zkSHIELD

Project report submitted for

Vth Semester Minor Project-V

in

Department of Computer Science & Engineering

By

Amrit Gupta (211000008)

Anirban Bhattacharjee (211000010)

Vinay Kiran Polisetti (211000037)

Department of Computer Science & Engineering

Dr. Shyama Prasad Mukherjee

International Institute of Information Technology, Naya Raipur

(A Joint Initiative of Govt. of Chhattisgarh and NTPC)

Email: iiitnr@iiitnr.ac.in, Tel: (0771) 2474040, Web: www.iiitnr.ac.in

CERTIFICATE

This is to certify that the project titled “zkSHIELD” by “AMRIT GUPTA, ANIRBAN BHATTACHARJEE, VINAY KIRAN POLISETTI” has been carried outunder my/our supervision and that this work has not been submitted elsewhere for a degree.

(Signature of Guide)

Dr. Satyanarayana Vollala

Assistant Professor

Department of CSE

Dr. SPM IIIT-NR

October, 2023

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Amrit Gupta	**Anirban Bhattacharjee**	**Vinay Kiran Polisetti**
(211000008)	**(211000010)**	**(211000037)**

Date : ____________________

Approval Sheet

This project report entitled “zkSHIELD” by “AMRIT GUPTA, ANIRBAN BHATTACHARJEE,
VINAY KIRAN POLISETTI” is approved for V Semester Project.

(Signature of Examiner - I)

____

Name of Examiner -I

(Signature of Examiner - II)

____

Name of Examiner -II

(Signature of Chair)

____

Name of Chair

Date:______________Place: ______________

zkSHIELD – A Web-Based Cross-Chain Zero Knowledge Asset Proof

***Abstract* — The global transition towards Web3 is propelling the development and proliferation of numerous blockchain networks. These diverse blockchains offer distinct advantages and serve various purposes, such as Ethereum for decentralized applications, Bitcoin for investment, and Polygon for scalability solutions. Each blockchain features its own native cryptocurrency, encompassing a wide array of tokens. In response to this fragmentation, several prominent blockchain platforms have introduced interoperability features, enabling tokens from different chains to coexist through the creation of wrapped tokens. As a result, users' assets are distributed across multiple tokens on various blockchain networks, further emphasizing the need for seamless cross-chain asset management. We have engineered a web application designed to facilitate the visualization of user assets across various blockchain networks and wallets. Notably, this platform boasts the capability to integrate secondary wallets seamlessly into users' accounts. Furthermore, the web application offers comprehensive support for tracking the historical trajectory of incoming and outgoing asset proof requests associated with individual users. The meta data of a request is securely stored on a blockchain including its corresponding proofs' Content Identifier (CID) after zkSNARKs proof generation and upon hosting it on the InterPlanetary File System (IPFS).**

***Impact Statement* - ZK Shield promises to redefine the cryptocurrency landscape by introducing a groundbreaking solution that allows individuals to verify their cryptocurrency holdings against specific financial thresholds without compromising their privacy. Utilizing cutting-edge Zero Knowledge Proofs (ZKPs), it empowers users to selectively disclose only essential information, preserving the confidentiality of their crypto assets. This innovation not only safeguards sensitive financial data but also opens doors to broader financial inclusion, paves the way for advancements in cryptographic techniques, and empowers users with control over their financial privacy, fundamentally reshaping the intersection of privacy and transparency in the digital era.**

***Index Terms* — zk-shield, zero knowledge proofs, asset proof, solvency proof, zk-SNARKs, non-interactive zero knowledge**

I. INTRODUCTION

PROBLEM STATEMENT

The problem at hand is to develop a robust protocol and platform that addresses this conundrum. It must enable users to demonstrate their cryptocurrency assets' compliance with predetermined financial criteria while safeguarding the privacy of sensitive information. This solution should empower individuals to selectively disclose only the relevant information for verification, thereby ensuring that unrelated details remain confidential. In the rapidly evolving landscape of cryptocurrencies, the challenge of reconciling financial transparency with individual privacy has become increasingly apparent. As blockchain technology inherently records all transactions transparently, individuals and institutions often require verification of cryptocurrency holdings against specified financial thresholds. However, individuals may be hesitant to disclose detailed information about their cryptocurrency assets, including precise balances, wallet addresses, and past and future transaction history, due to privacy concerns.

