации пользователя авторами был так же разработан HTML код Web-страницы авторизации пользователя (файл index.html) (рис. 6).

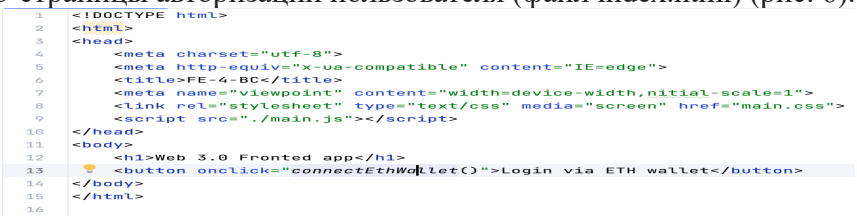

Рис. 6. HTML код файла index.html Web-страницы авторизации пользователя
Fig. 6. HTML code of the file index.html User authorization web pages

Подключение файла main.js выполняется в секции «script». Функция «connectEthWallet» вызывается тэгом «button». Внешний вид Web-страницы авторизации пользователя приведен на рисунке 7. После открытия страницы в браузере отображается заголовок, указанный в тэге h1 «Web 3.0 Fronted app и кнопка «Login via ETH wallet» для имитации входа в систему с использованием ethereum кошелька. После нажатия на эту кнопку вызывается функция «connectEthWallet», описанная в файле main.js и расширение откроет пользователю контекстное меню с выбором аккаунта через который будет осуществляться авторизация (рис. 8).

Рис. 7. Пример Web-страницы авторизации пользователя
Fig. 7. Example of a user authorization Web page

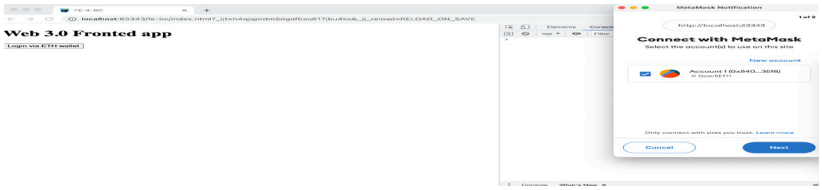

Рис. 8. Диалоговое окно выбора аккаунта пользователя для авторизации
Fig. 8. Dialog box for selecting a user account for authorization

После ввода пароля и выбора пользователем аккаунта «Account 1» становится доступным считывание данных идентификатора кошелька. Получив доступ к кошельку, программа выводит содержимое ответа – публичный адрес кошелька представленный частично после имени пользователя, который может быть использован для получения данных о транзакциях пользователя из сети. Адрес сайта «http://localhost:63343», который запрашивает доступ к кошельку пишется в верхней части диалогового окна расширения.

Получение баланса происходит через вызов метода «eth_getBalance» реализации JSON-RPC интерфейса. В качестве дополнительных параметров (поле «params») необходимо передать адрес кошелька и номер блока, до которого необходимо сделать анализ истории операций и рассчитывать баланс, или указать один из тэгов «latest» (последний), «earliest» (самый первый), «pending» (находящийся в обработке, но ещё не зафиксированный в сети).

Обсуждение результатов. С помощью приведенного в статье примера программного кода авторизации пользователя и описанного алгоритма его тестирования, становится возможным самостоятельно создавать личные анонимные аккаунт-кошельки Ethereum в сервисах, регистрация и дальнейшая работа в которых ведётся с использованием блокчейн аккаунта. Широкий перечень приложений, работающих на блокчейн Ethtereum приведён на официальном сайте сети – ethereum.org в разделе dapps.

Расширение MetaMask является стандартом в сфере криптовалют и блокчейн сетей как со стороны пользователя, так и разработчиков. MetaMask является удобным и надёжным инструментом для хранения криптоактивов. Функционал приложения позволяет не только совершать покупки и переводы, но проходить авторизацию на различных ресурсах, работающих на блокчейн платформах от индустрии цифровой экономики до развлекательных обучающих платформ, сайтов-кинотеатров и игр.

