Enhancing Metaverse Security through ZK-SNARK Protocol

By

Nabeeha Zahid

(Registration No: 00000401600)

A thesis submitted to the National University of Science and Technology, Islamabad,

in partial fulfillment of the requirements for the degree of

Masters of Science in

Information Security (MSIS)

Supervisor: Dr. Shahzaib Tahir

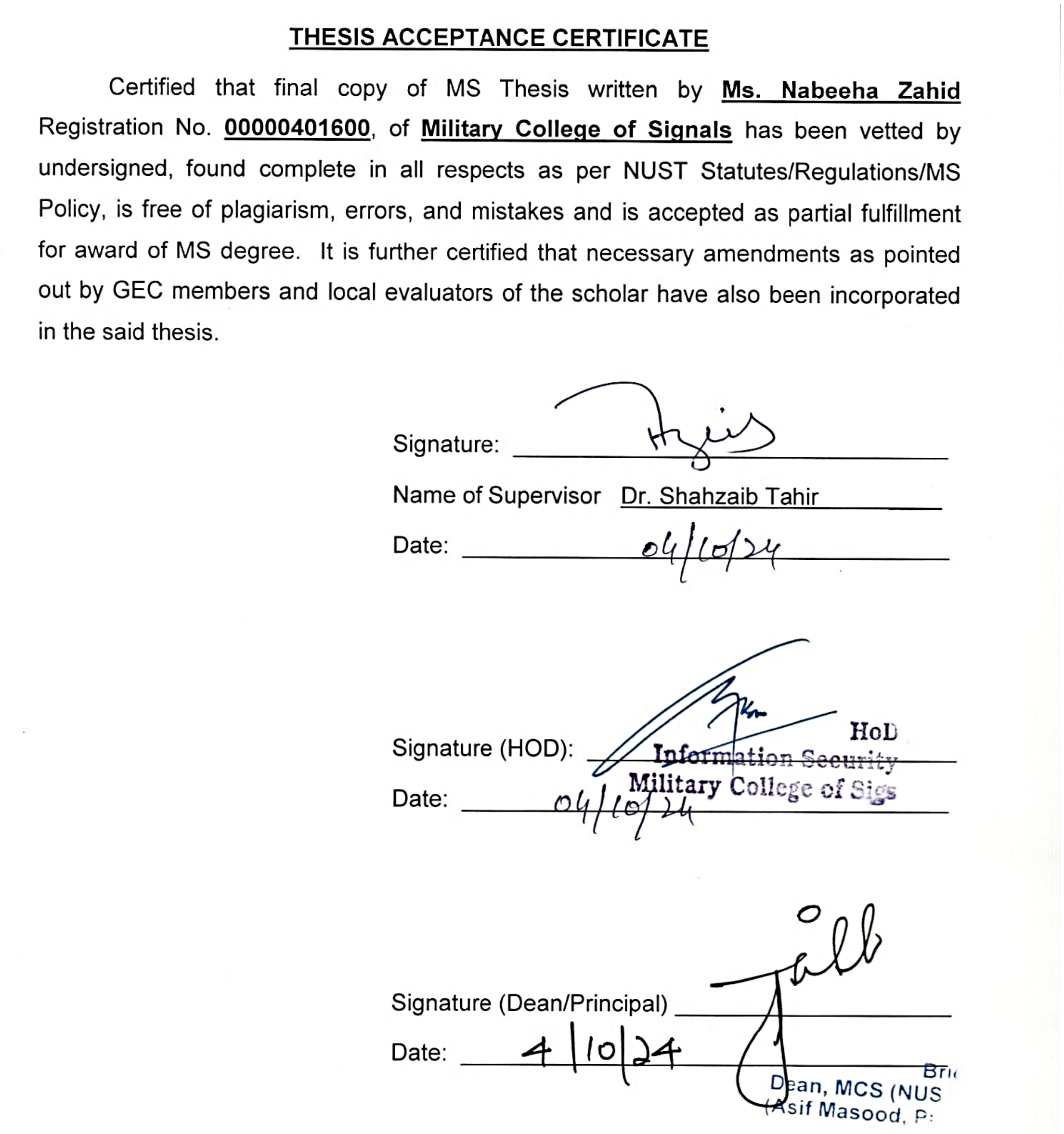
Military College of Signals (MCS)

National University of Science and Technology (NUST)

Islamabad, Pakistan.

