Application of 3-dimensional Matching Problem on hypergraphs in Authentication Based on Non-Interactive Zero-Knowledge Proof protocols

Zahra Tondpour
Iran University of Science and Technology
University St., Hengam St., Resalat Square, Tehran, Iran
ztondpour@gmail.com

ABSTRACT

This paper describes a new protocol for authentication in insecure environments which allows a receiver of a message to verify sender without expose any secret information during protocol. The first, this protocol is described based on the concept of Zero-Knowledge Proof using 3-dimensional matching problem on hypergraphs. Then, for more security and lower interactive complexity, described protocol is converted to a Non-Interactive Zero-Knowledge Proof protocol using the Fiat-Shamir heuristic.

KEYWORDS: Zero-Knowledge Proof, Non-Interactive Zero-Knowledge Proof, Authentication, Graph, Hypergraph, Matching, NP problem.

1 INTRODUCTION

Zero-Knowledge Proof (ZKP) is one of the most used fundamental building blocks in cryptography. ZKP protocol is an interactive protocol between two parties, Prover and Verifier (sender and receiver of message, respectively) where the goal of the Prover is to convince the Verifier that she/he knows password without expose any secret information during protocol. ZKP protocols have many applications in authentication, identification and other basic cryptographic operations. Currently ZKP protocols use in most major blockchains like Ethereum and Bitcoin.

The concept of Zero-Knowledge was introduced by Goldwasser, Micali and Rackoff [5]. The early version of their paper has existed as early as in 1982, and were rejected three times from major conferences (FOCS '83, STOC '84, FOCS '84) before appearing in STOC '85.

Goldreich's paper "A short Tutorial of Zero-Knowledge" is an excellent reference for studying ZKP protocols. The earliest version of the paper appeared in 2002 titled "Zero-Knowledge twenty years after its invention" [3]. In addition, Aaronson [1] discusses Zero-Knowledge proofs from a Philosophical point of view.

Also, Goldreich, Micali and Wigderson [4] showed how to construct Zero-Knowledge systems from any NP problem.

2 ZKP PROTOCOL

ZKP protocols received extensive attention, because they are an important branch of cryptography.

ZKP protocol needs to have three properties. They are:

Completeness: Completeness means that if Verifier and Prover are honest, the protocol should be accepted.

Soundness: Soundness means if Prover is dishonest, cannot convince Verifier except with some small probability.

Zero-Knowledge: Zero-Knowledge means repeating the protocol does not reveal any information about Prover except it has a valid password.

3 A NEW ZKP PROTOCOL USING 3-DIMENSIONAL MATCHING PROBLEM

Goldreich, Micali and Wigderson [4] showed that, under certain complexity assumptions, any NP problem can be used to define a ZKP protocol. Also according to Garey and Johnson [2], 3-dimentional matching problem is a NP problem, therefore we can define a new ZKP protocol using this problem.

3.1 The 3-dimensional matching problem

Three sets X, Y, Z are given so that

- Each set has the same size $|X| = |Y| = |Z| = n$
- The sets are disjoint $X \cap Y \cap Z = \phi$
- $E \subset X \times Y \times Z$

Is there a set of n triples $M \subset E$ so that each element of $X \cup Y \cup Z$ is contained exactly once in one of these triples?

Such a set of triples is a *perfect 3-dimensional matching* in 3-partite hypergraph $G = (V, E)$ where $V = X \cup Y \cup Z$ is vertices set and E is hyperedges set.

The 3-dimensional matching problem belongs to The krap's list of 21 NP problems.

