Azulik whitepaper first draft

Azulik Group

January 13, 2018

Abstract

Raiblocks outlined the concept of a "block lattice", where each account has it's own blockchain and each "block" contains just one transaction. Raiblocks is a very elegant protocol allowing instant and fee-less transactions. Raiblocks has everything to become the next Bitcoin, and as every great product needs an alternative we propose in this draft a potential approach allowing to enable the privacy feature. Zero-knowledge proof systems is well documented, however it has been rarely applied to directed acyclic graph especially with the view to transfer value instantaneously and without fees. This draft try to explain the problematic and a potential approach. Further approach have been or will be considered to solve the problem and allow to create an Anonymous Raiblock fork.

1 Disclaimer

The following draft isn't a whitepaper as you use to read. Indeed, it focus on applying privacy to cryptocurrency based on directed acyclic graph and especially the Raiblocks. We hope it will allow to start the discussion with the community about the feasibility to have an anonymous Raiblocks. The authors shall not be held liable for the up-to-dateness, correctness, completeness or quality of the information provided. This is solely a development project. By participating in this endeavor you renounce any legal rights covering losses that may result from participating, you hereby attest that you will not, under any form hold any other team members, legally responsible. Any donations will be treated as such: "Donation is the act by which the owner of a thing voluntarily transfers the title and possession of the same from himself to another person, without any consideration; a gift. A donation is never perfected until it is has been accepted, for the acceptance is requisite to make the donation complete. The person making the gift is called the donor and the person receiving the gift is called the donee.." This version is a very first draft.

2 Introduction

In cryptocurrency environment, the mobility of the network users necessitates an agile authentication system. Zero-knowledge proof systems allow an interaction between parties to determine trustworthiness in a quick and effective manner. In order to make these interactions as fast and secure as possible, they are most often based on problems from the NP-complete class (NP, or "nondeterministic polynomial," is the complexity class of problems with "polynomial time verifiers." I.e., if the problem has a solution, it must be possible to verify the solution in polynomial time, even if finding the solution is much harder.), which contains many graph theory problems. A strong and lightweight zero-knowledge protocol must satisfy the following criterion: it must have a small number of bits transferred between parties, it must require few iterations to achieve a given trust level, and it must be difficult for a cheater to pass as trustworthy.

3 Background

3.1 Graph theory background

This section is meant as a guide to some of the graph theoretic terms and concepts employed in this whitepaper draft. A graph is a pair $G = (V, E)$ such that E is a subset of VxV, where V is the set of vertices and E is the set of edges in the graph. Vertices can also be called nodes especially in the cryptocurrency field. An edge is incident to a vertex if the vertex is one of the

edge's endpoints. Two vertices are adjacent if they are connected by an edge. The degree of a vertex (or valency) is the number of edges incident to it. An adjacency matrix representation of a graph is a matrix in which the rows and columns represent the vertices and an entry equal to 1 in row and u column implies the existence of an edge between vertices and, while an entry equal to 0 implies that there is no edge between u and v.

3.2 NP-Completeness

The class NP is the class of decision problems for which any yes-instance has a solution that is verifiable in polynomial time. The class P contains all decision problems that can be solved in polynomial time, and hence also have solutions that can be verified in polynomial time, implying that P $\subseteq$ NP. A problem L in the class NP is in the subclass of NP-complete problems if every problem in NP can be reduced to the problem L in polynomial time. A reduction from problem K to problem L is an algorithm which takes as input an arbitrary instance of problem K and outputs an instance of problem L. Given this definition, it is clear that the class of NP-complete problems contains the hardest problems in the class NP, as an easy solution for one NP-complete problem leads to an easy solution for all problems in the class NP. In order to prove a problem is in the class of NP-complete problems, we need only prove that a known NP-complete problem L reduces to our problem.

3.3 Zero-Knowledge Proof Systems

We begin with the notion of an interactive proof system which is the key for privacy. An interactive proof system is an interaction between two participants, called the prover and the verifier, in which the prover attempts to prove some fact (or knowledge of some private input) to the verifier. An interactive proof system is formally defined as a protocol based on a decision problem which satisfies the following properties:

1. Completeness: Each yes-instance of the decision problem leads to acceptance by the verifier with probability at least $1 - n^{-k}$ for any constant $k > 0$, where is the size of the problem instance.

2. Soundness: Each no-instance of the decision problem leads to rejection by the verifier with probability at least $1 - n^{-k}$ for any prover.

An important component of a zero-knowledge proof system is the commitment. Zero-knowledge proof systems usually require that the prover has some method of "locking up" information about the problem instance prior to receiving the verifier's challenge. Otherwise, the prover would be able to manufacture a response to the verifier's challenge, and this new response may or may not be consistent with the problem instance.