(July, 2024)

THESIS ACCEPTANCE CERTIFICATE



THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS Thesis written by **Ms. Nabeeha Zahid** Registration No. **00000401600**, of **Military College of Signals** has been vetted by undersigned, found complete in all respects as per NUST Statutes/Regulations/MS Policy, is free of plagiarism, errors, and mistakes and is accepted as partial fulfillment for award of MS degree. It is further certified that necessary amendments as pointed out by GEC members and local evaluators of the scholar have also been incorporated in the said thesis.

Signature: ______________________

Name of Supervisor Dr. Shahzaib Tahir

Date: 04/10/24

Signature (HOD): ______________________

HoD
Information Security
Military College of Sigs

Date: 04/10/24

Signature (Dean/Principal) ______________________

Date: 4/10/24

Bri
Dean, MCS (NUS
(Asif Masood, P

Form TH-4

NATIONAL UNIVERSITY OF SCIENCES & TECHNOLOGY

MASTER THESIS WORK

We hereby recommend that the dissertation prepared under our supervision by **Nabeeha Zahid MSIS-21 Course** Regn No **00000401600** Titled: **"Enhancing Metaverse Security Through ZK-SNARK Protocol Brief Description"** be accepted in partial fulfillment of the requirements for the award of **MS Information Security** degree.

Examination Committee Members

1. Name :**Maj Bilal Ahmed** Signature: for

2. Name: **Maj Sarmad Idrees** Signature: for

3. Name: **Lec Anum Hassan** Signature:

Supervisor's Name: **Asst Prof Dr Shahzaib Tahir** Signature:

