**一，操作说明步骤**

**（1）/newTx （**[**http://localhost/#/newTx**](http://localhost/#/newTx)**）**

1. Alice打开/newTx路由，并选择chainA链
2. Alice选择地址addressA1 作为此次交易的发起账户
3. Alice选择Bob的地址addressB1作为此次交易的提币方（只有addressB1有资格提走合约中锁定的资产）
4. Alice 点击 genHashAndSecret 按钮 生成此次交易的hash锁和hash密码
5. Alice记录保存上步骤生成的hashlockAB和hashsecretAB
6. Alice选择一个有足够的操作时间的时间T1>now 作为时间锁
7. Alice输入此次交易锁定的chainA链上的token数额valueA
8. Alice点击按钮newTransaction并得到成功返回的回执receipt
9. Alice记录回执中的txId txIdA
10. Alice提交给Bob第5步生成的hashlockAB和第6步选择的T1和第9步生成的txIdA
11. Bob打开/newTx路由并选择chainB链
12. Bob选择地址addressB2作为此次交易的发起账户
13. Bob选择Alice的账户addressA2作为此次交易的提币方（只有addressA2有资格提走合约中锁定的资产）
14. Bob输入从Alice发过来的hashlockAB作为hashlock参数
15. Bob选择一个延后的时间T2作为参数timelock(now<T2<T1)，并保证T2-T1的时间足够Alice提走他的资产
16. Bob输入数额valueB，（按照市场兑换行情，chianA链上的valueA和chainB链上的valueB价值相等）
17. Bob点击按钮newTransaction并获得交易执行成功后的回执receipt
18. Bob记录上步骤receipt中的txId txIdB
19. Bob把时间T2和txIdB发给Alice

#### （2）/withdraw (http://localhost/#/withdraw)

1. Alice 打开 '/withdraw' 路由并选择chainB链
2. Alice选择账户addressA2作为此次交易的账户
3. Alice输入Bob发过来的txIdB
4. Alice输入hashsecret hashsecretAB
5. Alice 点击按钮withdraw 并得到Bob锁在合约中的valueB数额的chainB链资产（前提是在T2时间前操作）

#### （3）/getTx (http://localhost/#/getTx)

1. Alice 在chainB链上提走了Bob锁在合约里的资产，假设Bob监听了资产提取的事件，那Bob马上就知道了这件事
2. Bob打开路由/getTx选择chainA
3. Bob选择账户addressB1作为交易账户
4. Bob输入txId: txIdA
5. Bob点击按钮getTransaction 获得了Alice提走资产时输入的hashsecret:hashsecretAB
6. 有了txId和hashsecret 参照/withdraw的步骤Bob就能提走Aclice锁在chainA合约里的资产

#### （4）/refund (http://localhost/#/refund)

1. 如果超过了时间T1 Alice锁在合约中的chainA资产还没被提走，这时Alice就可以利用refund操作取回自己锁在chainA链上的资产。打开/refund路由，并选择chainA链
2. Alice选择资产锁定时的交易账户addressA1
3. Alice输入txId：txIdA
4. Alice点击按钮refund提走锁定在chainA中的资产
5. 同样，如果超过了时间T2，Bob锁定在chainB中的资产并未被提走，Bob可以按照一样的操作提走锁定的资产

**二，实践例子**

**1)前提说明**

Alice在ChainA上的地址addressA1: 0xaFEaE7a7b899C5068fB5C62b57C546aA5CDb2279

Bob在ChainA上的地址addressB1: 0x4eb0333A6A02b181c551A6E84279EeEAcB00E5F3

Alice在ChainB上的地址addressA2: 0x070C3160753fe41aC939D15B5adC2dCEa6274D23

Bob在ChainB上的地址addressB2: 0x4E668deee2A22aa126ae6813D37dE134236868b5

四个账户上的初始余额都是100

Alice想用chainA链上的10个token换Bob chainB链上的10个token

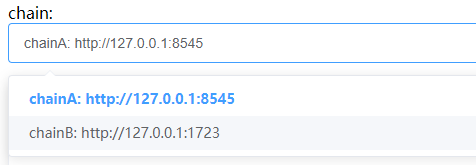
正好Bob想要用chainB上的10个token换chainA的10个token