MOTIVATION

The global landscape is witnessing a paradigm shift towards Web3, driving the development and deployment of multiple blockchains. Each of these blockchains presents unique advantages and caters to diverse use cases, exemplified by Ethereum's role in facilitating decentralized applications (dApps), Bitcoin's prominence in investment, and Polygon's emphasis on scalability. Notably, each blockchain is accompanied by its native currency in the form of distinct tokens. Recognizing the diverse strengths of various blockchains, major players in the space have incorporated interoperability features, enabling the existence of tokens from other chains in the form of wrapped tokens. However, this interoperability, coupled with the proliferation of blockchains, has led to a dispersion of user assets across different tokens on different chains.

Extending the discussion, this scattering of assets poses challenges for users in terms of managing and tracking their holdings efficiently. The fragmented nature of assets across multiple blockchains raises concerns related to user experience, security, and overall accessibility. As the Web3 ecosystem continues to evolve, addressing these challenges becomes crucial for fostering a seamless and user-friendly decentralized environment. Future research and development efforts should focus on innovative solutions to streamline asset management, enhance cross-chain interoperability, and mitigate the complexities associated with the dispersed nature of user assets in the Web3 era.

CONTRIBUTIONS

This paper makes significant contributions to the field of asset verification by employing zk-SNARKs to ensure privacy and security. It introduces a specialized circuit capable of processing users' net worth information, comparing it against predefined thresholds, and ensuring the integrity and accuracy of asset verification within the zkSHIELD system. The use of PowerofTau28 MPC for randomness generation during zk-SNARK setup and secure storage of .wasm and .zkey files on the server adds an extra layer of protection. The deployment of the verifier.sol smart contract on the Polygon Mumbai testnet, along with the Groth16 function from the snarkjs library for proof generation, contributes to the robustness of the verification process. It addresses scalability challenges by utilizing IPFS for storing proofs, ensuring efficient and secure handling of zero-knowledge proof generation. The proposed smart contract interacting with Chainlink and token contracts simplifies the collection of token quantities and exchange rates, streamlining the net worth calculation process. Overall, the paper presents a comprehensive and innovative approach to asset verification, combining cryptographic techniques, secure storage practices, user authentication, and efficient proof generation mechanisms.

II. LITERATURE REVIEW

Existing solutions lacks scalability when it comes to managing tokens across multiple blockchain networks and facilitating secondary wallet approvals. This limitation hinders its applicability in diverse and evolving blockchain ecosystems. The existing proposals fail to offer a method for onboarding different organizations into the system without necessitating significant modifications to the hosted code. This lack of adaptability may hinder broader adoption and integration. While the available solutions are effective in implementing asset proof within a private environment, it falls short in ensuring privacy for transactions in public use cases. This deficiency raises concerns regarding the protection of sensitive data in open and shared blockchain networks. The current approaches highlight zkSNARKs as the most suitable solution for asset-proof implementation. However, it acknowledges the considerable challenges associated with scaling the implementation to handle a large number of users and transactions effectively. The in-place strategies offer a robust framework for managing decentralized identities but overlooks the crucial aspect of ensuring the privacy of assets stored within user accounts. This oversight may pose privacy risks and limit its utility in scenarios where asset confidentiality where privacy is of paramount importance.