Date: 04/10/24

HoD
Information Security
Military College of Sigs
Head of Department

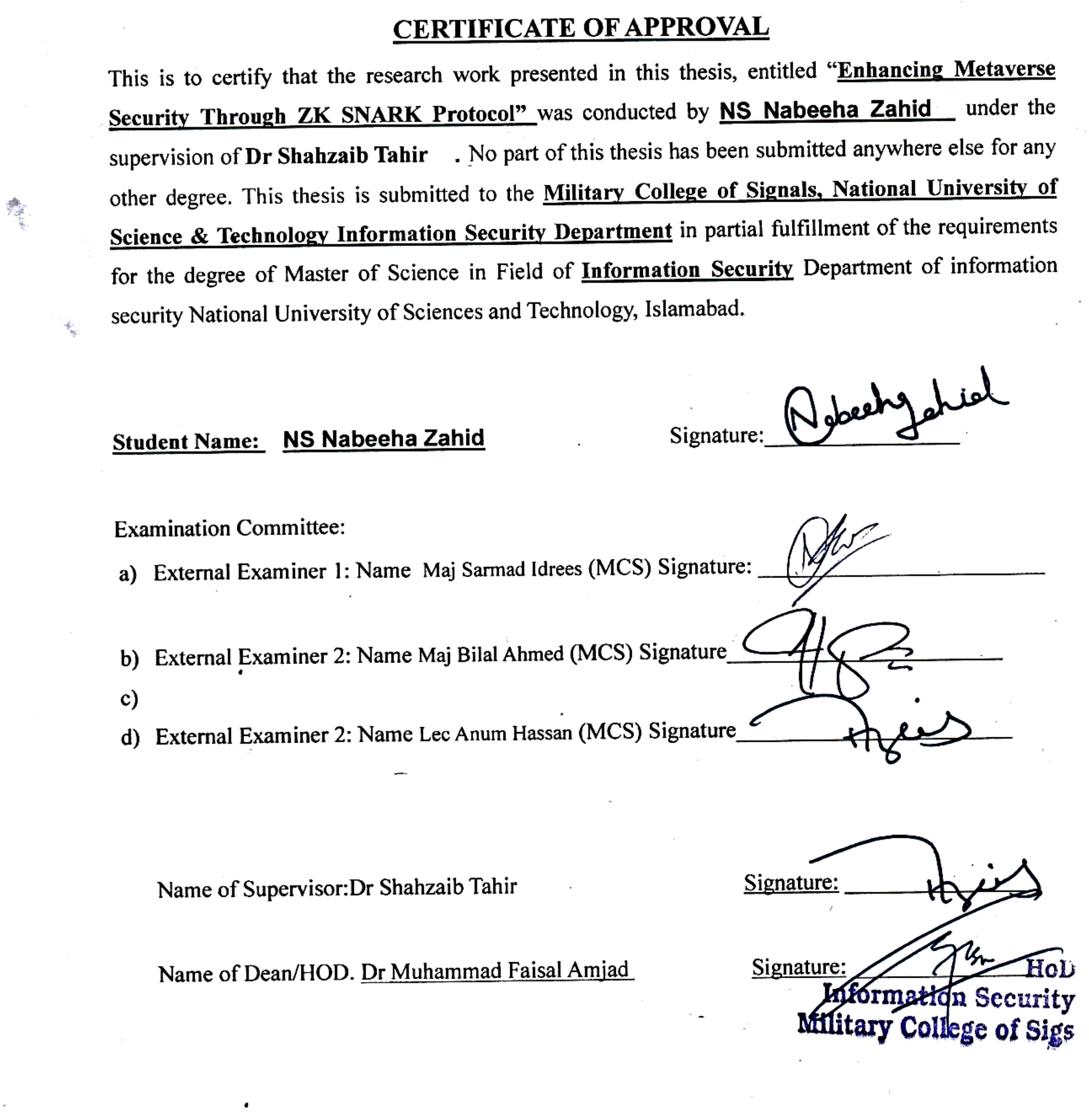
04/10/24
Date

COUNTERSIGNED

Date: 4|10|24

Brig
Dean, MCS (
Asif Masood
Dean

CERTIFICATE OF APPROVAL



CERTIFICATE OF APPROVAL

This is to certify that the research work presented in this thesis, entitled "**Enhancing Metaverse Security Through ZK SNARK Protocol"** was conducted by **NS Nabeeha Zahid** under the supervision of **Dr Shahzaib Tahir** . No part of this thesis has been submitted anywhere else for any other degree. This thesis is submitted to the **Military College of Signals, National University of Science & Technology Information Security Department** in partial fulfillment of the requirements for the degree of Master of Science in Field of **Information Security** Department of information security National University of Sciences and Technology, Islamabad.

Student Name: **NS Nabeeha Zahid** Signature: Nabeeha Zahid

Examination Committee:

a) External Examiner 1: Name Maj Sarmad Idrees (MCS) Signature: ____________

b) External Examiner 2: Name Maj Bilal Ahmed (MCS) Signature____________

c)

d) External Examiner 2: Name Lec Anum Hassan (MCS) Signature____________

Name of Supervisor:Dr Shahzaib Tahir Signature: ____________

Name of Dean/HOD. Dr Muhammad Faisal Amjad Signature: ____________ HoD Information Security Military College of Sigs