那交易完成后Alice的chainB上的账户addressA2会有将近110个token（因为要扣除交易手续费）

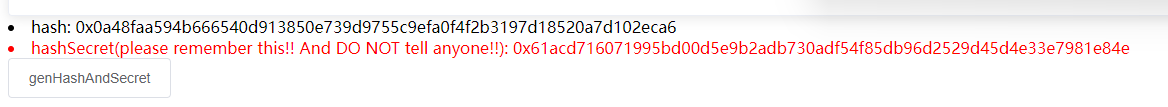
交易完成后Bob的chainA链上的账户addressB1会有将近110个token

**2）实践操作**

1，Alice在浏览器输入：<http://localhost/#/newTx> 并选择chainA



2，Alice点击genHashAndSecret生成hash和hashSecret



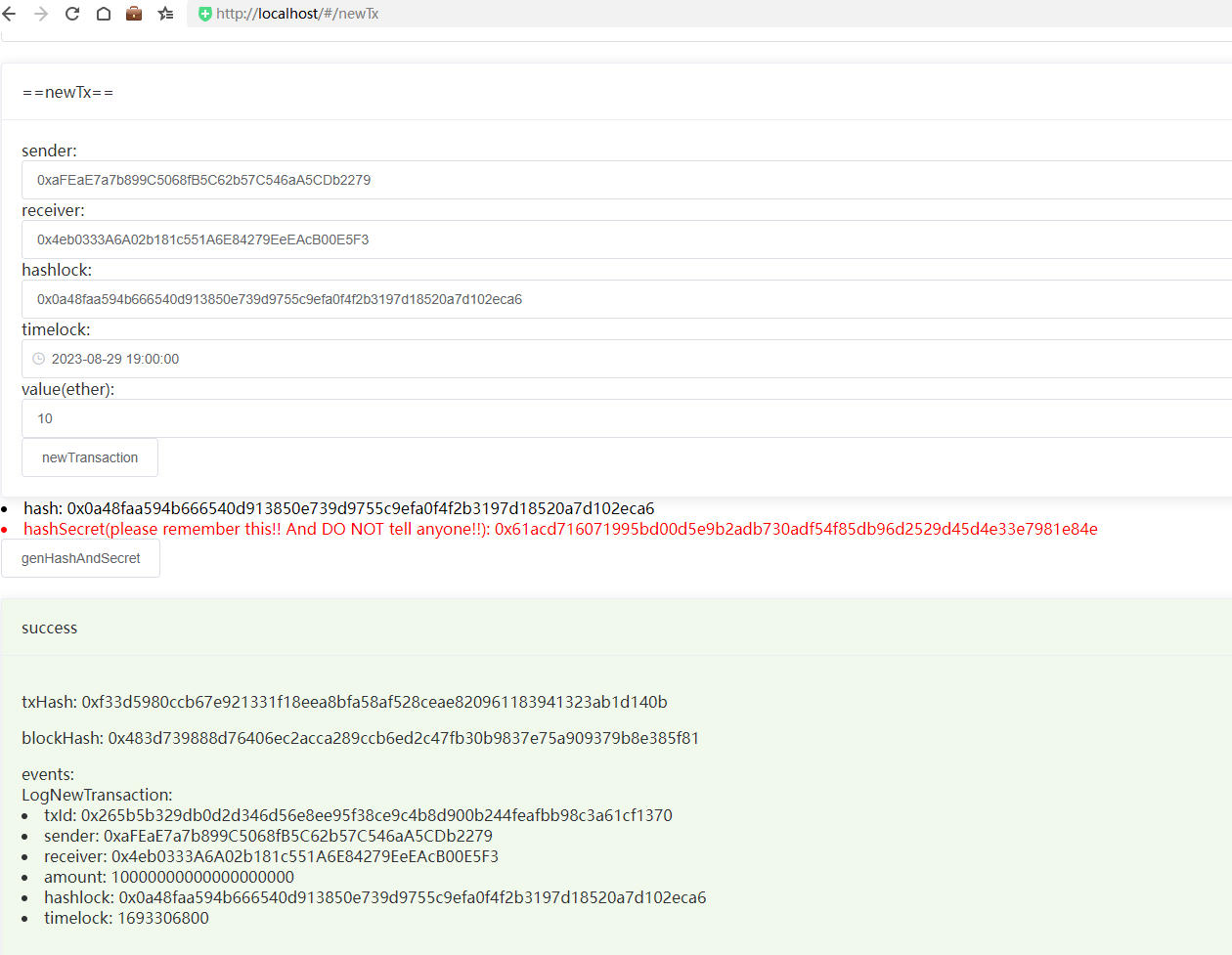
 hash: 0x0a48faa594b666540d913850e739d9755c9efa0f4f2b3197d18520a7d102eca6