R.no	*Paper*	*Advantages*	*Disadvantages*
1	***Huang, J., Huang, T., Wei, H., Zhang, J., Yan, H., Wong, D. S., & Hu, H. [1] (2022)***	***The paper provides a good proposal for building an asset-proof management system for a single organization.***	***The paper's proposed architecture isn't scalable for multiple-chain token management and secondary wallet approvals.***
2	***Tabacaru, Robert, et al. [2] (2023)***	***The paper suggests methods to implement asset proof in a private environment.***	***The paper's described framework fails to provide any privacy for the transactions in public use cases.***
3	***Konkin, A., Zapechnikov, S. [3] (2023)***	***The paper tests and recommends the use of zkSNARKs for asset-proof.***	***The paper mentions the challenges of the implementation at a large-use scale.***
4	***Dieye, M., Valiorgue, P., Gelas, J.-P., Diallo, E., Ghodous, P., Biennier, F., & Peyrol, É. [4] (2023)***	***The paper gives a good framework for handling decentralized identities.***	***The paper doesn't mention about the privacy of the assets stored in the account.***

Table 1 shows a comparative analysis of research papers based on asset management and related works. This table serves as a critical tool for identifying research gaps in the field of asset management and privacy, providing a comprehensive comparison of four key papers. The insights derived from the advantages and disadvantages of each paper guided the development of the project that addresses these gaps.

III. PROPOSED FRAMEWORK

The project employs Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge (zk-SNARKs) to ensure privacy and security in asset verification. To achieve this, we have developed a specialized circuit capable of processing users' net worth information, comparing it against a predefined threshold determined by the verifier. The generation of randomness during the zk-SNARK setup process is facilitated through the use of PowerofTau28 MPC. It's noteworthy that the .wasm and .zkey files, integral to the system's operation, are securely stored within the confines of the server for added protection. Furthermore, for the verification process, we deploy the verifier.sol smart contract, which is created from the .zkey, onto the Polygon Mumbai testnet. When a request for proof generation is made, we employ the Groth16 function from the snarkjs library to produce the required proof[6]. Subsequently, the verifier.sol smart contract evaluates the validity of the proof, thus determining whether it satisfies the specified conditions, thereby ensuring the integrity and accuracy of asset verification within the zkSHIELD system.

Within our project framework shown in Fig.1, we have implemented a user registration process aimed at ensuring user authenticity and security. New users are required to go dedicated registration page, where they are prompted to select a unique username. Additionally, they are required to sign a message originating from the wallet they intend to designate as their "primary wallet address." This message signing step serves the dual purpose of verifying the user's wallet ownership. Once the user successfully provides all necessary registration details, our system employs the RSA algorithm to generate a pair of keys. The private key is securely packaged in a downloadable .pem file, which is provided to the user. Simultaneously, the public key is stored on the Polygon Mumbai testnet. This public key plays a pivotal role in encrypting and decrypting data during the handling of incoming and outgoing requests for asset proof within the system. To access their account subsequently, users are required to enter their unique username, sign a message as an additional security measure, and provide the private key, thus ensuring a robust authentication process[5].

Upon accessing the system, users will be directed to a dedicated dashboard page meticulously designed to provide a comprehensive visualization of their assets. This visualization encompasses assets held across an array of wallets, blockchain networks, and wrapped tokens.

