Examining Schnorr's protocol in the context of zero-knowledge proofs

Manqi Yue

College of Letters and Science, University of California, Santa Barbara, Santa Barbara, 93106, United States

manqiyue@ucsb.edu

Abstract. The rise of technology has brought with it a heightened awareness of the necessity to shield personal data and maintain exclusive access to specific knowledge. A notable solution that emerged from this consciousness is Zero-Knowledge Proofs (ZKPs) and, more specifically, Schnorr's Protocol. Historically, Zero-Knowledge Proofs have a compelling lineage, tracing their roots back to the fervent discussions among cryptographers aiming to achieve a balance between information sharing and privacy. ZKPs are cryptographic methods that allow one party to prove to another that a statement is true, without revealing any specific information about the statement itself. In the midst of these developments, Schnorr's Protocol emerged as a renowned interactive proof system. It possesses an intuitive structure that has made it pivotal in the enhancement of digital security. The typical flow of Schnorr's Protocol begins with the prover sending a commitment to the verifier. The verifier then sends a random challenge back to the prover, who, in turn, produces a response. What's captivating is that the verifier can ascertain the validity of the proof without gaining insight into the underlying secret. Interactive Schnorr's Protocol involves real-time back-and-forth communication between the prover and verifier. On the other hand, the non-interactive version eliminates this need by using a cryptographic hash function, thereby streamlining the process.

Keywords: ZKPs, Schnorr's Protocol, Blockchain, zk-SNARK.

1. Introduction

A cryptographic method called zero-knowledge proof, developed in 1989 by MIT researchers including Shafi Goldwasser, Silvio Micali, and Charles Rackoff, enables the validation of a statement without revealing any additional information beyond its truth. Their seminal paper, "The Knowledge Complexity of Interactive Proof Systems" [1], set the groundwork for a deep understanding of ZKPs. Over time, Zero-Knowledge Proofs and Schnorr's Protocol have seen significant developments and have been employed in diverse areas. In the early 2000s, experts in the field expanded upon the theoretical underpinnings of ZKPs, ushering in the era of non-interactive and succinct zero-knowledge proofs, making them more suitable for real-world applications.

It's widely accepted that all zero-knowledge proofs must exhibit the attributes of Completeness, Soundness, and Zero-Knowledge. Completeness refers to the scenario where an honest prover can convince an honest verifier of the truth of a statement. Soundness means that it's exceptionally improbable for a deceptive prover to convince an honest verifier of the truth of a false statement. In situations where the statement is true, the verifier gains no knowledge, implying they discern nothing beyond the statement's veracity [2]. Presently, ZKPs and Schnorr's Protocol are cornerstone elements in contemporary cryptographic systems, fulfilling roles from bolstering privacy in blockchain dealings to safeguarding confidential information across various platforms.

2. Schnorr's Interactive Protocol

An important application of ZKP is Schnorr's protocol. Schnorr's Protocol, named after its creator Claus-Peter Schnorr, represents one of the pioneering real-world applications of Zero-Knowledge Proofs. Claus-Peter Schnorr introduced his protocol in 1989, in a paper titled Efficient Signature Generation for Smart Cards [3]. This protocol aimed to provide a secure and efficient digital signature scheme, which could be particularly valuable for smart card applications. Schnorr's protocol based on a solid foundation of mathematical principles and security assumptions, making it a well-regarded cryptographic algorithm for digital signatures and other applications. This was a significant leap forward in cryptography as it allowed for secure digital signatures that were both efficient and mathematically robust. Schnorr's protocol offered so many advantages, making it becomes an important component of today's cryptography.

2.1. Fundamental Components within the Framework

To build a such model, all main components and symbols are listed below:

V: People who verify whether people know the knowledge or not.

P: People who want to prove that they know the knowledge to the verifier.

r, c, g: choose any r, c, $g \in \mathbb{Z}p$ that r, c, $g \neq$ 0, 1. g also need to be published to all as the public key in order to fulfill the flow of algorithm.

$\mathbb{Z}p$: A cyclic group that generated by number g, where $\mathbb{Z}p$ = {0, 1, …, $p-1$}, which p = 2q +1(q should be a large prime number) [4].

x: The number or knowledge that P try to prove to V, also known as the private key, encoding by h $=gx$. Admittedly, the verifier does not need to know the prover's private key x.

2.2. Typical Operational Flow of the Algorithm

The general operational flow of the Schnorr's Interactive Protocol follows these three steps:

1st communication: P generates a random number r from the $\mathbb{Z}p$, then send u = g^r to the verifier V.