Example:

Let $X = \{1,2,3,4,5,6\}, Y = \{7,8,9,10,11,12\}, Z = \{13,14,15,16,17,18\}$ and hypergraph G is as follows:

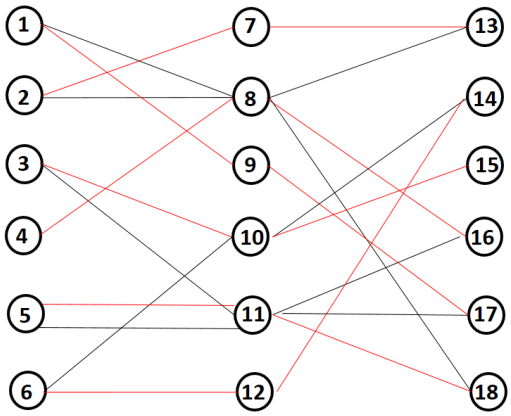

Figure 1: perfect 3-dimensional matching

For this hypergraph, $M = \{(1,9,17), (2,7,13), (3,10,15), (4,8,16), (5,11,18), (6,12,14)\}$ that is shown with red colour, is a perfect 3-dimensional matching.

3.2 ZKP protocol using 3-dimensional Matching Problem

We give a new ZKP protocol as follows:

Given a 3-partite hypergraph $G = (V, E)$.

Prover claims that has a perfect 3-dimensional matching *M* in *G*. Protocol performs as follows:

- Prover chooses a random permutation $\pi = perm\{1,2,...,|V|\}$ and calculates the adjacency matrix of hypergraph $G' = \pi(G)$. (a vertex $i \in V$ get mapped to $\pi(i) \in V'$ and an edge $(i, j) \in E$ gets mapped to an edge $(\pi(i), \pi(j)) \in E'$ and sends the adjacency matrix of hypergraph G' to Verifier as commitment.
- Verifier chooses a random challenge $C \in \{0,1\}$ and sends it to Prover.
- If $C = 0$, Prover sends π, else Prover sends $\pi(M)$ in G' to Verifier.

If $C = 0$, Verifier checks that the information sent by Prover matches with the commitment, i.e., if commitment is adjacency matrix of permuted hypergraph using permutation sent.

If $C = 1$, Verifier checks if the matching sent is actually a perfect 3-dimensional matching in G'. If these checks pass, then Verifier accepts.

Now, we check properties of defined protocol:

• **Completeness**: an honest Prover will always be able to correctly answers an honest Verifier's query in a way that causes the Verifier to accept.

• **Soundness**: The soundness of the protocol is $\frac{1}{2}$. A cheating Prover must either commit to permutation of the original hypergraph, in which case doesn't know a perfect 3-dimensional matching in the permuted hypergraph, or must commit to an unrelated hypergraph where knows a perfect 3-dimensional matching, but cannot provide a permutation mapping the original hypergraph to this unrelated hypergraph.

• **Zero Knowledge**: Repeating the protocol does not reveal any information about Prover except it has a valid password since finding isomorphism between hypergraph and permuted hypergraph is a NP problem, so knowledge of perfect 3-dimensional matching on permuted hypergraph does not reveal password.

Note that by running the protocol k times in parallel, namely, the Prover chooses k random permutations of vertices and commits to them, the Verifier makes k challenges at random, and the Prover answers each of the queries using the corresponding permutation. In this case the soundness error reduces to $(\frac{1}{2})^k$. Note that in this case protocol needs to more interaction between Prover and Verifier.

4 A NEW NON-INTERACTIVE ZKP PROTOCOL USING 3-DIMENSIONAL MATCHING PROBLEM

Non-Interactive ZKP protocols contain only one message sent by Prover to Verifier. These protocols are widely used in construction of cryptographic protocols due to their good security and lower interactive complexity.

We can convert defined protocol in previous section into a non-Interactive protocol using Fiat-Shamir heuristic. For this work, Prover builds a Merkle tree and then computes hash of Merkle root. This value is used as source of entropy that determines values of challenges. The process of

computing Merkle root and then using it to select values of challenges effectively substitutes the need for an interactive protocol.

In Non-Interactive ZKP, Verifier can verify without any interaction. Since output of hash function is unique when calculates with the same input. Therefore, Verifier can calculate Merkle root and understands challenges to verify.

