РЕАЛИЗАЦИЯ ПЛАТЕЖНОЙ СИСТЕМЫ С ИСПОЛЬЗОВАНИЕМ БЛОКЧЕЙНА

И.А. ГОЛОВКО,
Н.Н. РЕБРИКОВ,
Г.Л. КОЗИНА
Национальный университет «Запорожская политехника»
Запорожье, Украина
johnny.work.univer@gmail.com,
n.n.rebrikov@gmail.com,
ainc00@gmail.com

Аннотация. *Реализована легковесная электронная платежная система с использованием централизованной виртуальной валюты и легковесного аналога блокчейна. Система написана на языках программирования TypeScript и Python. Также на языке программирования Python 3.9 разработано консольне программное обеспечение, которое позволяет хешировать сообщения с помощью действующего стандарта Украины ДСТУ 7564:2014. Разработанная платежная система дает возможность пользователям безопасно и анонимно осуществлять транзакции и расчеты.*
Ключевые слова: *кибербезопасность, платежные системы, криптовалюта, блокчейн, веб-сервер, хеширование, ДСТУ 7564: 2014, PYTHON 3.9, TypeScript.*

Для реализации платежной системы использовались языки программирования TypeScript [1] и Python [2], а также фреймворк Nest js и система удаленного вызова процедур gRPC. Данные хранятся в SQL базе данных PostgreSQL. Обмен данными между клиентом и веб-приложением осуществляется по архитектуре REST API. Для написания архитектуры веб-приложения сервера был использован архитектурный паттерн Broker Message.

Для решения проблемы создания цифровой подписи на стороне клиента в браузере создано десктопное приложение подписания транзакции.

В качестве языка программирования был выбран Python 3.9. Он поддерживает многопоточные вычисления, имеет систему обработки

ошибок расчетов, встроенний механизм преобразования систем счисления. В отличие от JavaScript и Go с последней версии собственного интерпретатора Python имеет модульную интеграцию с кодом, написанным на C++, что дает возможность оптимизировать затратные расчеты. Таким образом, Python позволяет создавать высокоэффективные blockchain-системы [3].

Проект состоит из двух частей – клиентской и серверной.

Клиентская часть осуществляет три функции – создание нового кошелька, проверка баланса существующего кошелька и осуществление новой транзакции. Каждая функция запрашивает или посылает данные в серверную часть. Все взаимодействие с серверной частью происходит через HTTP протокол и запросы типа GET и POST.

При создании нового кошелька создается пара ключей – открытий и закрытый, которые «по совместительству» являются соответственно адресом и паролем кошелька.

Для получения баланса кошелька осуществляется запрос типа GET на соответствующий ендпоинт сервера. В запросе указывается адрес кошелька, после чего система выводит баланс бумажника. Поскольку платежная система в данной реализации является абсолютно анонимной, то любой пользователь может получить данные о балансе другого кошелька, даже того, что ему не принадлежит.

При выполнении транзакции используется хеширование данных.

В предлагаемом проекте хеширование осуществляется по алгоритму SHA-1.

Однако авторами разработано консольное программное обеспечение на языке программирования Python 3.9, которое позволяет хешировать сообщение в соответствии с действующим стандартом хеширования Украины ДСТУ 7564:2014 [4].

Данный стандарт является общепринятым в государственном и частном секторах, когда предприятие нуждается в создании собственных надежных систем защиты информации, циркулирующие в информационно-коммуникационных системах.

При проектрировании и разработке алгоритма хеширования была составлена диаграмма класса DSTU7564, представленного в программной разработке.

Корректность выполнения реализованного алгоритма была протестирована с использованием тестовых данных, представленных в Приложении Б ДСТУ 7564:2014.

В результате была получена програмная реализация алгоритма хеширования, которая удовлетворяет требованиям современных информационных систем.

После выполнения хеширования осуществляется подписание данных по протоколу RSA-PSS [5].

На стороне сервера существует четыре функции: функция, которая возвращает баланс кошелька пользователя; функция, которая проверяет существует, ли пользователь в базе данных; функция проверки правильности подписи; функция создания блока и записи его в блокчейн.

После подтверждения правильности и целостности присланных данных они поступают в функции формирования блока. Блок формируется, когда поступает определенное количество транзакций. Он дополняется хэшем из прошлого блока или нулем, если в прошлого блока не существует. После окончательного формирования блока осуществляется его хеширование и запись в базу данных, то есть к формируемому блокчейну.

Предлагаемая легковесная платежная система дает возможность пользователям безопасно и анонимно осуществлять транзакции и расчеты. Тем не менее данный проект может быть улучшен в частности за счет использования сервиса Redis, перевода с консольного к web-приложению и проч.

Литература.

1. Что такое TypeScript? – https://www.typescriptlang.org/
2. Официальный сайт Python – https://www.python.org/
3. Что такое технология блокчейна? – https://www.ibm.com/ru-ru/topics/what-is-blockchain
4. Функция хеширования ДСТУ 7564:2014 – https://usts.kiev.ua/wp-content/uploads/2020/07/dstu-7564-2014.pdf
5. Схема цифровой подписи на основе RSA (PKCS # 1 v1.5) – https://pycryptodome.readthedocs.io/en/latest/src/signature/pkcs1_v1_5.html