2nd communication: V picks another random number c that generate from $\mathbb{Z}p$ and sends it back to P.

3rd communication: P calculates the z = r + cx and send it back to V.

After then V must check whether g^z equals to u*h^c. If the results are the same, V should assume that P indeed knows x. If the results are different, V should deny P and refuse the fact P knows x [5]. Regularly, with the fact that all numbers used in whole procedure are generated randomly and it is too harsh to factorize big prime number, people should assume that the chance of cheating in such model is almost impossible at all, making the security of Schnorr's protocol become one of the best encrypting method in modern days.

2.3. Pertinent Use Cases

Consider a scenario that a person V want to know who knows his phone number among a group of his friends. Assuming these friends are all logistic and intelligent. A few friends asserts that they know the answer, so V knows that at least some of friends know his number. V want friends to prove to him about they know his number without letting others to know, while V also want to let the friends that know the number to convince other friends that don't know the number that they know the number without giving away any clue of the number itself.

To solve the problem, V first generate a random number g from $\mathbb{Z}p$ and publishes it to all his friends. For a friend who know the phone number of V, which is x in this case, this friend should generate a random number r from $\mathbb{Z}p$. Then the friend should pass u = g^r to V based on the algorithm. After receiving u, V generate a random number c from the $\mathbb{Z}p$ and sends it back to the friend. Finally, the friend calculates z = cx + r and send z to the verifier V. Now V just needs to check whether g^z equals to u*h^c. If yes, V finds out this friend know his phone number.

2.4. Underlying Rationale and Principles

Schnorr's protocol is built on the mathematical properties of the Discrete Logarithm Problem (DLP). Due to the perceived computational complexity of solving the Discrete Logarithm Problem, particularly in groups characterized by large prime orders—a fundamental security element in numerous cryptographic systems—Schnorr's protocol emerges as a relatively secure option for user trust. Consequently, it has gained extensive utilization in internet authentication. The key idea behinds this is the private key (x) used for signing a message corresponds to the solution of the DLP for a specific instance, which is h = g^x.

3. Schnorr's Non-Interactive Protocol

Schnorr's protocol is also able to be modified to generate non-interactive zero-knowledge proofs (NIZKPs) for a wide array of statements, adding to its versatility and usefulness in cryptographic protocols beyond signatures. Admittedly, the non-interactive Schnorr's protocol is highly similar to the interactive Schnorr's protocol. In short, the non-interactive Schnorr's protocol alternates the direction of info-transmission in 2nd step of communication in the interactive Schnorr's protocol, resulting in only one step of message transmission between P and V [6].

3.1. Contrasting the Interactive Variant

In contrast to the interactive version, Schnorr's Non-Interactive Protocol enables the prover, P, to demonstrate a statement's validity to the verifier, V, without the need for multiple rounds of communication between them. Instead of getting the random number c from V, the prover uses hash function (typically based on Fiat-Shamir transformation) to get a random number c by themselves. As a result, P sends u, c, z to the verifier together in one time of message transmission. Without the process of waiting for the feedback from the verifier in 2nd communication in Schnorr's interactive protocol, the flow of algorithm will be speeded up significantly, resulting in the non-interactive Schnorr's protocol to be much applicable when only a single verifier is facing a lot of provers.

3.2. Relevant Use Cases

The non-interactive Schnorr's Protocol is widely used in various cryptographic applications, such as digital signatures, identity verification, and more, where proving knowledge or authenticity without interactive communication is desirable.

In scenarios where multiple signatures need to be verified simultaneously, non-interactive Schnorr's Protocol allows for efficient batch verification. By implicating such protocol, verifier can check the validity of multiple signatures in a single operation, reducing computational overhead. Besides, Schnorr's non-interactive Protocol can also be used to create privacy-preserving credentials or attestations, or enables users to prove they possess a certain identity attribute without disclosing their actual identity. For example, a user can prove their eligibility for a service (e.g., age verification) without revealing their actual age, thus protecting their privacy. It is undeniable that non-interactive Schnorr's Protocol are pretty useful in enabling secure, privacy-preserving, and efficient cryptographic interactions, which are all crucial characteristics in various applications where privacy and security needed to be focused in [7].

3.3. Core Components within the Model

Generally, a model of non-interactive Schnorr's Protocol includes the statement to prove, public parameters, a commitment to the statement, a challenge, a response, and a verification equation. Noticed that these components are the same for the interactive Schnorr's Protocol:

Statement to Prove (or NP Language): This represents the fact or statement that the prover wants to prove knowledge of without revealing the actual information. In the context of Schnorr's NIZKP, this could be a statement like "I know that the phone number is 13355667."