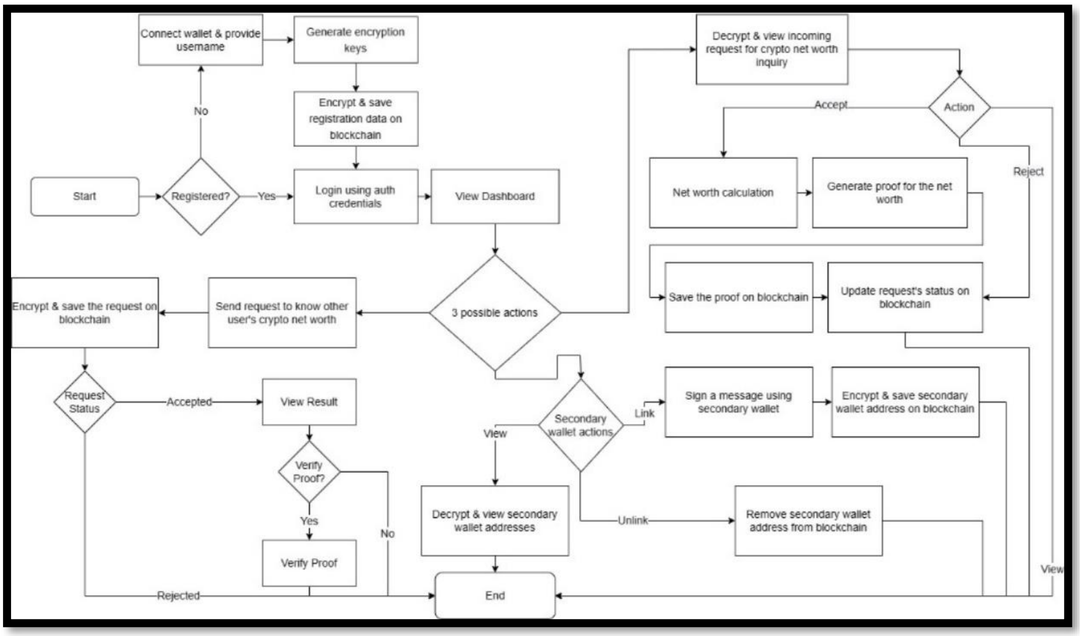
Connect wallet & provide username
Generate encryption keys
Encrypt & save registration data on blockchain
No
Start
Registered?
Yes
Login using auth credentials
View Dashboard
Decrypt & view incoming request for crypto net worth inquiry
Action
Accept
Reject
Net worth calculation
Generate proof for the net worth
Save the proof on blockchain
Update request's status on blockchain
Encrypt & save the request on blockchain
Send request to know other user's crypto net worth
3 possible actions
Request Status
Accepted
View Result
Verify Proof?
Yes
No
Verify Proof
Rejected
End
Secondary wallet actions
Link
View
Unlink
Sign a message using secondary wallet
Encrypt & save secondary wallet address on blockchain
Decrypt & view secondary wallet addresses
Remove secondary wallet address from blockchain
View

Fig.1 *Illustration of the proposed project architecture*

We have undertaken the crucial task of calculating net worth, a fundamental element of our system's functionality[6]. To illustrate this process as shown in Fig.2, let's consider a user, "Bob", who holds a wallet encompassing various assets across different blockchain networks such as Ethereum and Polygon. Given the expanding interconnectivity of blockchain networks, we encounter a challenge when attempting to aggregate these assets, as their units vary significantly. For instance, Ethereum supports its native currency, Ethereum, as well as Bitcoin (in the form of wrapped Bitcoin), Matic (in the form of wrapped Matic), and USDC (in the form of wrapped USDC)[7]. To ascertain Bob's net worth, it is essential to sum the quantities of all the tokens held in his wallet. However, performing a direct addition of tokens with differing units is unfeasible. This is where the Chainlink network comes into play. Chainlink operates as a collection of oracles that deliver reliable real-world data to blockchain networks, offering exchange rates for various tokens in terms of US dollars at any given time[8].

