Stacks Tenure Transaction*

Phillip G. Bradford†

March 11, 2025

Abstract

Miner tenure is updated by a tenure transaction in the Stacks chain. A tenure tranasaction is a special type of transaction. Miners are elected to produce blocks in the Stacks chain by the stackers.

1 Tenure tranasaction

Currently, there seems to be no *v2* API for tenure transactions. So we use the v1 API:

https://*api.mainnet.hiro.so/extended/v1/tx?type=tenure_change&limit=1*

For example, try:

Listing 1: Curl command for getting the current tenured transaction

```
1  Windows> curl.exe  "https://api.mainnet.hiro.so/extended/
     v1/tx?type=tenure_change&limit=1"
```

This endpoint gets the last tenure transaction. The response is a JSON transaction object in Listing 2. The tenure transaction is a special type of transaction. Its tx_type has the value tenure_change. The tenure transaction is used to change the consensus hash for the next tenure on the Stacks chain. The consensus hash is a hash of the block header. The consensus hash is used to determine the next block producer.

*These notes use github co-pilot AI.

†phillip.bradford@uconn.edu, phillip.g.bradford@gmail.com, University of Connecticut, Department of Computing, Stamford, CT USA

Listing 2: Tenure transaction


```
{
  "tx_id": "0xeda08c23df80be0cf6fde074e
     6c7651f117f306a426f92fd660dbd0d18643942",
  "nonce": 18469,
  "fee_rate": "0",
  "sender_address": "
   SP3MCVE3HJP6T8QS4A9R0ETVJVGRZJA04MKJTPHG5",
  "sponsored": false,
  "post_condition_mode": "deny",
  "post_conditions": [],
  "anchor_mode": "on_chain_only",
  "block_hash": "0xd3c4d7b45303f0450bf2e015e
      b7bafc9b8dd05096089664b87613e95209981f9",
  "block_height": 749062,
  "block_time": 1741635624,
  "block_time_iso": "2025-03-10T19:40:24.000Z",
  "burn_block_time": 1741635606,
  "burn_block_height": 887215,
  "burn_block_time_iso": "2025-03-10T19:40:06.000Z",
  "parent_burn_block_time": 1741635353,
  "parent_burn_block_time_iso": "2025-03-10T19:35:53.000Z
   ",
  "canonical": true,
  "tx_index": 0,
  "tx_status": "success",
  "tx_result": {
    "hex": "0x0703",
    "repr": "(ok true)"
  },
  "event_count": 0,
  "parent_block_hash": "0
   xd523a4e7e3d719752caba280e9fa77521
     16cbb7c49abbcd5c9b93a3cdf8e385b",
  "is_unanchored": false,
  "microblock_hash": "0x",
  "microblock_sequence": 2147483647,
  "microblock_canonical": true,
  "execution_cost_read_count": 0,
  "execution_cost_read_length": 0,
  "execution_cost_runtime": 0,
  "execution_cost_write_count": 0,
```

```
    "execution_cost_write_length": 0,
    "events": [],
    "tx_type": "tenure_change",
    "tenure_change_payload": {
      "tenure_consensus_hash": "0
       x626ae97693d16ea81e2bb97a454f78fbc97008c0",
      "prev_tenure_consensus_hash": "0
       x68209d52defca5f053ffda848aa3f86f972095a7",
      "burn_view_consensus_hash": "0
       x626ae97693d16ea81e2bb97a454f78fbc97008c0",
      "previous_tenure_end": "0
       x5ddc1865e9cdfcc02d8a7d4d028586fb
        7a5e9af3a431f063fd31ce8cdb4b9f30",
      "previous_tenure_blocks": 17,
      "cause": "block_found",
      "pubkey_hash": "0
       xe8cdb871958da45f245270076b72dc31f92804a4"
    }
  }
```


Listing 3 shows the key fields of a tenure transaction. The tenure transaction has a tx_type of tenure_change. The tenure_change_payload has the following fields:

1. tenure_consensus_hash is the consensus hash for the next tenure.
2. prev_tenure_consensus_hash is the consensus hash for the previous tenure.
3. burn_view_consensus_hash is the consensus hash for the burn view.
4. previous_tenure_end is the end of the previous tenure.
5. previous_tenure_blocks is the number of blocks in the previous tenure.
6. cause is the cause of the tenure change. It can be block_found or timeout. The cause is block_found if a block was found in the previous tenure. The cause is timeout if no block was found in the previous tenure.
7. pubkey_hash is the public key hash of the block producer.

Cosider a tenure transaction and its prev_tenure_consensus_hash field. This field can be used to find the previous tenure transaction. Hence the previous miner.

Listing 3: Tenure part

```
"tenure_change_payload": {
  "tenure_consensus_hash": "0x626ae97693d16ea81e2bb97a
    454f78fbc97008c0",
  "prev_tenure_consensus_hash": "0x68209d52defca5f053f
    fda848aa3f86f972095a7",
  "burn_view_consensus_hash": "0x626ae97693d16ea81e2bb
    97a454f78fbc97008c0",
  "previous_tenure_end": "0
   x5ddc1865e9cdfcc02d8a7d4d028586fb
    7a5e9af3a431f063fd31ce8cdb4b9f30",
  "previous_tenure_blocks": 17,
  "cause": "block_found",
  "pubkey_hash": "0xe8cdb871958da45f245270076
    b72dc31f92804a4"
}
```

References

[1] How tranasactions work: https://docs.stacks.co/concepts/transactions/transactions

[2] api.hiro.so endpoint documentation: https://docs.hiro.so/stacks/api/transactions

[3] Kenny Rogers, Nakamoto in 10 minutes: https://docs.stacks.co/nakamoto-upgrade/nakamoto-in-10-minutes 2025-03-10.

[4] https://tc39.es/ecma262/