Public Parameters: These are publicly known values or parameters shared by both the prover and verifier, which are the exact same as in part 2.1.

Commitment to the Statement: The prover generates a commitment value that represents their knowledge of the statement without revealing the actual statement or the knowledge itself.

Challenge (Randomness): The verifier generates a challenge value e, typically a random number or a value derived from the statement and commitment.

Response (Proof): The prover calculates a response value "s" based on the challenge and their private knowledge, such as the private key x.

Verification Equation: The verifier checks whether the following equation holds, which is the part of checking whether g^z equals to u*h. Similar to the interactive Schnorr's Protocol, if this equation holds, it implies that the verifier is assured that the prover indeed possesses the necessary knowledge to fulfill the statement without disclosing the actual knowledge [8].

4. Broader Applications of ZKPs

Zero-Knowledge Proofs have a wide range of applications, especially in the domains of cryptography and blockchain. By using varieties type of algorithm demonstrating the interactive and non-interactive zero knowledge proof, ZKPs offer solutions for privacy, security, and efficiency challenges in various fields for many companies, while it also helps to preserving user privacy.

4.1. Leveraging ZKP Mechanisms in Blockchain

A blockchain is a distributed and decentralized digital ledger system that uses numerous computers to record transactions while preserving the data's security, transparency, and immutability. Beyound cryptocurrencies, blockchains present a diverse array of applications, including but not limited to supply chain oversight, electoral systems, identity authentication, and numerous others. They are often used to create trustless, transparent systems for various industries. In the context of blockchain, ZKPs are used to prove various statements without revealing the underlying data, ensuring privacy, scalability, and security. These mathematical techniques enable complex computations to be performed off-chain while providing verifiable proofs of correctness on-chain, which is crucial for blockchain applications like privacy coins, secure smart contracts, and efficient scaling solutions [9].

Cryptocurrencies such as Bitcoin can disclose more transaction details than users may desire and be benefited from it. With the application of Zero-Knowledge Proofs, it enhances the transactional privacy in coins. By using Quadratic Arithmetic Programs in zk-SNARKs, people can convert information into polynomial forms, allowing for proof of correct program execution. Recursive zk-SNARKs enable proof verification of other proofs, reducing data and computational demands for improved scalability, eliminating the characteristic of Blockchains' decentralized characteristic [10].

4.2. The Role of ZKPs in Authentication Systems

Nowadays, ZKPs eliminates many parts of the need for traditional passwords. By exert the ZKPs, users can prove their identity without revealing their actual password. For example, a user can prove they know a secret without disclosing the secret itself. In systems that use fingerprints or facial recognition, ZKPs can be applied to verify a user's biometric features without storing or transmitting sensitive biometric templates. This enhances privacy and reduces the risk of biometric data breaches.

ZKPs also allow users to prove specific attributes or properties about themselves without revealing unnecessary information. This is useful for services that require attribute-based access control. For instance, if a hospital wants to collect the blood types of all citizens, by exerting ZKPs, the hospital will easily establish a way for people to send their blood type information to the database without asking additional knowledge about the privacy of each person. In this case, ZKPs allow individuals to prove their identity, credentials, or attributes while minimizing the disclosure of sensitive information. It not only enhances security but also helps protect user privacy in an increasingly digital and interconnected world [11].

4.3. zk-SNARK

Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge, often shortened as zk-SNARKs, represent a cryptographic method enabling one party to demonstrate possession of particular information to another party. zk-SNARKs, which leverage elliptic curves and other mathematical techniques, enable the creation of compact, non-interactive zero-knowledge proofs. These proofs have been instrumental in enhancing privacy, scalability, and security in blockchain technology, leading to the emergence of privacy coins like Zcash and efficient scaling solutions like zk-Rollups. Just like other type of ZKPs, as it offers a highly ingenious method for affirming the accuracy of a statement, it exposes nothing more about information that is more than the need for interaction between the parties. To illustrate, when someone provides a signature for a specific public key in relation to specified data, zk-SNARK serves as evidence confirming that they indeed hold the corresponding private key for that public key.

