AFRICAN INSTITUTE FOR MATHEMATICAL SCIENCES

(AIMS RWANDA, KIGALI)

Name: Sittana Osman Afifi Mohamedelmubarak ch 1
Zero Knowledge proof Date: May 11, 2020

1 Introduction

Recently many companies and banks ask their clients about personal data like their contact information, bank receipt, their tastes, or even their behavior patterns. This information is very valuable; A lot of companies try to collect as much information about their clients as they can, so they'll be able to sell more ads, give better recommendations or keep users longer on their platforms. This is a problem since users can't leave these companies to keep their privacy. Here is where zero-knowledge proof comes in, for example, users can prove to the bank that they have sufficient funds to get loans without telling them exactly how much they have.

1.1 Research Objectives

1. To explore Zero-Knowledge proof protocol and mathematics behind it.
2. To demonstrate Zero-Knowledge proof protocol for graph isomorphism using python.
3. To study real application of ZKP protocol in blockchain

1.2 Brief History of ZKP

Zero-knowledge was introduced in 1989 by Shafi Goldwasser, Silvio Micali, and Charles Rackoff in their paper "The Knowledge Complexity of Interactive Proof-System"[1] The first version of their paper has existed since 1982 and was rejected three times from major conferences (FOCS '83, STOC '84, FOCS '84) before appearing in STOC '85. Goldreich, Micali, and Wigderson showed how to construct zero-knowledge systems from any NP-set.

1.3 A very simple definition

Zero-knowlege proof (ZKP) is a way of handling data verification without revealing that piece of data.
Basically, we have two parts:

1. **Prover**: the part that knows the secret and wants to prove to the second party that he knows the secret.
2. **Verifier**: the part that wants to be convinced that the prover has the secret.

A Zero-Knowledge Proof must satisfy three properties:[3]

1. **Completeness**: if the statement is true and prover and verifier are both honest(applying the protocol probably) then the verifier will be convinced by the fact.
2. **Soundness**: if the statement is false no verifier will be convinced by a cheating prover this statement is true, except with a small probability.
3. **Zero-Knowledge**: the prover does not leak any information more than that he knows the secret.this ensured by necessitating that the verifier can do the protocol alone without knowing any additional information other than from its interaction with the prover.

There are main two types of zero-knowledge proof:[5]

1. **Interactive zero-knowledge proof**: This type includes interchanging several messages between the prover and the verifier.
2. **Non-interactive zero-knowledge proof**: In this type, there is no interaction, it contains only a single flow from the prover to the verifier.

1.4 Simple example (Two balls and the color-blind friend)

Imagine you have a friend (the verifier) who is color-blind and can't see the difference between a green and a red ball, to him, the balls have the same color and you (the prover) want to prove to him that they are in fact different. He doesn't need to know which is red and which is green, just whether or not they are different. So you give him the two balls and take note of which ball is in which hand. Then, he puts the balls behind his back and chooses to either switch them around or not. After that, he shows them to you again. Now you have to tell him whether or not the balls have switched hands. If they were the same color you would not be able to differentiate between the two options. So, there would be no way you could guess correctly with a probability of more than 0.5 since the probability that you can guess randomly the right answer is 50%, but this probabilty will reduce to an acceptable level if we repeat this process as often as necessary.
This proof is zero-knowledge protocol because your friend just learns there are two different balls but he doesn't know which one is red and which one is green.

1.5 Application of ZKP

1. **Authentication systems**: ZKP help uses the user who wants to demonstrate its identity (password for example) via some secret to a second party without learning anything about this secret.
2. **Ethical behavior**: ZKP uses to oblige a user to prove that his behavior is correct according to the protocol, because ZKP saves the privacy of the user's secret during the process of providing the proof.
3. **Nuclear disarmament**: ZKP assists inspectors to confirm whether or not an object is indeed a nuclear weapon without recording, sharing, or revealing the internal workings which might be secret.
4. **Blockchain**: by using ZKP we can perform a valid transaction with keeping the sender, the recipient, and all other transaction details remain hidden.

2 Mathematical Background:

In the previous swction, we gave a brief introduction about ZKP, in this chapter, we introduce some concepts in graph theory and mathematical aspect of ZKP,in addition Graph Isomorphism based Zero-Knowledge Proofs.

2.1 Zero-Knowledge Proof

Definition 1. *(zero-knowlege proof):[3] Suppose L is a language. A zero-knowledge protocol is an interaction between two probabilistic polynomial-time algorithms P and V with P trying to convince V that $x \in L$ and satisfying properties (1), (2), and (3) described below.:-[3]*

1. ***Completeness**: If $x \in L$, w is the correct witness, and the protocol is honestly executed, then V outputs ACCEPT.[3]*
2. ***Soundness**: if $x \notin L$, for every probabilistic polynomial time algorithm $\hat{P}$, there is a negligible function $negl(\cdot)$ such that:*

$$Pr[\hat{P} \text{ convinces } V \text{ that } x \in L] \leq negl(\cdot).[3]$$

3. ***Zero-knowledge**: This definition represents the idea that P does not leak any information by necessitating that V can perform the protocol "alone", and if V can perform the protocol alone, then V does not learn any additional information from its interaction with P.This intuition is formalized by necessitating that there is a standalone simulator S which can produce a transcript which is indistinguishable from that of P and V.*
   Formally, if for all $x \in L$ there is a probabilistic polynomial time algorithm S which can output a transcript $\acute{\tau}$ such that $\tau \approx_c \acute{\tau}$, where τ is the distribution of the original interaction(between P and V) transcript. The algorithm S is often called a simulator.[3]