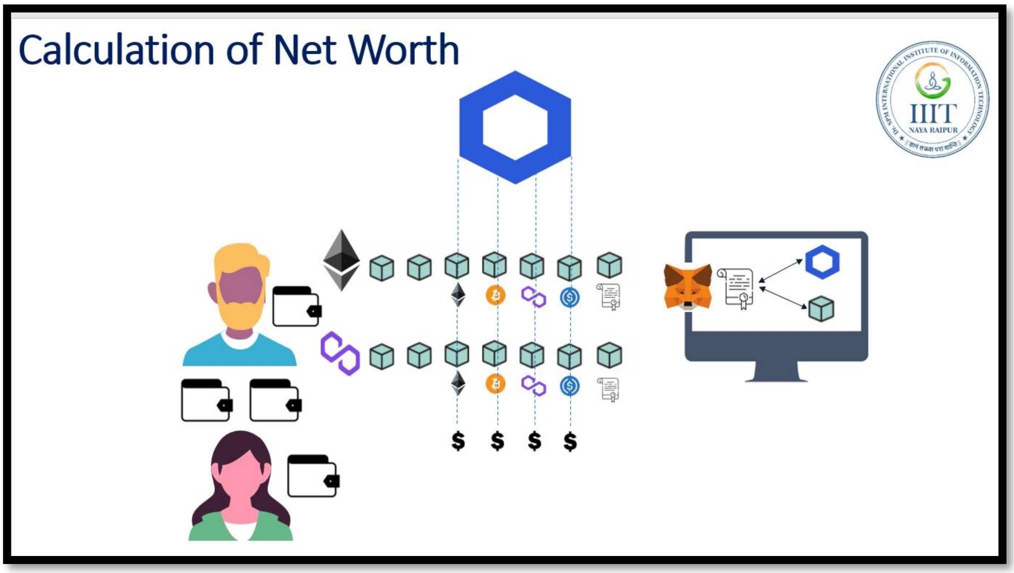
Calculation of Net Worth

Fig.2 Illustration of the calculation of net worth

With this data at our disposal, the task becomes more straightforward. We can collect the token quantities within Bob's wallet, multiply them by the corresponding exchange rates, and aggregate the results. This approach remains consistent whether we are assessing the net worth of another user, such as "Alice," with a single wallet or even when Bob possesses multiple wallets. To facilitate the acquisition of token quantities and exchange rates, our project involves the deployment of a smart contract that interacts seamlessly with both Chainlink and token contracts. This smart contract acts as an intermediary, orchestrating data retrieval when requests are made and subsequently deploying it on the blockchain, thus serving as a valuable resource for our application's users.

In our project, we delve into the intricacies of zero-knowledge proof generation as shown in Fig.3, an integral component of our system. This process involves two key entities: the verifier and the prover. The verifier, who could represent an institution or any party wishing to verify an individual's asset worth, sets specific asset thresholds, as is common in educational institutions when assessing prospective students. Leveraging the capabilities of the Polygon Chain, we undertake the task of zero-knowledge proof generation. The process unfolds with a series of meticulously orchestrated steps, encompassing blockchain state reads and decryption. This journey commences with the verifier acquiring crucial information, including the Verifier's address, the Prover's address, and the predetermined asset threshold. Subsequently, a proof request's metadata is created and stored on the blockchain through a transaction that includes indicators for the prover's response status (initially set to "0," indicating non-acceptance of the request) and a declaration of "No proof" attached to this record.