Как правило, разработчики этих сайтов используют общий интерфейс блокчейн сети, который предоставляется MetaMask при установке. Использование программного интерфейса значительно упрощает и ускоряет процесс разработки, он прост и является оболочкой над программным интерфейсом сети Ethereum. Блокчейн как технология может быть внедрена во множество сфер. Рассмотренные авторами базовые инструменты могут быть полезны пользователям в понимании практического использования этой технологии

для реализации собственных проектов.

Вывод. В процессе изучения технологии блокчейн авторами были исследованы информационные ресурсы как русскоязычного, так и англоязычного сегментов сети Интернет. Большая часть открытых источников освещает теоретические основы данной технологии с точки зрения устройства блокчейн сети, при этом не затрагивая вопросы практической реализации и предоставляемых разработчикам и пользователям возможностей и инструментов для работы в рамках конкретной блокчейн сети.

Рассмотренные авторами инструменты MetaMask и примеры их использования могут быть применены при реализации блокчейн ориентированных Web-приложений, а также помочь в понимании базовых принципов построения интеграции пользователя блокчейн сети в закрытую систему.

Библиографический список:

1. Rauchs M., Hileman G. Global Cryptocurrency Benchmarking Study / Rauchs M., Hileman G. [Электронный ресурс] // Cambridge Centre for Alternative Finance Reports : [сайт]. — URL: https://ideas.repec.org/b/jbs/altfin/201704-gcbs.html (дата обращения: 10.03.2023).
2. Soylu P. K., Okur M., Çatıkkaş Ö., Altintig Z. A. Long Memory in the Volatility of Selected Cryptocur-rencies: Bitcoin, Ethereum and Ripple / Soylu P. K., Okur M., Çatıkkaş Ö., Altintig Z. A. [Электрон-ный ресурс] // Journal of Risk and Financial Management : [сайт]. — URL: https://www.mdpi.com/1911-8074/13/6/107 (дата обращения: 20.03.2023).
3. Qiu T., Zhang R., Gao Y. Ripple vs. SWIFT: Transforming Cross Border Remittance Using Blockchain Technology [Текст] // Procedia Computer Science. — 2019. — № 147. — С. 428-434.
4. Shafaq N. K., Faiza L., Chirine G., Elhadj B., Anoud B. Blockchain smart contracts: Applications, challenges, and future trends [Текст] / Shafaq N. K., · Faiza L., · Chirine G., · Elhadj B., · Anoud B. // Peer-to-Peer Networking and Applications. — 2021. — № 14. — С. 2901–2925.
5. Малышенко, Т. И. Развитие технологии blockchain в испании и россии [Текст] / Т. И. Малышенко // Экономика и управление: проблемы, решения. — 2017. — № 9. — С. 117-122.