 hashSecret: 0x61acd716071995bd00d5e9b2adb730adf54f85db96d2529d45d4e33e7981e84e

3，Alice选择时间T1:



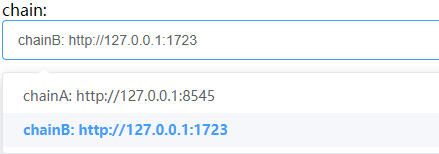
4，输入要转换的token数量10，然后点击newTransaction



1. 得到成功的结果后记录txID：

txId: 0x265b5b329db0d2d346d56e8ee95f38ce9c4b8d900b244feafbb98c3a61cf1370

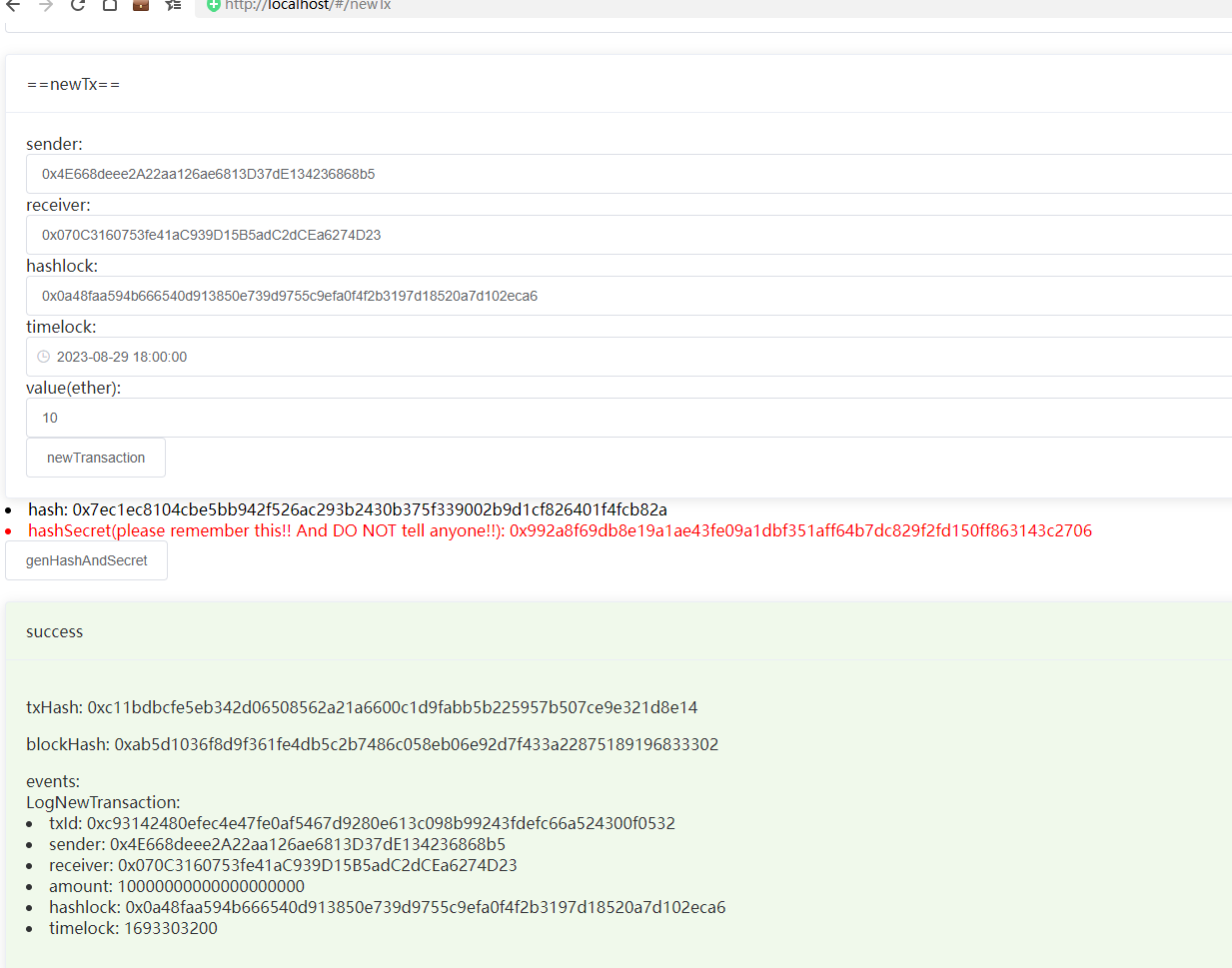
1. Bob在浏览器输入<http://localhost/#/newTx>，便选择chainB



