ELEKTROTEHNIŠKI VESTNIK 78(1-2):
ENGLISH EDITION

Zero knowledge-proof

Ante Sosa
Erasmus student

Fakulteta za računalnistvo in informatiko, Univerza v Ljubljani
E-mail: as2473@student.uni-lj.si

Abstract. Zero knowledge-proof is a method by which one party can prove the other party that some statement is true, without reveling statement or any information about statement. The requirement is that the verifier is honest i.e. is following the protocol of verifying. Zero knowledge-proofs are applied to various fields like authentication systems, ethical behaviour, blockchains and many others. In this paper the idea of zero knowledge-proof will be introduced as well as application to blockchain.

Key words: Zero knowledge-proof, Cryptography, Blockchain, Privacy

1 INTRODUCTION

A zero knowledge proof must satisfy following three properties:

- Completeness
- Soundness
- Zero-knowledge

The completeness stands that if statement is true the honest prover can convince honest verifier that statement is true. The soundness stands that if statement is false no prover can convince honest verifier that statement is true except with some small probability. The zero-knowledge stands that there is no additional information leak during the proces of proving the statement. In the other words only true statement is enough to prove it without reveling any other information about it.
Here is a list of some problems that can be proved while being not revealed:

- Two balls and the colour-blind verifier
- Yao's millionaires problem
- Discrete logarithm of a given value

1.1 Two balls and the colour-blind verifier problem

In this problem the prover wants to convince color-blind verifier that two balls are different colour. The prover is able to do that by giving the balls to verifier who can only feel that balls are same shape, but have no information about color of the balls. The verifier hides the balls and shows one to the verifier. After prover have seen the ball, verifier again hides the balls and shows either the same ball or the other ball. Prover tells the verifier that ball is switched or not switched, depending on what he saw. After prover has guessed correctly, verifier is still not convinced that prover can distinguish colors of the balls, since there is 50% chance that he guessed it by chance. If they repeat the process 20 times the probability that the prover guessed it by chance is 1 to million.

1.2 Yao's millionaires problem

This problem introduces two rich persons which want to know which one is richer without revealing their actual balances. The solution is given by the protocol of 0-encoding and 1-encoding published in [1].
Let $s = s_n s_{n-1} \dots s_1 \in \{0,1\}^n$ be a binary string of length n. The 0-encoding of string s is set $S_s^0 = \{s_n s_{n-1} \dots s_{i+1} 1 | s_i = 0, 1 \leq i \leq n\}$. The 1-encoding of the same string is set $S_s^0 = \{s_n s_{n-1} \dots s_i | s_i = 1, 1 \leq i \leq n\}$.
If x is encoded in S_x^1 and y is encoded in S_y^0 than $x > y$ if and only if S_x^1 and S_y^0 has common element.
For example let $x = 11 = 1011_2, y = 9 = 1001_2$. Then $S_x^1 = \{1011, 101, 1\}$ and $S_y^0 = \{101, 11\}$. Clearly $S_x^1 \cap S_y^0 \neq \emptyset$, so we have proved that $x > y$ without revealing exact numbers x, y.

1.3 Discrete logarithm of a given value

The prover wants to convince the honest verifier that he knows solution to Discrete logarithm problem. More specific, he wants to prove that he knows x such that $\alpha^x \equiv \beta \pmod N$ without actually revealing x to the verifier. Both prover and verifier know the values of α, β, but x is known only by prover. The protocol proceeds as follows:

1) The prover chooses by random $r \in \{1, \dots, \phi(N)\}$, calculates $\gamma :\equiv \beta^r \pmod N$ and sends it to verifier.
2) The verifier chooses $b \in \{0, 1\}$.
3) Prover sends $y :\equiv r + bx \pmod N$ to verifier.
4) Verifier checks if $\alpha^y \equiv \alpha^r \beta^b \pmod N$ holds.

Cheating probability is again 50 % for each round, so solution is executing multiple rounds in a sequence, until the probability gets low enough.[2]

2 Application on blockchain

2.1 Bitcoin

Bitcoin, as the most famous blockchain in the world is considered to be anonymous but that is exactly not true. In fact, the users of mentioned network are hidden behind sequence of letters and numbers that are called addresses. Transactions, which include sending and receiving addresses and amount of coins are publicly viewable on the public ledger. If someone connects user's address with his name, he is able to track all transactions user have ever made on the network and also all the future transactions connected with given address. The main problem is the fact that miners can't validate the transaction if address of sender and amount of coins are hidden. In other words, miners can't check if the address has sufficient amount of coins in their address to send proposed amount.

2.2 Zcash

The real privacy is introduced to blockchain by Zcash. Zcash has two kinds of addresses:

- z-addresses - private
- t-addresses - transparent

From any of these two types of addresses, coins could be sent to any address type. While implementing zero knowledge-proof to blockchain the main problem was size of proofs and computational complexity to validate transactions from or to z-addresses, which has been solved using zero-knowledge succinct non-interactive arguments of knowledge, or shorter zk-SNARKS. It is special kind of zero-knowledge proof used by Zcash. The main benefits of applying zk-SNARKs to blockchain are reducing size of proofs and computational effort of verfying them.

The verifying function is a function which takes transaction as input and returns 0/1 output. It is computationally complex, so in order to be implemented on blockchain it has to be reduced to some equivalent algebraic equation, through a chain of reducing operations. The first operation of reducing transaction validation function is breaking down logical steps into as small as possible operations and thus creating arithmetic circuit. The next step is reducing to Rank 1 Constraint System, or shorter R1CS, which checks that each step of computation in previously gotten arithmetic circuit are done correctly. In [3] Quadratic Asymetric Program (QAP) is introduced to reduce checking between numbers to checking between polynomials. At the first glance, reducing from comparing numbers to comparing polynomials seems not to be a reduction at all, but this is not true. Further more, when polynomials are not equal, the identity will fail to hold at most of the points, so it is enough to check that polynomials match at single randomly chosen point. With zk-SNARKs, mathematical techniques like homomorphic encryption and pairings of elliptic curves are used to evaluate polynomials without knowing which point is evaluated. Once when proofs and verifying function are reduced using zk-SNARKs and profs are non-interactive privacy is applicable to blockchain, without loss of scalability.

3 Conclusion

Zero-knowledge proofs introduced a new way of handling most personal information. Zero-knowledge proofs are introduced to blockchain to allow for transactions to be verified while protecting user and transaction privacy. One big benefit of implementing zk-SNARKs, apart from privacy is scaling, which makes the network much more powerful. Although zero-knowledge profs are introduced before thirty years, it is hard to say how far can this technology take us.

References

[1] H. Y. Lin and W. G. Tzeng; An Efficient Solution to The Millionaires's Problem Based on Homomorphic Encryption, 2005.
[2] D. Chaum, J. H. Evertse; J. van de Graaf, An Improved Protocol for Demonstrating Possession of Discrete Logarithms and Some Generalizations. Advances in Cryptology - EuroCrypt '87: Proceedings. 304: 127-141, 1987.
[3] R. Gennaro, C. Gentry, B. Parno, M. Raykova; Quadratic Span Programs and Succinct NIZKs without PCPs, 2012.