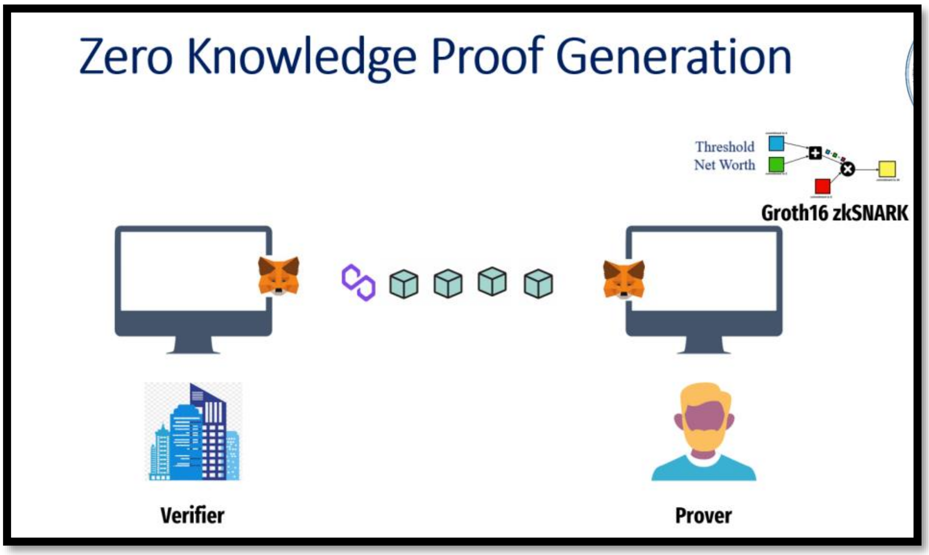

Fig.3 Pictorial Representation of the Proof Generation I

If the prover accepts the request, the process advances, with the prover retrieving the record from the blockchain. Here, the threshold is extracted, and the prover calculates their net worth, which serves as a vital witness in the Groth16 zkSNARK circuit. This circuit, in turn, facilitates the production of a proof in a text format. Recognizing that blockchains often grapple with scalability and throughput limitations, we opt to store the proof on IPFS (InterPlanetary File System), a decentralized protocol for file storage. This move results in the generation of a unique Content Identifier (CID)[9], which is returned to the user. At this juncture, the prover seeks to amend the record's status to "1" (indicating acceptance or rejection of the request) and attaches the CID to the proof (or a "-" symbol to signify rejection), ultimately updating the record on the network. For the verifier's benefit, access to the updated record becomes possible through the blockchain network. The verifier retrieves the CID, triggering a request to IPFS for the associated proof.txt file, ultimately returned to the user. The final phase involves interaction with a verifier circuit[10], deployed as a smart contract on the blockchain by the zkSNARK. This circuit, when presented with the proof.txt file, conclusively determines the validity or invalidity of the proof[11], enabling a meticulous and secure assessment of whether the prover's assets meet the predefined threshold as shown in Fig.4.

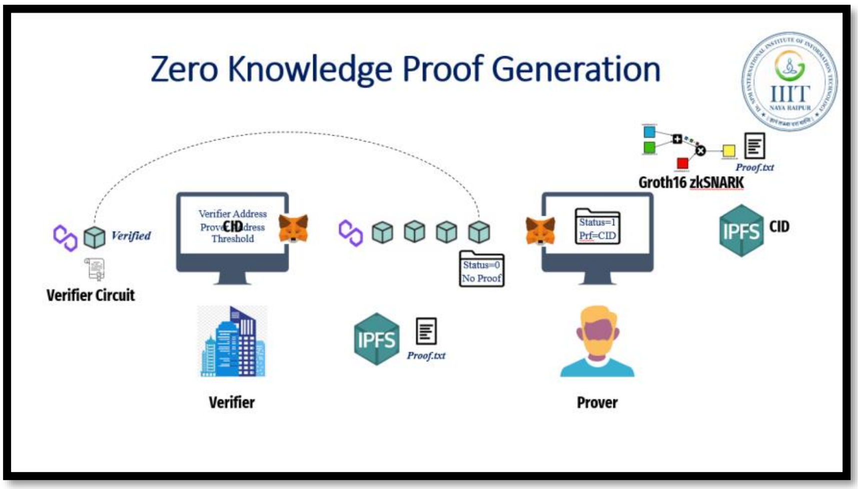

Fig.4 Pictorial Representation of the Proof Generation II

IV. EXPERIMENTAL RESULT AND DISCUSSION

We have achieved the successful development of a decentralized web-based system that facilitates various functionalities apart from ZKPs including the following:

1. The project successfully implemented Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge (zk-SNARKs) to ensure robust privacy and security in asset verification.
2. A specialized circuit was developed to process users' net worth information, comparing it against predetermined asset thresholds, thereby enabling precise and secure asset verification.
3. Randomness generation in the zk-SNARK setup process was achieved through the effective use of PowerofTau28 Multi-Party Computation (MPC).
4. Sensitive .wasm and .zkey files integral to the system's operation were securely stored within the server for enhanced protection.
5. A verifier.sol smart contract was deployed on the Polygon Mumbai testnet, enabling efficient verification of asset proofs.
6. The user registration process was successfully implemented, enhancing user authenticity and security.
7. The registration process involved the generation of RSA key pairs, with private keys provided in a downloadable .pem file, ensuring secure user authentication.
8. The public key was stored on the Polygon Mumbai testnet, facilitating secure data encryption and decryption during asset proof requests.
9. A user-friendly dashboard was created, offering a comprehensive visualization of user assets, including assets across multiple wallets, blockchain networks, and wrapped tokens.
10. The project achieved accurate net worth calculations by leveraging the Chainlink network for real-time exchange rate data, thus enabling the aggregation of diverse assets with varying units.

