

区块链基础及应用 2024 Exercise 4

在本次任务中，你需要创建一个称为**跨链原子交换**的交易，允许两个实体在不同的区块链上安全地交换加密货币的所有权。同样将使用 `python bitcoinlib` 提供启动代码。建议使用 **Linux** 操作系统。

1 介绍

原子跨链交换

在这项作业中，你需要实现 Alice 和 Bob 两方之间跨链原子交换代码的关键部分。Alice 在 BTC Testnet3 上有比特币，这是 `project1` 使用的标准比特币测试网。Bob 在 BCY Testnet 上拥有比特币，BCY Testnet 是 Blockcypher 的比特币测试网，由 Blockcypher 独家挖矿和维护。他们希望安全地交换各自 `coin` 的所有权，这是一个简单交易无法完成的事情，因为它们位于不同的区块链上。

这里的想法是围绕一个只有一方（Alice）知道的秘密 x 建立交易。在这些事务中，只有 $H(x)$ 将被发布，而 x 为秘密。交易将以这样的方式建立，一旦 x 被揭露，双方都可以赎回对方发送的硬币。如果 x 永远不会被揭露，双方将能够安全地取回他们的原始硬币，而不需要另一方的帮助。

这种方法也适用于其他加密货币。

1.2 如何工作？

请参考比特币 `wiki` 页面中的原子跨链交易：

https://en.bitcoin.it/wiki/Atomic_cross-chain_trading。

以便更好地理解我们在这个 `project` 中需要做什么。

2 安装程序

1. 运行课程代码 `pip install -r requirements.txt` 安装所需的依赖项。确保使用的是 `python3`。
2. （a）为 Alice 和 Bob 创建 BTC testnet 密钥。你可以用 `keygen.py` 生成密钥，把它填入 `keys.py` 中合适的地方。
（b）在 Project1 中相同的 `coina faucet` 上，<https://coinafaucet.eu/en/btc-testnet/>，为 Alice 的 BTC 地址领取测试币。
3. （a）在 Blockcypher 注册帐户以获取 API token: <https://accounts.blockcypher.com/>。
（b）为 Alice 和 Bob 创建 BCY testnet 密钥并填入 `keys.py`。

```
curl -X POST https://api.blockcypher.com/v1/bcy/test/addr?token=YOURTOKEN
```

(c) 在 Blockcypher 测试网 (BCY) 上为 Bob 的 BCY 地址领取测试币。

```
curl -d '{"address":"BOB_BCY_ADDRESS", "amount": 1000000}'  
https://api.blockcypher.com/v1/bcy/test/faucet?token=YOURTOKEN
```

4. 使用 `split_test_coins.py` (填写文件中的相关字段) 划分领取的币。
5. 填写 `swap.py`.
6. 阅读 `swap.py`, `alice.py`, 和 `bob.py`, 以及 https://en.bitcoin.it/wiki/Atomic_cross-chain_trading 中的伪代码, 完善 `swap_scripts.py` 中的脚本。

3 提交你的代码

请提交此作业的所有代码以及 `design_doc.txt`。当 `swap.py` 使用 `broadcast_transactions=False` 运行时, 进行本地验证。请创建一个 `.zip` 文件, 包含本次作业需要的文档。通过邮箱提交。

4 练习

- A. 考虑创建跨链原子交换所需事务所需的 `ScriptPubKey`。此交易必须可由接收者赎回 (如果他们有一个与 `Hash(x)` 对应的秘密 `x`), 或者可以用发送者和接收者的两个签名赎回。

完善 `swap_scripts.py` 中的脚本 `coinExchangeScript`。

- B. 完善脚本:

(a) 在接收者知道秘密 `x` 的情况下, 编写赎回交易所需的 `ScriptSig`。在 `swap_scripts.py` 中完善 `coinExchangeScriptSig1`。

(b) 在发送方和接收方都签署事务的情况下, 编写赎回事务所需的 `ScriptSig`。在 `swap_scripts.py` 中完善 `coinExchangeScriptSig2`。

- C. 运行你的代码 `swap.py`。注意, 文件中需要填写区块高度, 代码注释中的方法不可用, 请使用以下地址, BTC 高度查询地址: <https://live.blockcypher.com/btc-testnet/>, BCY 区块高度查询地址: <https://live.blockcypher.com/bcy/>

我们不需要广播事务, 因为这需要等待一些时间来验证。设置 `broadcast_transactions=False`。将本地验证 `ScriptSig+ScriptPK` 是否返回 `true`。将 `alice_redeems=True` 和 `alice_redeems=False` 分别尝试此操作。

可选: 尝试使用 `broadcast_transactions=True`, 这将使代码休眠一段适当的时间, 以便将所有内容发布到区块链并正确验证。需要 20-60 分钟甚至更长时间才能运行。

D. 请写一个简短的关于这个项目的**设计文档**。需要包括以下内容：

- (a) 解释你写的代码内容，以及 `coinExchangeScript` 是如何工作的。
- (b) 以 Alice 用 `coinExchangeScript` 向 Bob 发送硬币为例：
 - 如果 Bob 不把钱赎回来，Alice 为什么总能拿回她的钱？
 - 为什么不能用简单的 1/2 multisig 来解决这个问题？
- (c) 解释 Alice (Bob) 创建的一些交易内容和先后次序，以及背后的设计原理。
- (d) 以该作业为例，一次成功的跨链原子交换中，数字货币是如何流转的？如果失败，数字货币又是如何流转的？