In earlier zero-knowledge protocols, the prover and verifier engaged in multiple rounds of communication. On the contrary, in non-interactive setups, the entire proof resides within a single message sent from the prover to the verifier. For instance, by exerting zk-SNARK technology, law enforcement can demonstrate to the public that the DNA profile of a Presidential Candidate is not present in their forensic DNA database. Importantly, this proof is created by law enforcement itself and doesn’t rely on any external trusted entity. Furthermore, it does not give any additional information about the database’s contents or the candidate’s profile. Notably, no DNA data is exposed to any party beyond law enforcement. The proof itself is compact, taking up less space than the DNA database, and can be verified more swiftly than the time it takes to manually inspect the database [12].

5. Conclusion

Interactive and non-interactive versions of Schnorr’s protocol are pivotal in the sphere of zero-knowledge proofs. With the interactive variant, the prover and verifier participate in multiple communication rounds. In this process, the prover incrementally persuades the verifier of their understanding regarding a statement without disclosing the specific knowledge. This interactive method boasts rigorous security and soundness attributes but demands continuous engagement.

Conversely, the non-interactive form condenses the entire demonstration into a single message from the prover to the verifier, showcasing its efficiency and adaptability for diverse applications, including the realm of blockchain technology. In both methodologies, Schnorr’s protocol capitalizes on mathematical principles, such as discrete logarithms, to guarantee the prover’s compelling display of knowledge while upholding the secrecy of the intrinsic information.

Unquestionably, Schnorr’s protocol carries profound implications for privacy, authentication, and secure transactions across various sectors. Given that ZKPs empower users to substantiate their identities and attributes without exposing confidential details, it stands as a promising candidate to fortify digital interactions. As technology progresses, such potent tools can elevate the security and privacy of authentication frameworks, diminishing the hazards of data breaches and privacy infringements.

References

[1] Goldwasser, S., Micali, S., & Rackoff, C. (1989). The Knowledge Complexity of Interactive Proof Systems. SIAM Journal on Computing, 18(1), 186–208. https://doi.org/10. 1137/ 0218012.

[2] Hasan, J. (2019). Overview and Applications of Zero Knowledge Proof (ZKP). IJCSN - International Journal of Computer Science and Network, 8(5).

[3] Schnorr, C. P. (1991). Efficient signature generation by smart cards. Journal of Cryptology, 4(3). https://doi.org/10.1007/bf00196725.

[4] Morais, E., Koens, T., van Wijk, C., & Koren, A. (2019). A survey on zero knowledge range proofs and applications. SN Applied Sciences, 1(8). https://doi.org/10.1007/s42452-019-0989-z.

[5] Wu, H., & Wang, F. (2014). A Survey of Noninteractive Zero Knowledge Proof System and Its Applications. The Scientific World Journal, 2014, 1–7. https://doi.org/10.1155/2014/560484.

[6] Al-Adhami, A. H., Ambroze, M., Stengel, I., & Tomlinson, M. (2019, March 1). An Effencient Improvement of RFID Authentication Protocol Using Hash Function ZKP. IEEE Xplore. https://doi.org/10.1109/SCCS.2019.8852614.

[7] Gong, Y., Jin, Y., Li, Y., Liu, Z., & Zhu, Z. (2022, January 1). Analysis and comparison of the main zero-knowledge proof scheme. IEEE Xplore. https://doi.org/10.1109/ BDICN55575. 2022.00074.

[8] Partala, J., Nguyen, T. H., & Pirttikangas, S. (2020). Non-Interactive Zero-Knowledge for Blockchain: A Survey. IEEE Access, 8, 227945–227961. https://doi.org/10.1109/ access. 2020.3046025.

[9] Bandara, E., Liang, X., Foytik, P., Shetty, S., & Zoysa, K. D. (2021). A Blockchain and Self-Sovereign Identity Empowered Digital Identity Platform. 2021 International Conference on Computer Communications and Networks (ICCCN). https://doi.org/10.1109/ icccn52240. 2021.9522184.

[10] Ahmad, Md. O., Tripathi, G., Siddiqui, F., Alam, M. A., Ahad, M. A., Akhtar, M. M., & Casalino, G. (2023). BAuth-ZKP—A Blockchain-Based Multi-Factor Authentication Mechanism for Securing Smart Cities. Sensors, 23(5), 2757. https://doi.org/10.3390/s23052757.

[11] Ni, N., & Zhu, Y. (2022). Enabling Zero Knowledge Proof by Accelerating zk-SNARK Kernels on GPU. Journal of Parallel and Distributed Computing. https://doi.org/10.1016/ j.jpdc. 2022.10.009.

[12] Ben-Sasson, E., Bentov, I., Horesh, Y., & Riabzev, M. (2018). Scalable, transparent, and post-quantum secure computational integrity. https://eprint.iacr.org/2018/046.pdf.