AUTHOR'S DECLARATION

AUTHOR'S DECLARATION

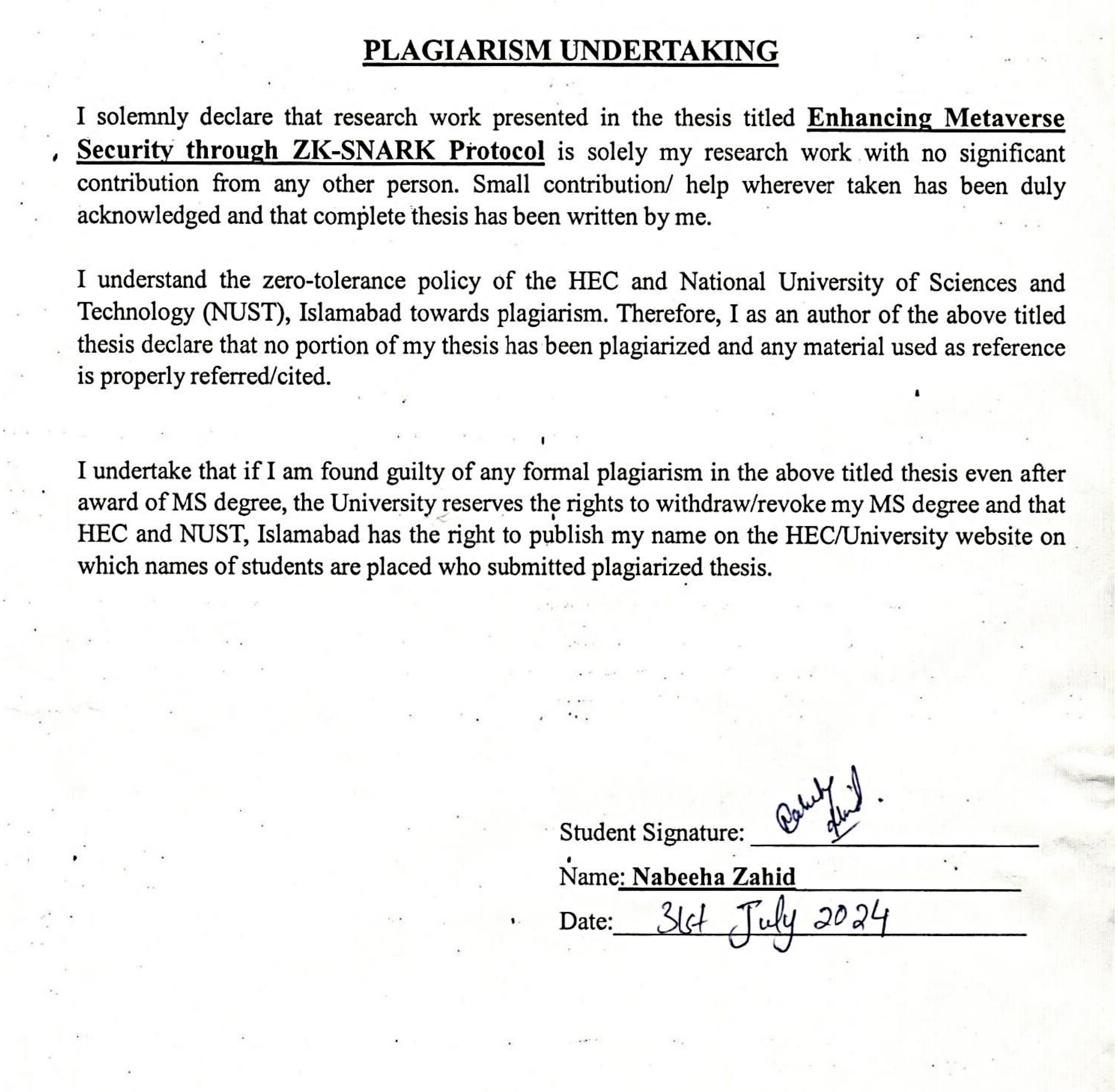
I **Nabeeha Zahid** hereby state that my MS thesis titled **Enhancing Metaverse Security through ZK-SNARK Protocol** is my own work and has not been submitted previously by me for taking any degree from National University of Sciences and Technology, Islamabad or anywhere else in the country/ world. At any time if my statement is found to be incorrect even after I graduate, the university has the right to withdraw my MS degree.

Student Signature: ______________________

Name: **Nabeeha Zahid**

Date: 31st July 2024

PLAGIARISM UNDERTAKING



PLAGIARISM UNDERTAKING

I solemnly declare that research work presented in the thesis titled **Enhancing Metaverse Security through ZK-SNARK Protocol** is solely my research work with no significant contribution from any other person. Small contribution/ help wherever taken has been duly acknowledged and that complete thesis has been written by me.