1. Bob输入addressB2作为sender, addressA2作为receiver,并输入Alice发过来的hashlock: 0x0a48faa594b666540d913850e739d9755c9efa0f4f2b3197d18520a7d102eca6,输入value:10
2. Bob选择一个时间T2



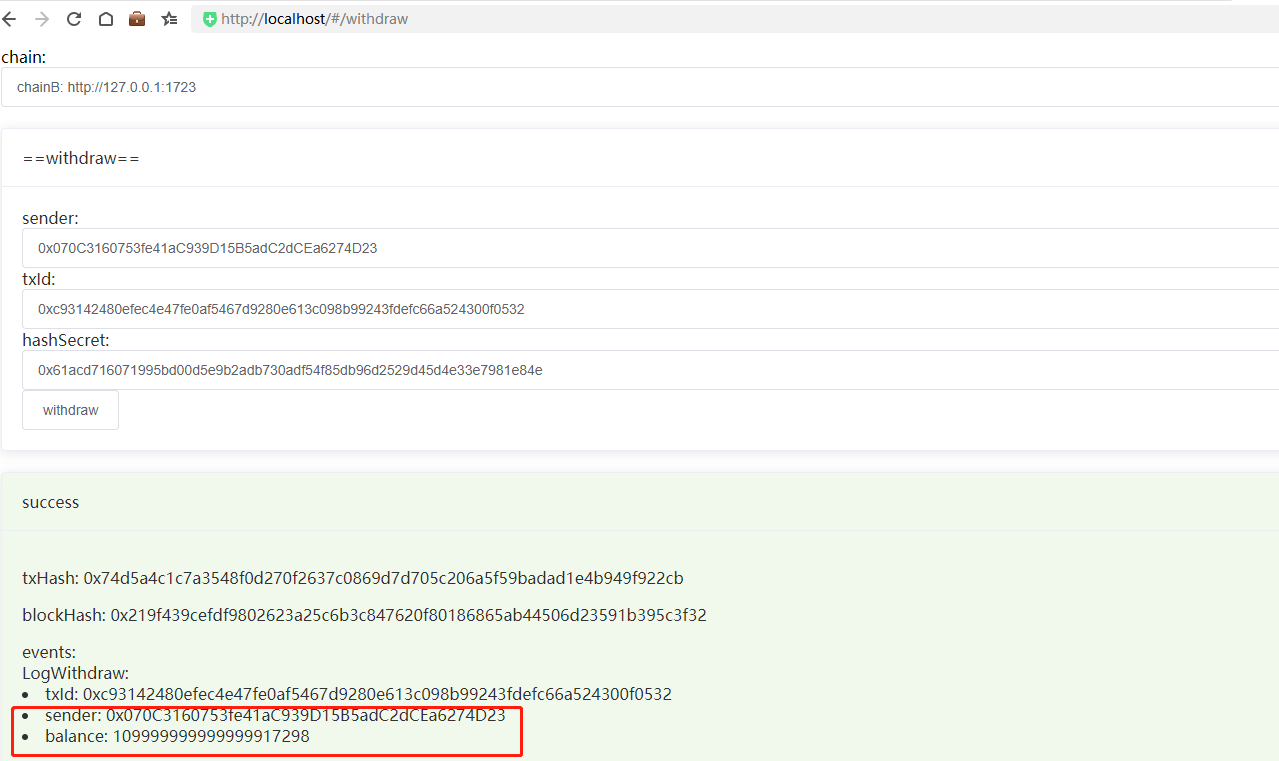
