ПРОТОКОЛ ZELDHASH

Полюйте на найрідкісніші транзакції Bitcoin і заробляйте ZELD.

Автор: Ouziel Slama

1 Мотивація

- Заради азарту полювання. Кожна транзакція стає можливістю відкрити щось рідкісне — цифровий скарб, прихований на виду в блокчейні.
- Ці патерни ведучих нулів не просто рідкісні — вони також можуть покращити стиснення, потенційно оптимізуючи зберігання та ефективність обробки блокчейну.
- Будь-хто може заробляти ZELD, полюючи на рідкісні транзакції — не як у майнінгу блоків Bitcoin, де один переможець на блок. Полювання відкрите для всіх.
- У разі успіху токени ZELD зможуть врешті-решт відшкодовувати комісії за транзакції — винагороджуючи мисливців, які знаходять найрідкісніші знахідки!

2 Майнінг ZELD

Щоб майнити ZELD, ви повинні транслювати транзакцію Bitcoin, txid якої починається щонайменше з 6 нулів. Винагорода розраховується на основі того, як ваша транзакція порівнюється з найкращою транзакцією в блоці:

- У даному блоці транзакції, що починаються з найбільшої кількості нулів, заробляють 4096 ZELD
- Транзакції з одним нулем менше, ніж найкращі транзакції, заробляють 4096/16 або 256 ZELD
- Транзакції з двома нулями менше заробляють 4096 / 16 / 16 або 16 ZELD
- і т.д.

Тому використовується наступна формула:

```
reward = 4096 / 16 ^ (max_zero_count - zero_count)
```

Де `max_zero_count` дорівнює кількості нулів, якими починається найкраща транзакція, а `zero_count` — кількість нулів, якими починається транзакція, для якої ми розраховуємо винагороду.

Примітка: Coinbase-транзакції не мають права на винагороди ZELD.

3 Розподіл ZELD

ZELD, зароблені транзакцією, що починається з 6 або більше нулів, розподіляються по UTXO. Розподіл здійснюється наступним чином:

- Якщо є один не-OP_RETURN UTXO, він отримує всю винагороду.
- Якщо є два або більше не-OP_RETURN UTXO, винагорода розподіляється між усіма UTXO, крім останнього, пропорційно вартості кожного UTXO

- Оскільки обчислення виконуються лише з цілими числами, можливий залишок від ділення розподіляється на перший не-OP_RETURN UTXO.

Наприклад, якщо транзакція, що заробляє 256 ZELD, містить 4 виходи з 500, 500, 500 та 2000 сатоші відповідно, перший вихід отримує 86 ZELD винагороди, другий і третій — по 85 ZELD.

4 Переміщення ZELD

Коли UTXO з прикріпленими ZELD витрачаються, ZELD розподіляються на нові UTXO в транзакції. Існує два методи розподілу ZELD при їх переміщенні:

4.1 Метод 1: Автоматичний пропорційний розподіл

За замовчуванням розподіл виконується точно так само, як і винагороди — пропорційно на основі Bitcoin-вартості вихідних UTXO, виключаючи останній вихід, якщо їх декілька.

4.2 Метод 2: Користувацький розподіл через OP_RETURN

Ви можете точно вказати, як мають розподілятися ZELD, включивши у вашу транзакцію вихід OP_RETURN з даними користувацького розподілу. Це дозволяє точно контролювати перекази ZELD.

4.2.1 Формат OP_RETURN:

- Скрипт OP_RETURN повинен містити дані, що починаються з 4-байтового префікса “ZELD”
- Після префікса дані повинні бути закодовані у форматі CBOR
- Дані CBOR повинні представляти вектор беззнакових 64-бітних цілих чисел (Vec)
- Кожне ціле число вказує, скільки ZELD надіслати на відповідний вихідний UTXO

4.2.2 Правила розподілу:

- Кількість значень у масиві розподілу автоматично коригується для відповідності кількості не-OP_RETURN виходів
- Якщо масив занадто довгий, зайві значення видаляються
- Якщо масив занадто короткий, додаються нулі
- Загальна сума значень розподілу не може перевищувати загальну кількість ZELD, що витрачаються
- Якщо сума менша за загальну кількість, різниця додається до першого виходу
- Якщо сума перевищує загальну кількість, транзакція повертається до пропорційного розподілу
- Винагороди ZELD від нового майнінгу завжди розподіляються пропорційно, а потім об’єднуються з користувацьким розподілом

4.2.3 Приклад:

Якщо у вас є 1000 ZELD для розподілу по 3 виходах, і ви хочете надіслати 600 на перший, 300 на другий і 100 на третій, ваш OP_RETURN міститиме “ZELD”, за яким слідує CBOR-кодування [600, 300, 100].

Примітки:

- Якщо допустимий розподіл OP_RETURN не знайдено, транзакція автоматично використовує метод пропорційного розподілу.
- Якщо транзакція містить лише один вихід OP_RETURN, будь-які ZELD, прикріплені до входів транзакції, **та будь-які щойно зароблені винагороди** назавжди спалюються, тому що немає витратних виходів для їх отримання.
- При наявності кількох виходів OP_RETURN для розподілу враховується лише той, що з'являється останнім у транзакції та несе дійсне корисне навантаження `ZELD`+CBOR.