V. CONCLUSION AND FUTURE DIRECTIONS

In addressing the pressing challenge of reconciling financial transparency with individual privacy in the realm of cryptocurrencies, zkSHIELD has laid a solid foundation. The protocol and platform have successfully demonstrated the potential to empower users while safeguarding their sensitive financial information. By utilizing zk-SNARKS, zkSHIELD has set a precedent for the industry, allowing individuals to verify their cryptocurrency holdings against specific financial thresholds without compromising their privacy. The privacy-preserving approach adopted by ZK Shield, combined with its robust security mechanisms and seamless integration with multiple blockchains, positions it as a pioneering solution in the cryptocurrency space. The project has successfully addressed key challenges related to privacy preservation, selective disclosure, data integrity, user control, and scalability, making it a promising tool for individuals and institutions alike.

REFERENCES

[1] J. Groth, "On the size of pairing-based proofs," in Advances in Cryptology—EUROCRYPT 2016: 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Proceedings, Part I, T. Iwata and K. Dodis, Eds., Cham: Springer International Publishing, 2016, pp. 186–216.

[2] Huang, J., Huang, T., Wei, H., Zhang, J., Yan, H., Wong, D. S., & Hu, H. ZkChain: A privacy-preserving model based on zk-SNARKs and hash chain for efficient transfer of assets. Transactions on Emerging Telecommunications Technologies, e4709. https://doi.org/10.1002/ett.4709.

[3] Tabacaru, Robert, et al. "The Challenges of Proving Solvency While Preserving Privacy." Cryptology ePrint Archive, Paper 2023/079 (2023). https://eprint.iacr.org/2023/079.

[4] Konkin, A., Zapechnikov, S. Zero knowledge proof and ZK-SNARK for private blockchains. J Comput Virol Hack Tech 19, 443–449 (2023). https://doi.org/10.1007/s11416-023-00466-1.

[5] Dieye, M., Valiorgue, P., Gelas, J.-P., Diallo, E., Ghodous, P., Biennier, F., & Peyrol, É. (2023). A Self-Sovereign Identity Based on Zero-Knowledge Proof and Blockchain. IEEE Access, 11, 49445-49455. DOI: 10.1109/ACCESS.2023.3268768.

[6] B. Bünz, B. Fisch, and N. Smart, "Composability proofs for succinct non-interactive arguments of knowledge," in Advances in Cryptology—CRYPTO 2018: 38th Annual International Cryptology Conference, Proceedings, Part II, H. Shacham and A. Boldyreva, Eds., Cham: Springer International Publishing, 2018, pp. 158–189.

[7] V. Bulatov, J. Groth, and R. Ostrovsky, "Non-interactive zero-knowledge proofs for boolean circuits," in Proceedings of the 42nd Annual ACM Symposium on Theory of Computing, STOC 2010, Cambridge, Massachusetts, USA, June 5–8, 2010, R. Ostrovsky, Ed., ACM, 2010, pp. 115–124.

[8] Z. Bootle, J. Groth, and A. O. Narula, "Proofs of knowledge of homomorphic commitments and related applications," in Advances in Cryptology—CRYPTO 2016: 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Proceedings, Part II, M. Robshaw and J. Katz, Eds., Cham: Springer International Publishing, 2016, pp. 158–188.

[9] B. Bünz, B. Fisch, and N. Smart, "Bulletproofs: Short and efficient zero-knowledge proofs for range proofs," in Advances in Cryptology—CRYPTO 2018: 38th Annual International Cryptology Conference, Proceedings, Part II, H. Shacham and A. Boldyreva, Eds., Cham: Springer International Publishing, 2018, pp. 315–344.

[10] A. Tomescu and B. Bünz, "Zero-knowledge proofs with hidden provers," in Advances in Cryptology—CRYPTO 2020: 40th Annual International Cryptology Conference, Proceedings, Part I, D. Lieman and M. Mannan, Eds., Cham: Springer International Publishing, 2020, pp. 704–732.

[11] M. Beller, A. Chiesa, I. Eyal, and R. A. Popa, "Proof-based state machines: A cryptographic perspective," in Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 3, 2017, B. Coan and D. S. Wallach, Eds., ACM, 2017, pp. 1255–1270.