4 Zero-Knowledge Proofs applied to Raiblocks

4.1 First approach proposed

The longest path problem for a general graph is not as easy as the shortest path problem because the longest path problem doesn't have optimal substructure property. In fact, the Longest Path problem is NP-Hard for a general graph. However, the longest path problem has a linear time solution for directed acyclic graphs.
The above figure illustrates the step by step long path problem. A zero-knowledge proof system for the longest path problem could hence be applied. The common inputs to the protocol are a graph G, and a positive integer k, which represents the length of a longest path in G. The private input is the longest path itself. The prover (P) chooses randomly an isomorphism π to permute the graph G and then sends a commitment to this new graph π(G), , to the verifier (V). V then chooses a random bit c which is sent to P. If c=0, P sends π to V along with the decommitment information for π(G) and V checks that π(G) was formed correctly from G and π. If c=1, then P sends the decommitment information for π(P) to V (where π(P) represents the entries corresponding to the edges of the path that is the private input) and V checks that π(P) forms a path of the specified length. the prover does not need to send any information in addition to the edges of the path

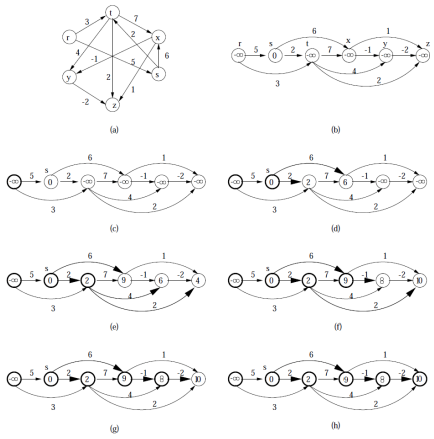

Figure 1: Long path problem decomposition

to the verifier. The permutations used by the prover are unnecessary information for the verifier, as checking that the edges revealed form a path is a simple task without the knowledge of the permutations. The prover also does not need to identify the vertices that are endpoints on the path, as the verifier can determine these from the revealed entries by examining which rows and columns have one and only one entry equal to 1.
This figure illustrate a possible way to apply a zero-knowledge proof system and hence privacy to DAG. However even if we believe this approach will allow fee-less transaction they won't be instant anymore (at this stage of our reflection). We could hence think about a protocol allowing privacy as on option. We will soon develope a WIP approach.

To be continued ...

Thanks for reading

Azulik Group

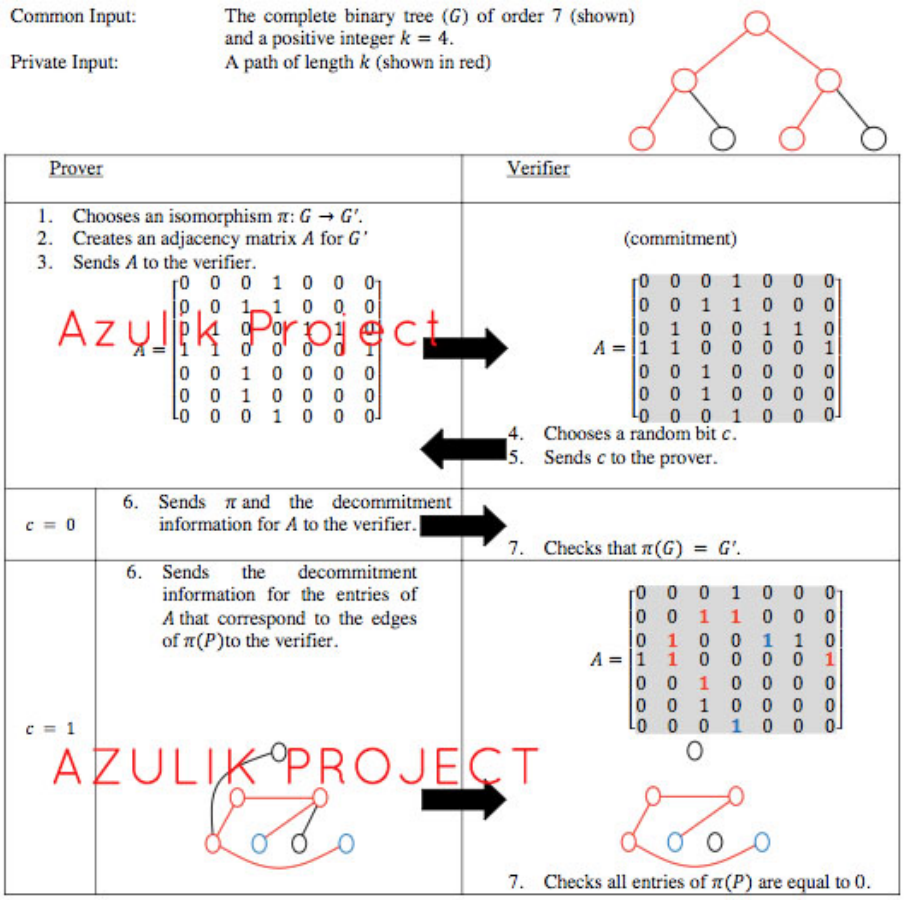

Figure 2: Hamiltonian cycle illustration