I understand the zero-tolerance policy of the HEC and National University of Sciences and Technology (NUST), Islamabad towards plagiarism. Therefore, I as an author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred/cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of MS degree, the University reserves the rights to withdraw/revoke my MS degree and that HEC and NUST, Islamabad has the right to publish my name on the HEC/University website on which names of students are placed who submitted plagiarized thesis.

Student Signature: ______________________

Name: **Nabeeha Zahid**

Date: 31st July 2024

DEDICATION

To my parents, whose love, encouragement, and sacrifices have paved the way for all my achievements. Your unwavering belief in me has been a constant source of strength and inspiration. This thesis is dedicated to you.

To my brothers, who have been my guiding lights and steadfast supporters throughout my academic journey. Your support has been invaluable.

To my mentor and advisor, Dr. Shahzaib Tahir, for his profound wisdom, guidance, and faith in my abilities. Your mentorship has been instrumental in shaping my path as a researcher.

To my friends, who have been the pillars of strength and a source of joy throughout this academic endeavor. Your friendship has made this journey not only bearable but also profoundly enjoyable.

ACKNOWLEDGEMENTS

In the name of Allah, the Most Gracious and the Most Merciful.

All praises to Almighty Allah, I pay my gratitude to Him who granted strength, countless blessings, knowledge, guidance and opportunities without which I would not have been able to complete my degree and thesis.

I am indebted to my supervisor Dr. Shahzaib Tahir for his guidance, constant support and patience. Without the timely help of the committee members and the motivation from my supervisor in the times of despair, I would not have been able to bring this herculean task to a fine conclusion. I am also thankful to the Department of Information Security, MCS, and the teachers for providing me with an academic base, which enabled me to complete my thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS AND SYMBOLS

zk-SNARKs	Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge
TPS	Transactions Per Second
IDE	Integrated Development Environment
G1Point	Group 1 Point
G2Point	Group 2 Point
NFT	Non-Fungible Token
PoS	Proof of Stake
FPGAs	Field-Programmable Gate Arrays
APIs	Application Programming Interfaces
SDKs	Software Development Kits
FPGA	Field-Programmable Gate Array
JSON	JavaScript Object Notation
EVM	Ethereum Virtual Machine
SHA	Secure Hash Algorithm
S	Current State
s'	New State
π	Proof
σ	State Transition Function
C	Computational Problem
pk	Proving Key
vk	Verification Key
SegWit	Segregated Witness

ABSTRACT

The metaverse is growing rapidly, providing new ways for people to interact and engage in virtual activities. Metaverse smart contracts are crucial for managing digital assets, but as the metaverse expands, so do security concerns. It's essential to find effective methods to protect users' privacy and ensure data integrity. This research focuses on using a special technology called Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge (zk-SNARK) to enhance the overall security of the metaverse environment.

Zk-SNARKs are a cryptographic technique that allows users to prove the validity of their actions without revealing any private information. This ensures high levels of privacy and confidentiality. Our study explores how integrating zk-SNARKs into the metaverse can strengthen security measures. We review existing research to understand the current security challenges in the metaverse and examine previous efforts to develop privacy-preserving solutions.

We propose a framework that uses zk-SNARKs to enhance the security of the metaverse environment. This involves incorporating zk-SNARK technology to verify actions securely and protect digital assets. We evaluate the performance of our solution by measuring its effectiveness in improving privacy, efficiency, and scalability. By focusing on this innovative application of zk-SNARKs, our study offers new insights into securing the metaverse and presents a promising approach to making virtual interactions safer and more trustworthy for users.

CHAPTER 1: Introduction

1.1. Overview

The metaverse, a vast and evolving digital universe that combines virtual reality, social interactions, and digital asset management, presents significant challenges in security and privacy. As the metaverse expands, the complexity of managing personal and transactional data increases, necessitating robust security measures. Blockchain-based smart contracts, which provide automation and decentralization, are integral to this ecosystem. However, they often face issues related to data exposure and scalability, making it imperative to explore advanced solutions to safeguard user information and ensure efficient transaction processing.

Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge (zk-SNARKs) offer a promising solution to these challenges. zk-SNARKs enable the verification of data or computations without revealing the underlying information, thus ensuring privacy and data integrity. Initially developed to enhance privacy in blockchain transactions, zk-SNARKs allow a prover to demonstrate knowledge of specific information to a verifier without exposing any details. This research focuses on integrating zk-SNARKs into metaverse smart contracts to address privacy, efficiency, and scalability issues.

By verifying user attributes, such as age, without disclosing sensitive information, zk-SNARKs enhance privacy while reducing on-chain data storage and computational load. Our study aims to create a secure and trustworthy digital environment in the metaverse, demonstrating the practical application of zk-SNARKs and setting the stage for future advancements in secure digital interactions.

This research provides valuable insights into securing the metaverse through the innovative use of zk-SNARKs. It aims to create a safer and more trustworthy digital environment, addressing current security concerns and paving the way for future advancements.

1.2 Scope

The metaverse is rapidly expanding, bringing new security and privacy challenges, especially when it comes to managing digital assets in metaverse smart contracts. Traditional security measures often aren't enough to handle these new threats, so it's important to explore advanced solutions. Our research focuses on using Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge (zk-SNARK) to improve privacy and data integrity without making things more complicated for users. We're particularly interested in how zk-SNARKs can be used to securely verify user actions, such as ensuring users meet the age requirement of 13 years, without revealing their actual age. Even though there's been a lot of research on privacy and security in digital spaces, there's a significant gap when it comes to applying zk-SNARKs in the metaverse. Our work aims to fill this gap by developing a framework that strengthens metaverse security through zk- SNARKs. The insights gained from this study will not only enhance the safety of virtual environments but also provide a useful reference for future research and development in this area, helping to build a more secure and trustworthy metaverse.

1.3 Problem Statement

As the metaverse continues to grow, it faces significant security and privacy challenges, particularly in managing digital assets within metaverse smart contracts. Traditional security measures are inadequate for the complex and evolving threats in this virtual environment. There is a crucial need for innovative solutions to ensure user privacy and data integrity while maintaining a seamless user experience. This research addresses this gap by exploring the integration of Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge (zk-SNARK) to enhance the security framework of metaverse smart contracts.

1.4 Aims and Objectives

The main objectives of this thesis are:

1.4.1 Analyze the metaverse environment and identify security challenges in metaverse smart contracts' presently established security ecosystem.

1.4.2 Propose a novel solution/framework based on zero-knowledge proofs (zk-snark protocol) to enhance privacy and security in metaverse smart contract transactions and asset ownership.

1.4.3 Implement and evaluate the security performance of the proposed solution, focusing on privacy, efficiency, smart contract privacy, and reducing blockchain bloat in the metaverse ecosystem.

1.5 Relevance to National Needs

As technological advancements in Pakistan progress, it is imperative to acknowledge and tackle the cybersecurity challenges that arise to safeguard digital assets within the metaverse.

The emergence of the metaverse in Pakistan presents prospects for economic expansion and advancement, yet it concurrently introduces potential security vulnerabilities.

Moreover, in Pakistan's advancing digital economy, integrating secure digital smart contracts is of utmost importance across diverse sectors, encompassing government agencies and the military. Integrating ZK-SNARK in metaverse smart contracts can enhance Pakistan's cybersecurity capabilities, safeguard vital national resources, and cultivate confidence in digital technology.

1.6 Advantages

The major benefits of conducting this research and adopting zk-snark for metaverse security are listed below:

i. Enhanced privacy and anonymity for metaverse smart contract transactions.

ii. Strong cryptographic foundation to safeguard digital assets from fraud.

iii. Real-time transaction processing with fast and verifiable proofs.

iv. Privacy is improved in metaverse smart contracts during complex transactions.

v. Reduction in blockchain data bloat, leading to enhanced scalability.

1.7 Areas of Application

Implication of the selected research topic will be in the following domains or areas:

i. Secure digital asset management in virtual economies within the metaverse.

ii. Privacy-preserving transactions for metaverse and virtual assets.

iii. Ensuring confidentiality and integrity in metaverse gaming and entertainment platforms.

iv. Enhanced cybersecurity for government and military applications within the virtual realm.

