Stellar Transactions

Saravanan Vijayakumaran
sarva@ee.iitb.ac.in

Department of Electrical Engineering
Indian Institute of Technology Bombay

September 18, 2018

Stellar Transactions

- Commands to modify ledger state
- Contain upto 100 operations
- List of possible operations
  - Create Account
  - Payment
  - Path Payment
  - Manage Offer
  - Create Passive Offer
  - Set Options
  - Change Trust
  - Allow Trust
  - Account Merge
  - Inflation
  - Manage Data
  - Bump Sequence
- Transaction fees = Number of operations $\times$ Base fee
  - Current base fee = 100 stroops = 10^{-5} XLM
- Fees added to fee pool and distributed via inflation voting

Transaction Fields and Sets

- Transaction fields
  - **Source account** = Account ID transaction source
  - **Fee** = Transaction fees
  - **Sequence number**: Must be 1 greater than source account sequence number
  - **List of operations**
  - **List of signatures**: Upto 20 signatures can be included
  - **Memo** = Optional data field (upto 32 bytes long)
  - **Time bounds** = Optional lower and upper UNIX times specifying transaction validity
- Transaction sets
  - Collections of transactions proposed for inclusion in next ledger closing
  - Stellar consensus protocol (SCP) is used to achieve consensus
  - The transaction set picked by SCP is applied to current ledger state

Inflation

- New lumens added to the network at the rate of 1% per year
- Each week these lumens are distributed via the Inflation operation
- Distribution algorithm
  1. Calculate inflation pool as

$$\text{Total lumens in existence} \times \text{Weekly inflation rate} + \text{Fee pool}$$

  2. Calculate vote threshold as

$$\text{Total lumens in existence} \times 0.0005$$

  3. Determine the accounts which receive more votes than the threshold
  4. Allocate lumens to winners proportional to the votes they received
  5. Return unallocated lumens to the fee pool

Trustline Operations

- **Change Trust**: Used by regular accounts to create, update, or delete trustline with anchor
  - Inputs
    - **Line** = Asset in trustline
    - **Limit** = The limit of the trustline
  - Possible errors
    - CHANGE_TRUST_NO_ISSUER = Issuer of asset cannot be found
    - CHANGE_TRUST_LOW_RESERVE = Account does not have enough XLM to allow addition of new trustline subentry
- **Allow Trust**: Used by anchors to authorizes user-created trustlines
  - Inputs
    - **Trustor** = Account ID of recipient of trustline
    - **Type** = Asset in trustline
    - **Authorize** = Flag indicating trustline authorization
  - Possible errors
    - ALLOW_TRUST_NO_TRUST_LINE = Trustor does not have trustline with anchor performing this operation

Multisignature in Stellar

- Each account specifies upto 20 signers (in addition to owner)
  - Each signer is a public key and a weight
  - Account owner also has a weight called master key weight
  - Example: Master key weight = 1, Alice's key weight = 1, Bob's key weight = 1
- Thresholds for account operations
  - Operations have three possible categories: low, medium, high
    - **Low security**: Inflation, Allow Trust, Bump Sequence
    - **High security**: Updating signers and thresholds, Account Merge
    - **Medium security**: Payment, Create Account, Everything else
  - Thresholds for each category are an integer from 0 to 255
  - Example: low thres = 1, medium thres = 1, high thres = 3
- For each operation, sum of weights of signatories should exceed threshold
- Anchor setup example
  - Master key weight = 2, Additional key weight = 1
  - low thres = 0, medium thres = 2, high thres = 2
  - Master key is kept offline and additional key is kept online

Ledger

- State of the Stellar system at a given time
- Ledger header fields
  - **Version** = Protocol version
  - **Previous Ledger Hash**
  - **SCP Value** = Result of Stellar consensus protocol
    - **Transaction set hash**: Hash of transaction set applied to previous ledger
    - **Close time**: Time at which network closed this ledger
    - **Upgrades**: Base fee changes and protocol upgrades (optional)
  - **Transaction set result hash** = Hash of results of applying transaction set
  - **Bucket list hash** = Hash of all ledger objects
  - **Ledger sequence** = Sequence number of this ledger
  - **Total coins** = Total number lumens in existence
  - **Fee pool** = Number of lumens paid in fees since last inflation operation
  - **ID pool** = Last used global ID. Used to generate unique offer IDs.
  - **Maximum number of transactions**: Currently 50
  - **Base fee**
  - **Base reserve**
  - **Skip list** = Hashes of 4 past ledgers

References

- Transactions https://www.stellar.org/developers/guides/concepts/transactions.html
- List of operations https://www.stellar.org/developers/guides/concepts/list-of-operations.html
- Inflation https://www.stellar.org/developers/guides/concepts/inflation.html
- Multisignature https://www.stellar.org/developers/guides/concepts/multi-sig.html
- Ledger https://www.stellar.org/developers/guides/concepts/ledger.html
- XDR definitions of Stellar data structures https://github.com/stellar/stellar-core/tree/master/src/xdr
- External Data Representation (XDR) http://tools.ietf.org/html/rfc4506.html