1. Bob点击newTransaction



1. Bob记录返回的结果里的txId:

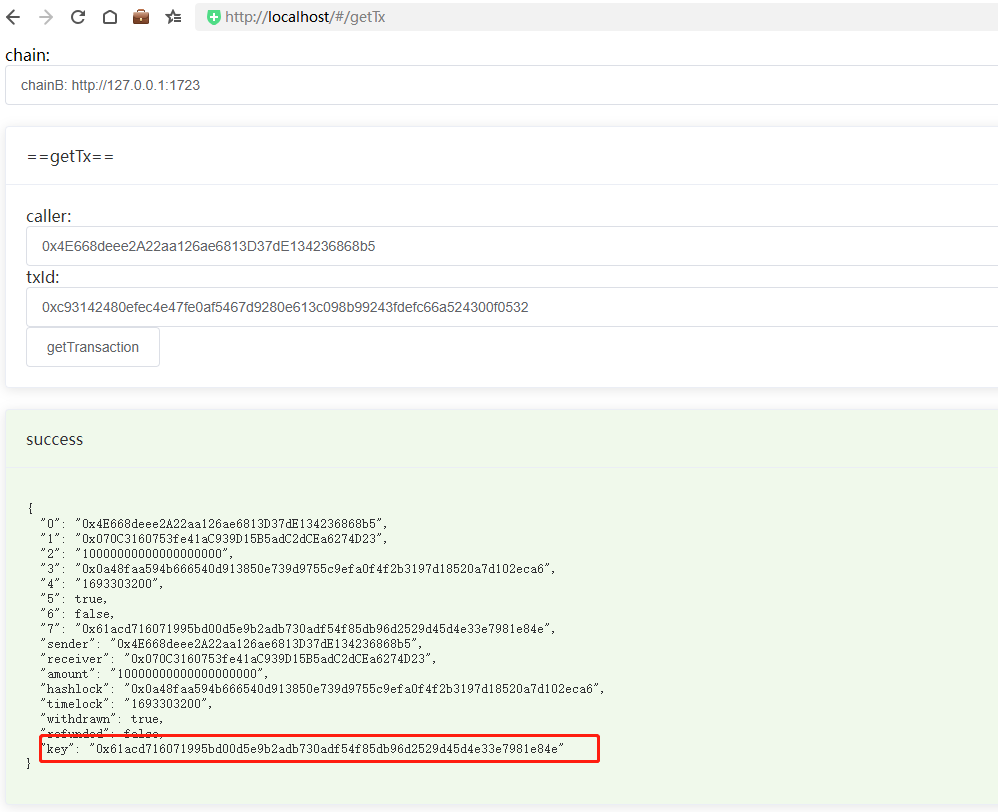
0xc93142480efec4e47fe0af5467d9280e613c098b99243fdefc66a524300f0532并把txId和T2发给Alice