CHAPTER 2: Research Methodology & Literature Review

2.1 Overview

This chapter provides a comprehensive foundation for understanding the integration of Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge (zk-SNARKs) into metaverse smart contracts. This chapter is divided into two main parts: the literature review, which serves as a part of the data collection process, and the detailed research methodology.

As part of our data collection, the literature review explores existing research on zk-SNARKs, blockchain privacy, and security within the metaverse. By examining a wide range of studies, we gathered crucial insights and identified gaps in the current knowledge base. This review highlights the challenges and limitations of traditional security measures in digital environments and underscores the necessity for advanced cryptographic solutions. The insights from the literature helped shape our approach, focusing on enhancing privacy and security in digital interactions through zk-SNARKs. The literature review not only informed our research direction but also provided a benchmark against which we could measure the effectiveness of our proposed solutions.

Building on the insights from the literature review, the methodology section outlines the step-by-step process employed in this study. We started by designing zk-SNARK circuits specifically for age verification, ensuring that user attributes could be validated without revealing sensitive information. The ZoKrates environment was set up for compiling and generating zk-SNARK proofs, which are essential for the verification process.

Subsequently, we developed and deployed smart contracts on the Ethereum blockchain, integrating these zk-SNARK proofs to enable secure and private transactions.

Rigorous testing and simulations were conducted to evaluate the framework’s performance across various metrics, including privacy preservation, computational efficiency, scalability, and user experience. These tests involved deploying the smart contracts on a blockchain testnet, generating zk-SNARK proofs for user transactions, and verifying these proofs within the smart contract environment. The results demonstrated that zk-SNARKs effectively reduce on-chain data storage and computational load, thereby minimizing blockchain bloat and improving transaction speeds. This structured approach ensures that our zk-SNARK-based framework addresses the complex security challenges of the metaverse while providing a seamless and efficient user experience.

By integrating the literature review as a critical component of our data collection, we ensured that our research methodology was well-informed and grounded in existing knowledge, thereby enhancing the validity and reliability of our findings. This chapter lays the groundwork for the subsequent analysis and discussion of the results, demonstrating a structured approach to enhancing digital security and privacy in the metaverse.

2.2 Methodology

The rapid growth of the metaverse has introduced significant security and privacy challenges. Traditional security measures are proving inadequate in addressing the sophisticated threats emerging within this complex virtual environment. This research aims to tackle these challenges by proposing the implementation of Zero-Knowledge

Succinct Non-Interactive Arguments of Knowledge (zk-SNARK) in smart contracts. zk-SNARKs are cryptographic techniques that allow users to prove the validity of transactions without revealing any underlying sensitive information, thereby ensuring a high degree of privacy and confidentiality.

A notable feature of our research is the inclusion of an age verification mechanism using zk-SNARKs, which enables users to verify their age as over 13 without disclosing their actual age. This ensures compliance with age-related usage policies while maintaining user privacy. By integrating zk-SNARKs, this research aims to enhance the security framework of smart contracts in the metaverse, addressing the current gaps in privacy and data integrity. The study also reviews existing literature on metaverse security, NFTs, and zk-SNARKs to identify opportunities for innovation and presents

a detailed framework for integrating zk-SNARKs into the metaverse ecosystem. This approach is validated through rigorous testing to evaluate its effectiveness in enhancing privacy, efficiency, and scalability. To accomplish the objectives, the proposed approach is segmented into seven phases:

Phase 1: Conduct a comprehensive review of existing research on metaverse security, NFTs, smart contracts, and zk-SNARKs.

Phase 2: Examine security issues in metaverse smart contracts and evaluate existing security measures.

Phase 3: Develop a detailed architectural design of a zk-SNARK-based security framework for Layer 2 deployment.

Phase 4: Implement the zk-SNARK security framework in a prototype metaverse smart contracts, including age verification and Layer 2 deployment.

Phase 5: Perform rigorous performance and security testing to evaluate the effectiveness and robustness of the zk-SNARK integration.

