УДК 004.8

ВИРІШЕННЯ ПРОБЛЕМИ МАСШТАБУВАННЯ БІТКОІН БЛОКЧЕЙНУ ЗА ДОПОМОГОЮ ЦЕНТРАЛІЗОВАНОГО СЕРВІСУ АГРЕГАЦІЇ ТРАНЗАКЦІЙ

Щербіна Євгеній, Месюра Володимир

Вінницький національний технічний університет

Анотація

Проаналізовано проблему масштабування біткоін блокчейну. Розглянуто підходи до її вирішення. Виділено основні типи рішень. Детально розглянуто використання централізованих систем агрегації транзакцій.

Abstract

Bitcoin blockchain scaling problem is analyzed. Approaches to its solution are considered. The main types of solutions are highlighted. The use of centralized transaction aggregation systems is discussed in detail.

Вступ

На сьогоднішній день у світі дуже розповсюджений спосіб оплати з використанням електронних гаманців та готівки, переведеної у криптовалюту. Існує декілька сотень різних криптовалют, проте найпопулярнішою з них є біткоін - електронна валюта, концепт якої був озвучений і 2008 році її розробником - Сатосі Накамото [1].

Основна частина

Для досягнення децентралізації біткоїну всі її учасники розглядаються як рівнозначні. Таким чином кожен з учасників (вузлів) має зберігати весь реєстр транзакцій та приймати участь у мережевому обміні повідомленнями, що містять у собі інформацію про транзакції та блоки. Саме через це ми стикаємося з проблемою масштабування [3].

Існує декілька основних підходів до вирішення даної проблеми:

1. Оф-чейн протоколи
2. Сайдчейн протоколи
3. Централізовані оптимізації.

Дані підходи не є взаємовиключними, отже можуть бути поєднані при реалізації блокчейн системи. Рішення проблеми за допомогою оф-чейн та сайдчейн протоколів були вичерпно розглянуті та проаналізовані відомими вченими, робота буде сфокусована на вирішенні проблеми за допомогою централізованих систем агрегації транзакцій [4].

Модель аккаунтів та транзакцій у біткойн блокчейні

Біткоін блокчейн має так звану UTXO(unspent transaction output). UTXO є подібним до монети або ж купюри у реальномі світі. У блокчейні існує велика кількість UTXO(монет), при чому у кожної з монет є власник. Людина(аккаунт) може володіти будь-якою кількістю UTXO.

Транзакція у блокчейні полягає у знищені одних та створенні інших UTXO. На рисунку 1 подано приклад транзакції з трьома входами та двома виходами, отже дана транзакція "знищує" три UTXO - 0.5 BTC, 0.5 BTC та 1 BTC та створює два нових - 0.2 BTC та 1.8 BTC.

Дана транзакція відображає покупку товару(ів) вартістю 1.8 BTC аккаунтом, який володіє трьома UTXO номіналами: 0.5 BTC, 0.5 BTC та 1 BTC, а 0.2 BTC в даному випадку відіграє роль "здачі". З поданого прикладу видно, що це дуже схоже на звичайну покупку товарів у магазині за готівку.

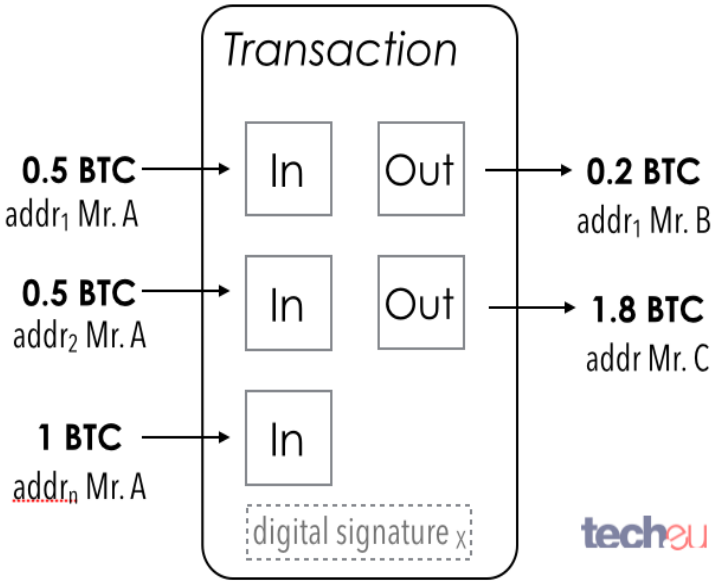

Рисунок 1– Приклад транзакції в біткоін блокчейні

Централізовані системи агрегації транзакцій

Основна ідея сервісу централізованої агрегації транзакцій полягає у тому щоб накопичувати наміри клієнтів виконати транзакції протягом деякого часу та загрегувати їх в одну велику транзакцію [5]. Також у разі виконання декількох транзакцій для одного і того ж отримувача у кінцевій транзакції буде менша кількість "Outputs". Оскільки "Output" включає в себе scriptPubKey (locking_script) - це є суттєвою оптимізацією [6].

На рисунку 2 можемо бачити приклад службових елементів Version, Sequence та Locktime, що займають чотири байти кожен. Також подано приклад елементу scriptPubKey, що в даному випадку являє собою pay-to-public-key-hash output, який є похідною від публічного ключа та займає приблизно 33 байти.