6. Утечки информации ограниченного доступа в России за 2022 год / [Электронный ресурс] // https://www.infowatch.ru : [сайт]. — URL: (дата обращения: 11.04.2023).
7. Крылов Г. О., Лисицын А. Ю. , Поляков Л. И. Сравнительный анализ волатильности криптовалют и фиатных денег [Текст] / Г.О. Крылов, А.Ю. Лисицын, Л.И. Поляков // Финансы: теория и прак-тика. — 2018. — № 2. — С. 66-89.
8. Chang S., Park Y. Wuthier Uncle-Block Attack: Blockchain Mining Threat Beyond. Block Withhold-ing for Rational and Uncooperative Miners [Текст] / S. Chang, Y. Park, S. Wuthier // Applied Cryp-tography and Network Security. — 2019. — № 11464. — С. 241–258.
9. Qin R., Yuan Y., Wang F. Research on the Selection Strategies of Blockchain Mining Pools [Текст] // Transactions on Computational Social Systems . — 2018. — № 5. — С. 748 - 757.
10. Buterin V., Hernandez D., Kamphefner T., Pham K., Qiao Z., Ryan D., Sin J., Wang Y., Zhang Y. X . Combining GHOST and Casper / V. Buterin, D. Hernandez, T. Kamphefner, K. Pham, Z. Qiao, D. Ryan, J. Sin, Y. Wang, Y. X. Zhang [Электронный ресурс] // https://arxiv.org : [сайт]. — URL: https://arxiv.org/pdf/2003.03052.pdf (дата обращения: 12.04.2023).
11. Berlin K., Dhenakaran S.S An Overview of Cryptanalysis of RSA Public key System [Текст] / Berlin K., Dhenakaran S.S // International Journal of Engineering and Technology. — 2017. — № 9. — С. 3575-3579.
12. Маринкин Д.Н. Проблемы информационной безопасности и криптографии пользователя современного блокчейна [Текст] / Д.Н. Маринкин // Проблемы правоохранительной деятельности. — 2019. — № 3. — С. 39-41.
13. Pramulia D., Anggorojati B. Implementation and evaluation of blockchain based e-voting system with Ethereum and Metamask / Pramulia D., Anggorojati B. [Текст] // International Conference on Infor-matics, Multimedia, Cyber and Information System. — Jakarta, Indonesia:, 2020. — С. 18-23.
14. MetaMask / [Электронный ресурс] // https://chrome.google.com : [сайт]. — URL: https://chrome.google.com/webstore/detail/metamask/nkbihfbeogaeaoehlefnkodbefgpgknn?hl=ru (дата обращения: 11.05.2023).
15. MetaMask / [Электронный ресурс] // https://addons.mozilla.org : [сайт]. — URL: https://addons.mozilla.org/ru/firefox/addon/ether-metamask/ (дата обращения: 30.04.2023).
16. Састрпцин А. Что такое Testnet и Mainnet? Отличия / Састрпцин А. [Электронный ресурс] // https://bytwork.com : [сайт]. — URL: https://bytwork.com/articles/chto-takoe-testnet-i-mainnet-otlichiya (дата обращения: 11.05.2023).
17. JSON-RPC API /[Электронный ресурс] // https://ethereum.org : [сайт]. URL: https://ethereum.org/en/developers/docs/apis/json-rpc/ (дата обращения: 01.05.2023).