1. Alice 在浏览器中输入http://localhost/#/withdraw ，并选择chainB
2. Alice输入sender:addressA2,txId和hashSecret，点击withdraw



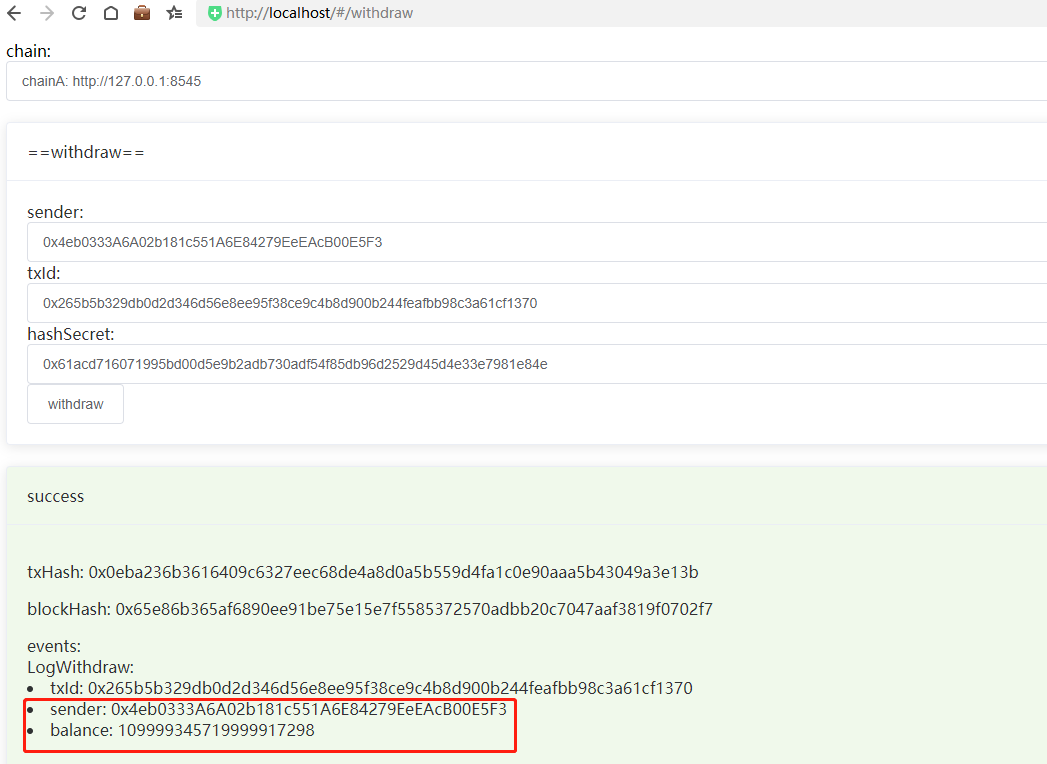
可以看到返回的账户余额为将近110个token，至此Alice提走了自己的ChainB的token

1. 因为Bob监听了chainB的事件，所以Bob及时知道了Alice提币的事件，Bob可以调用接口getTx获取Alice提币输入的hashSecret
2. Bob在浏览器输入：<http://localhost/#/getTx>,并选择chainB
3. Bob输入caller（addressB2）和txID，点击getTransaction



根据返回的结果，Bob拿到了hashSecret:0x61acd716071995bd00d5e9b2adb730adf54f85db96d2529d45d4e33e7981e84e

1. Bob在浏览器输入： [http://localhost/#/withdraw](http://localhost/#/getTx) ,并选择chainA
2. Bob输入sender:AddresB1, txID和hashSecret，点击withdraw



从返回的成功的结果中看到Bob的账户addressB1的余额将近110，至此，Alice和Bob利用hash时间锁合约完成了从链chainA到链chanB的跨链转账。