Службовий елемент Version використовується задля того, щоб мати можливість підмінити стару транзакцію новою (з більшим значенням атрибуту Version). Цікавим є той факт, що хоч така семантика і закладена у протоколі, наразі вона не працює, оскільки не має жодних інструментів, що змусили б майнерів підпорядковуватись даному правилу.

Службовий елемент Locktime використовується для того щоб не дозволити блокчейну прийняти транзакцію до певного часу у майбутньому. Також варто зазначити, що час у майбутньому може бути заданий як номером блоку у блокчейні, так і за допомогою UNIX TimeStamp. Якщо число менше 500 млн., то воно інтерпретується як номер блоку, інакше - як UNIX TimeStamp.

Даний сервіс надає наступні переваги:

1. Зменшується середній розмір транзакції, внаслідок чого зменшується навантаження на блокчейн.
2. Кінцеві користувачі сплачують меншу комісію за транзакцію [7].

Для більшої наочності наведемо приклад, де агрегація транзакцій може суттєво зменшити розмір кінцевої транзакції. Уявімо, що є дві людини(аккаунта), кожен з яких володіє UTXO номіналом 1 BTC. Вони обидва хочуть купити товар вартістю 1 BTC у одного і того ж продавця. У звичайній ситуації ми мали б згенерувати 2 транзакції, сумарно у яких були б два входи та два виходи. Проте у разі використання системи агрегації транзакцій ми генеруємо лише одну транзакцію з двома входами і одним виходом [8].

Усього вище переліченого ми можемо досягти за допомогою sighash індикаторів. Існують такі індикатори:

1. SIGHASH_ALL
2. SIGHASH_ONE
3. SIGHASH_NONE

Також існує модифікатор ANYONECANPAY, що разом з sighash індикаторами значно розширює можливості біткоін транзакцій.

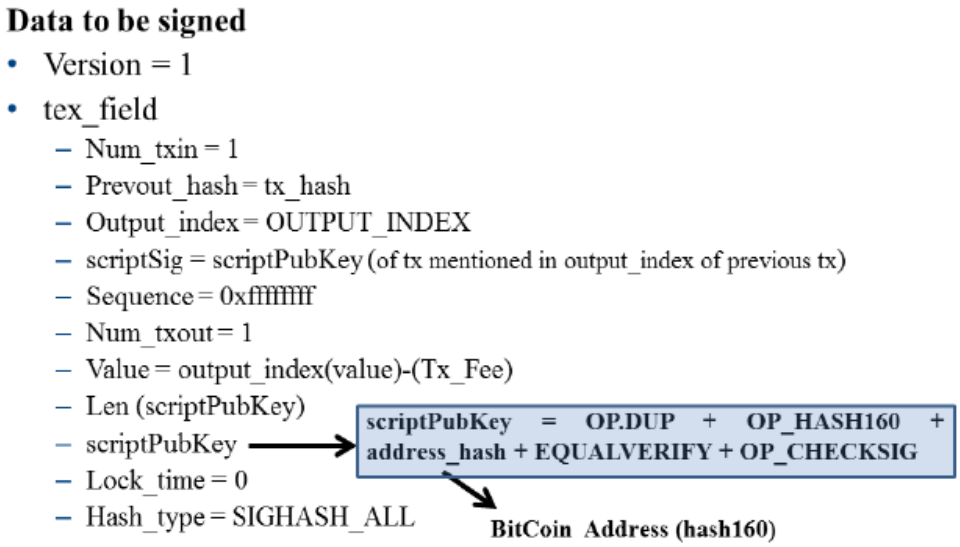

Рисунок 2 – Детальний опис елементів біткоін транзакції

Висновки

На сьогоднішній день можна виділити три основні підходи до вирішення проблеми масштабування блокчейну: оф-чейн протоколи, сайдчейн протоколи, централізовані оптимізації. Було детально розглянуто підхід з використанням централізованої системи агрегації транзакцій та виділено його основні переваги:

1. Зменшується середній розмір транзакції, внаслідок чого зменшується навантаження на блокчейн.
2. Кінцеві користувачі сплачують меншу комісію за транзакцію.
3. В ході роботи буде здійснено програмну реалізацію даного підходу.

Список використаних джерел

1. Сатоші Накамото "Bitcoin: A Peer-to-Peer Electronic Cash System" - https://bitcoin.org/bitcoin.pdf
2. W. Dai, "b-money," http://www.weidai.com/bmoney.txt, 1998.
3. H. Massias, X.S. Avila, and J.-J. Quisquater, "Design of a secure timestamping service with minimal trust requirements," In 20th Symposium on Information Theory in the Benelux, May 1999.
4. D. Bayer, S. Haber, W.S. Stornetta, "Improving the efficiency and reliability of digital time-stamping," In Sequences II: Methods in Communication, Security and Computer Science, pages 329-334, 1993.
5. S. Haber, W.S. Stornetta, "Secure names for bit-strings," In Proceedings of the 4th ACM Conference on Computer and Communications Security, pages 28-35, April 1997.
6. A. Back, "Hashcash - a denial of service counter-measure," http://www.hashcash.org/papers/hashcash.pdf, 2002.
7. R.C. Merkle, "Protocols for public key cryptosystems," In Proc. 1980 Symposium on Security and Privacy, IEEE Computer Society, pages 122-133, April 1980.
8. W. Feller, "An introduction to probability theory and its applications," 1957.