References

1. Hileman G. [Electronic resource] // Cambridge Centre for Alternative Finance Reports : [website]. — URL: https://ideas.repec.org/b/jbs/altfin/201704-gcbs.html (accessed: 03/10/2023).
2. Soylu P. K., Okur M., Çatıkkaş Ö., Altintig Z. A. Long Memory in the Volatility of Selected Cryptocurrencies: Bitcoin, Ethereum and Ripple / Soylu P. K., Okur M., Çatıkkaş Ö., Altintig Z. A. [Electronic resource] // Journal of Risk and Financial Management : [website]. — URL: https://www.mdpi.com/1911-8074/13/6/107 (accessed: 03/20/2023).
3. Qiu T., Zhang R., Gao Y. Ripple vs. SWIFT: Transforming Cross Border Remittance Using Blockchain Technology [Text] / Qiu T., Zhang R., Gao Y. // Proceedings Computer Science. 2019;147:428-434.
4. Shafaq N. K., · Faiza L., · Chirine G., · Elhadj B., · Anoud B. Blockchain smart contracts: Applications, challenges, and future trends [Text] / Shafaq N. K., · Faiza L., · Chirine G., · Elhadj B., · Anoud B. // Peer-to-Peer Networking and Applications. 2021; 14: 2901-2925.
5. Malyshenko, T. I. Development of blockchain technology in Spain and Russia [Text]. *Economics and management: problems, solutions.* 2017; 9:117-122. (In Russ)
6. Leaks of restricted access information in Russia for 2022 / [Electronic resource] // https://www.infowatch.ru : [website]. — URL: (accessed: 11.04.2023). (In Russ)
7. Krylov G. O., Lisitsyn A. Yu. , Polyakov L. I. Comparative analysis of the volatility of cryptocurrencies and fiat money [Text] *Finance: theory and practice*. 2018;2:66-89. (In Russ)
8. Chang S., Park Y. Wuthier Uncle-Block Attack: Blockchain Mining Threat Beyond. Block Withholding for Rational and Uncooperative Miners *Applied Cryptography and Network Security*. 2019;11464:. 241-258.
9. Qin R., Yuan Y., Wang F. Research on the Selection Strategies of Blockchain Mining Pools [Text] .*Transactions on Computational Social Systems*. 2018; 5: 748-757.
10. Buterin V., Hernandez D., Kamphefner T., Pham K., Qiao Z., Ryan D., Sin J., Wang Y., Zhang Y. X . Combining GHOST and Casper / V. Buterin, D. Hernandez, T. Kamphefner, K. Pham, Z. Qiao, D. Ryan, J. Sin, Y. Wang, Y. X. Zhang [Electronic resource] // https://arxiv.org : [website]. — URL: https://arxiv.org/pdf/2003.03052.pdf (accessed: 12.04.2023).
11. Berlin K., Dhenakaran S.S An Overview of Cryptanalysis of RSA Public key System [Text]. *International Journal of Engineering and Technology*. 2017; 9: 3575-3579.
12. Marinkin D.N. Problems of information security and cryptography of the modern blockchain user [Text] / *Problems of law enforcement.* 2019; 3: 39-41. (In Russ)
13. Pramulia D., Anggorojati B. Implementation and evaluation of blockchain based e-voting system with Ethereum and Metamask / Pramulia D., Anggorojati B. [Text]. *International Conference on Informatics, Multimedia, Cyber and Information System. Jakarta, Indonesia:,* 2020; 18-23.
14. MetaMask / [Electronic resource] // https://chrome.google.com : [website]. — URL: https://chrome.google.com/webstore/detail/metamask/nkbihfbeogaeaoehlefnkodbefgpgknn?hl=ru (accessed: 05/11/2023).
15. MetaMask / [Electronic resource] // https://addons.mozilla.org : [website]. — URL: https://addons.mozilla.org/ru/firefox/addon/ether-metamask / (accessed: 30.04.2023).
16. Sastrptsin A. What are Testnet and Mainnet? Differences / Sastrptsin A. [Electronic resource] // https://bytwork.com : [website]. — URL: https://bytwork.com/articles/chto-takoe-testnet-i-mainnet-otlichiya (accessed: 05/11/2023). (In Russ)
17. JSON-RPC API / [Electronic resource] // https://ethereum.org : [website]. — URL: https://ethereum.org/en/developers/docs/apis/json-rpc / (accessed: 01.05.2023).

Сведения об авторах:

Окладникова Ольга Дмитриевна, магистрант, факультет программной инженерии и компьютерной техники; hathawway@gmail.com

Буков Александр Викторович, магистрант, факультет программной инженерии и компьютерной техники; 370702@niuitmo.ru

Жуков Николай Николаевич, кандидат физико-математических наук, доцент (квалификационная категория «ординарный доцент»), факультет программной инженерии и компьютерной техники; nnzhukov@itmo.ru

Information about authors:

Olga D. Okladnikova, Master's student, Faculty of Software Engineering and Computer Science; hathawway@gmail.com

Alexander V.Bukov, Master's student, Faculty of Software Engineering and Computer Science; 370702@niuitmo.ru

Nikolai N. Zhukov, Cand. Sci. (Physico-Mathematical), Assoc. Prof. (qualification category «Ordinary assoc. Prof.»), Faculty of Software Engineering and Computer Technology;: nnzhukov@itmo.ru

Конфликт интересов/Conflict of interest.

Авторы заявляют об отсутствии конфликта интересов/The authors declare no conflict of interest.

Поступила в редакцию/ Received 27.06.2023.

Одобрена после рецензирования/ Reviced 20.07.2023.

Принята в печать/ Accepted for publication 20.07.2023.