Scam token exclusion policy for RChain genesis block

This document describes how the scam tokens and tainted addresses will be handled for the genesis ceremony. There is a simple recursive algorithm that tracks the taint and handles tokens mixed in with legitimate addresses without much burden on the whole community.

Suppose that Alice has t tokens at an address on an exchange, call it A. She is conned into accepting s tainted tokens, say for BTC or fiat, or something else of value. The total tokens at A are s+t. Now, let us say that after the address A is tainted with the scammer's tokens, Alice sends x tokens to Bob and y tokens to Carrol, and this is the total list of transactions between when Alice received the tainted tokens and block 9371743. We know that x + y + balance(A) = s+t. Therefore,

$$1 = x/s+t + y/s+t + balance(A)/s+t$$

We use these fractions to calculate the taint in each of Bob's address (B), Carrol's address (C), and Alice's remaining balance. Specifically,

x/(s+t) * s is the number of tainted tokens that have landed at B
y/(s+t) * s is the number of tainted tokens that have landed at C
(x + y)/s+t * s is number of tainted tokens remaining in A.

Example

Say, s = 10, t = 90, s+t = 100, x = 25, y = 25

```
25/100 * 10 = 2.5 tainted tokens Bob received
25/100 * 10 = 2.5 tainted tokens Carrol received
50/100 * 10 = 5   tainted tokens remaining in A
------------------------------------------------------------
total         10  tainted tokens originally sent to Alice
```

Notice that once tainted tokens have landed at B the same procedure applies. If Bob sends tokens from B, then a percentage of each transaction originating from B after the address has been tainted will be considered tainted tokens and deducted accordingly from the addresses receiving tokens from B at the genesis ceremony.

Covering all tainted addresses

Now, we begin at the scammer's original address and take the transitive closure of all the addresses involved in transactions that originate from the scammer address and we can calculate the tainted tokens at every address, including addresses where tainted tokens are mixed with legitimate tokens.

Let A be an address tainted by transaction T, with balance t(A) prior to receiving s(A) tainted tokens in transaction T. Let T_1, …, T_n be transactions such that src(T_i) = A and T_i all occur after T and before blockheight. Without loss of generality, we assume there are no transactions, U, occurring after T and before blockheight such that trgt(U) = A. Let amt(T_i) represent the amount of tokens being transferred out of A.

taintedTokensTransmitted(T_i) = (amt(T_i)/s(src(T))+t(src(T))) * s(src(T))

Write T > S if T occurs after S. Write T < blockHeight if T occurs before blockHeight.

transitiveClosure(A,S,blockheight) =
  { T | src(T) = A or src(T) ∈ map(src, transitiveClosure(A,S)), blockheight > T > S }

adjustedBalance(Addr,A,S,blockheight) =
  balance(Addr) - $\sum_{\{T \in \text{transitiveClosure(A,S,blockheight)} \mid \text{trgt(T)} = \text{Addr}\}}$ taintedTokensTransmitted(T)

Genesis block adjustment

The genesis block balances will be adjusted from the snapshot taken at 9371743 according to these rules. The balances of the tainted RHOC will be returned to the Cooperative.