

跨链转移流程

genesis里面tokenChain中的信息是可以从那条链上转到本链(设计成数组,但是只可以写一个)
如果B链想从A链跨链转,B链就写A链的信息

交易一流程

- 1.A链,账户1先调用内置合约XChainTransferWithdraw,to写合约地址,payload里面写chainid(目标链),Chainown(目标链),DstUserAddr(目标地址)
- 2.交易调用到内置合约,走withdraw方法,,纪录要转的明细,生成key(是用 conf.PDXKeyFlag, "withdraw", from.String(), payload.DstUserAddr.String(), payload.DstChainID,)

减去A链from的钱

- 3.取from生成key(WithdrawTxKeyHash),取出跨链列表([]WithdrawTx),把本次的交易调用数据存入数据库,交易一完成
- 3.1交易完成之后调用API来确认交易一是否完成(GetWithdrawTransactionByHash)
 - (1)先从数据库中找到对应交易(交易1)
 - (2)把交易1中data(payload)解码
 - (3)取出当前commit,再用交易1对应的normalBlock取出commitBlock,然后用
 - (4)取出最新的state
 - (5)用HomesteadSigner生成from,或者用EIP155计算出from(如果是受保护的就EIP155)
 - (6)用合约地址和from找出交易列表(A链),对应里面要查的hash,如果列表里有这个hash表示成功,如果没超过2个commit(XChainTransferCommitNum可以设置),返回1如果超过返回2
 - (7)如果没找到从数据库中找到,就从txpool中找,返回状态0,可以得到txmsg(交易1),用txmsg可以得转成transaction ,发第二笔交易的时候要带上第一笔交易的交易报文(为了解析交易1的from)

交易二流程

- 1.调用内置合约XChainTransferDeposit,payload中存的txmsg是从交易一查到的
- 2.用txmsg转成transaction格式,然后再用HomesteadSigner生成from
- 3.用transaction中的data解析出withdrawPayload,里面是(
DstChainID string json:"dst_chain_id" (A链)
DstChainOwner common.Address json:"dst_chain_owner" (A链)
DstUserAddr common.Address json:"dst_user_addr" (发起跨链的from)
DstContractAddr string json:"dst_contract_addr" (合约地址)
)
- 4.如果发起交易二的from不等于解析出的from(发送交易一的),直接加入黑名单(交易二是免费的),

还要确认是否已经存过,查询数据库,具体存储在第八步(为了判断是否是自己操作交易1 和 交易2 都是自己操作)

5.去srcChain(A链)查询withdraw的状态,通过genesis中tonkenchain中的信息对比chainID,然后获取IP和端口,然后调用getWithdrawTransactionByHash(再次查询交易一的状态),如果状态是2,在查看(交易一)是否超过两个commit(XChainTransferCommitNum可以设置),最终返回要给目标账户转账金额

6.对比转账金额是否正确(交易一的转账金额,和交易二的转账金额是否相等(交易二免费 value可以填,但是必须跟交易一相同))

7.给目标账户+钱

8.然后在B链的合约地址上存入交易一的hash对应的金额,证明已经给目标账户加过钱了

交易三(API)查询交易二是否完成

1.调用API查询交易一的状态(GetUncompletedTransactionByAddress),根据from和内置合约地址(XChainTransferWithdraw),来找出第三步的跨链列表([]WithdrawTx)

2.根据列表中的chainID取对应的链上(B链)查询给目标账户的转账是否完成(CheckTxCompletedByHash)

(1)获取最新的stateDB

(2)根据from和txHash来查询交易二是否成功,如果取出值证明这笔交易已经存过了,证明交易二完成