According to above explanation, Non-Interactive ZKP protocol performs as follows:

- Prover chooses k random permutations $\pi_i = perm\{1,2,...,|V|\}$, $1 \le i \le k$ and calculates the adjacency matrix of hypergraph $G_i^{'} = \pi_i(G)$, A_i , $1 \le i \le k$ as commitment and sends $A = (A_1, A_2, ..., A_k)$ to Verifier.
- Verifier chooses a random number r and sends to Prover.
- Prover builds a Merkle tree whose leaves are r times degrees of vertices of G, and then computes hash of Merkle root and sets k challenges $c_i, 1 \le i \le k$ as $C = (c_1, c_2, ..., c_k) =$ the first k bits of hash output and sends $P = (P_1, P_2, ..., P_k)$ where $P_i = \pi_i$ if $c_i = 0$ else $P_i = \pi_i(M)$ .to Verifier.

Verifier using calculation hash of Merkle root, understands values of challenges, therefore

If $c_i = 0$, Verifier checks that the information sent by Prover matches with the commitment i.e. if commitment A_i is adjacency matrix of permuted hypergraph using permutation sent and if $c_i = 1$, Verifier checks if the matching sent is actually a perfect 3-dimensional matching in $G_i^{'}$. If these checks pass, then Verifier accepts.

5 SOME OF THE APPLICATIONS OF ZKP PROTOCOLS

In this section, we want to show a variety of applications of ZKP protocols that are applicable with real life problems.

5.1 Blockchain

The transparency of public blockchains such as Bitcoin and Ethereum enable public verification of transactions. However, it also implies little privacy and can lead to deanonymization of users. ZKP protocols can introduce more privacy to public blockchains. For instance, the cryptocurrency Zcash is based on Zero-Knowledge Succinct Non-Interactive Argument of Knowledge (Zk-SNAKR), a type of Zero-Knowledge cryptographic method.

5.2 Multy Party Computation

ZKP protocols have notably found several applications in the area of secure multiparty computation. In secure multiparty computation, several parties would like to compute something useful with their secret, but would not like to share them.

5.3 Schnorr Signature

The main problem that Schnorr signatures aim to solve is that of signing a message so that receiver can ensure that it was not tampered with. This algorithm is based on ZKP.

5.4 Secure Remote Password

Common schemes usually use a plaintext approach, in which the user communicates a username and a password to a (trusted) server, that then has the burden of securely maintaining said information. The Secure Remote Password protocol [6](SRP) is the most common example of ZKPs used for authentication. SRP is based on Diffie-Hellman that use NP-complete problem "discrete logarithm".

6 CONCLUSION

In conclusion, for both the mathematicians and cryptographers, the evidence of ZKP is of significant theoretical and practical concern. In this paper, a new ZKP based on 3-dimensional matching problem was described and The required properties was checked. Then by using Fiat-Shamir, protocol converted to a Non-Interactive ZKP. Also some of the applications of ZKP in Blockchain, MultiParty Computation, Schnorr Signature and Secure Remote Password have introduced.

7 ACKNOWLEDGEMENTS

The author acknowledges the Department of Mathematics, Iran University of Science and Technology.

REFERENCES

[1] S. Aaronson. Why philosophers should care about computational complexity.CoRR, abs/1108.1791, 2011.

[2] M. R. Garey and D. S. Johnson. Computers and Intractability: A Guide to the Theory of NPCompleteness. A Series of Books in the Mathematical Sciences. San Francisco, Calif.: W. H. Freeman and Co. pp. x+338. ISBN 0-7167-1045-5. MR 0519066, 1979.

[3] O. Goldreich. Zero-knowledge twenty years after its invention. Electronic Colloquium on Computational Complexity (ECCC), (063), 2002.

[4] O. Goldreich, S. Micali, and A. Wigderson. Proofs that yield nothing but their validity, or all Languages in np have zero-knowledge proof systems. Journal of the ACM, 38 (1), 1991, 690-728.

[5] Sh. Goldwasser, S. Micali, and C. Rackoff. The knowledge complexity of interactive proof-systems (extended abstract). In 17th Annual ACM Symposium on Theory of Computing (STOC'85), (1985), 291-304.

[6] T. Wu. The Secure Remote Password Protocol. In Proceedings of the 1998 Internet Society Network and Distributed System Security Symposium, (1997), 97-111.