Public Ledger for Auctions

Cristiana Silva up201505454
Nuno Tomás up201503467
Rui Santos up201805317

5 de junho de 2022

1 Introdução

Com o aparecimento de sistemas *blockchain* baseados em criptomoedas o uso de *public ledgers* ganhou popularidade como um mecanismo de registo de identidades de participantes, os seus saldos e transações válidas realizadas. Neste trabalho foi-nos pedido para aplicar este conceito de *public ledger* a um sistema de leilão com compradores e vendedores, onde registamos e verificamos as transações através do *public ledger* e espalhamos informação aos participantes com uma rede P2P baseada em Kademlia. Neste relatório vamos falar sobre os vários componentes do nosso sistema, como os implementamos e que escolhas fizemos. Terminamos por refletir sobre como decorreu o desenvolvimento deste trabalho e as dificuldades que tivemos.

2 Componentes do sistema

Nesta secção vamos falar sobre os vários componentes do nosso sistema e explicamos como os implementamos. A figura abaixo dá-nos uma visão geral sobre a arquitetura do sistema.

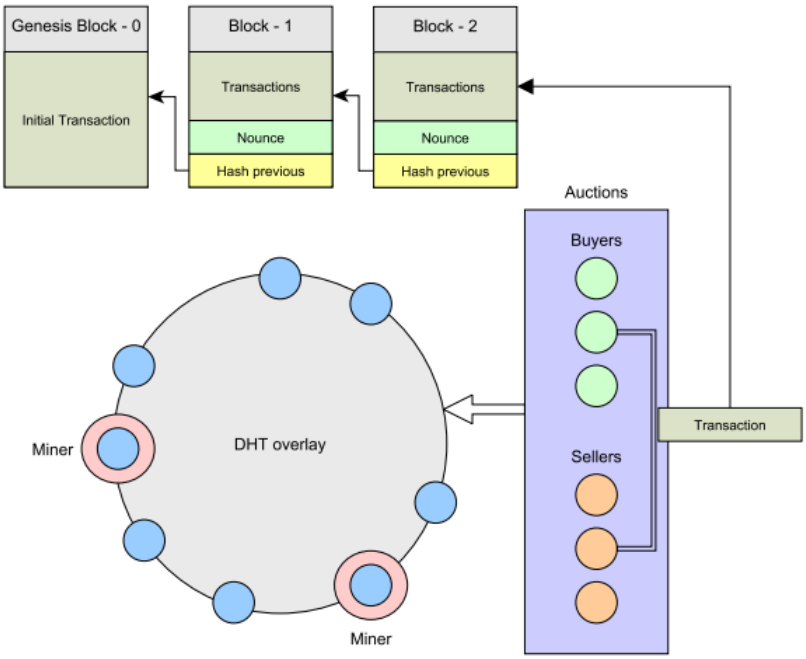

Figura 1: Visão geral do sistema

2.1 Camada P2P segura

A rede P2P é responsável por guardar e espalhar informação necessária para suportar a *blockchain*.

2.1.1 Kademlia

Para a rede P2P utilizamos o Kademlia uma *peer-to-peer distributed hash table*, que sendo descentralizada, remove um ponto de falha central tornando-a mais robusta a certos ataques. Implementámos o Kademlia seguindo este artigo[1]. O protocolo Kademlia consiste de quatro RPCs :

- PING : verifica se um nó está online
- STORE : guarda um par chave-valor num nó
- FIND NODE : retorna informação sobre os k nós mais próximos de um id
- FIND VALUE : semelhante ao FIND NODE, mas se o nó tiver uma chave associada, retorna o valor guardado

No nosso sistema cada nó tem um id, um endereço e uma porta e podemos calcular a distância entre dois nós aplicando um xor entre os *nodeIds* deles. Para implementar os diferentes RPCs utilizamos o gRPC, criando os ficheiros *proto* necessários. Acabamos por só conseguir implementar o PING.

2.1.2 Proteção contra ataques Sybil e Eclipse

Não chegamos a implementar proteção contra ataques de Sybil e de Eclipse, mas uma das possíveis soluções seria recorrer a uma *Certification Authority* (CA) centralizada [2].

2.2 Distributed Ledger

A nossa *distributed ledger* implementa uma *blockchain* baseada em *proof-of-work*.
Ao criar a *blockchain*, nenhum dos utilizadores começa com moedas e por isso não é possível fazer transações entre utilizadores. Por esta razão começamos a *blockchain* com um bloco *genesis* que quando minado distribui algumas moedas a novos utilizadores. Cada bloco contém a *hash* anterior, um *nonce*, um *timestamp* e raíz da *Merkle Tree* que são usados para criar uma nova hash e mais tarde para minar o bloco. Cada utilizador pode também criar uma carteira, que tem uma chave pública e uma privada utilizadas nas transações, bem como a sua lista de UTXOs , a qual nós chamamos *unspentTrans*. Conseguimos que a *blockchain* funcionasse corretamente.

2.2.1 Proof of Work

Proof-of-Work é uma forma de validar transações. Durante o processo de mineração, os vários *miners* competem entre si para validar as transações e usá-las para criar um novo bloco na *chain*, o primeiro *miner* a fazê-lo é recompensado. No nosso sistema os *miners* devem encontrar a *hash* correta do bloco, para isso devem fazer computações para calcular uma *hash* com um certo número de zeros iniciais dada pela dificuldade. Quanto maior a dificuldade mais difícil fica este processo, o que desencoraja os *miners* de validarem transações falsas.

2.3 Leilão

Num leilão, um utilizador deve conseguir colocar um item à venda e enviar esse anúncio a outros nós da rede de forma a que nós interessados nesse item consigam licitar nele. Quando um dos participantes ganha o leilão ele deve fazer uma transação para o vendedor e só quando a transação é verificada,é que o vencedor receberá o seu item. Por isso é importante todo este sistema ser construído sobre o Kademlia.

2.3.1 Transações

As nossas transações correspondem a transferências de moedas entre carteiras. Cada transação contém a chave pública do emissor e do recetor, a quantidade a ser transferida e uma lista de inputs, que são referências a transações antigas que provam que o emissor tem moedas para gastar. As transações são ainda assinadas com a chave privada de forma a prevenir que as informações da transação sejam alteradas.

2.3.2 Sistema de publish/subscribe

Como indicado em cima, o sistema de leilão devia ser um sistema de *publish* (publicar o anúncio do item que se quer vender) e *subscribe* (participantes subscrevem a itens a que estão interessados para poder licitar neles).Infelizmente, como não conseguimos implementar por completo o Kademlia, também não conseguimos construir este sistema de leilão.

3 Conclusões

Embora a *blockchain* esteja plenamente funcional e com *proof-of-work*, as dificuldades que encontramos com gRPC não nos permitiu implementar o Kademlia completamente (apenas a função PING) e consequentemente os leilões também não. Como nunca trabalhamos com gRPC esta parte do trabalho revelou-se mais complexa do que esperávamos.

Referências

[1] P. Maymounkov and D. Mazieres. Kademlia: A peer-to-peer information system based on the xor metric.

[2] R. Pecori. S-kademlia: A trust and reputation method to mitigate a sybil attack in kademlia.