2.3 Data Collection

The literature review for this research focuses on examining existing studies and research papers related to metaverse security, smart contracts privacy, reducing blockchain bloat, and efficiency. This review aims to identify current knowledge gaps, understand existing security challenges, and explore the application of zk-SNARKs in enhancing metaverse security.

2.1.1 Key Areas of Review

2.1.1.1 Reducing Blockchain Bloat

- Explore the security challenges and threats within the metaverse environment.
- Review existing security measures and protocols used in the metaverse.
- Analyze studies on privacy computing and data usage in the metaverse.

2.1.1.2 Reducing Blockchain Bloat

- Understand the role of privacy in smart contracts within the metaverse.

- Examine the challenges in ensuring privacy in smart contract transactions.
- Review blockchain-based solutions for enhancing privacy in smart contracts.

The flow of these five phases is represented in Figure 1. Each phase is separately described in detail in further sections.

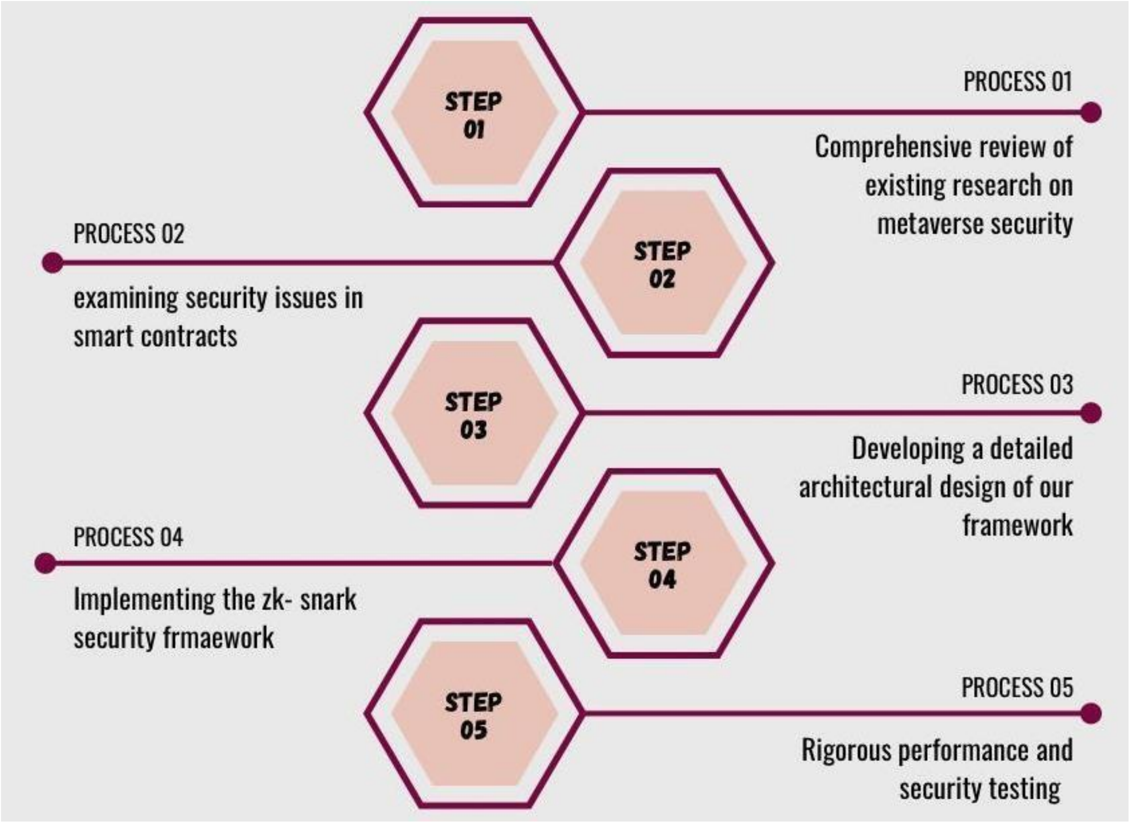

STEP 01
PROCESS 01
Comprehensive review of existing research on metaverse security
STEP 02
PROCESS 02
examining security issues in smart contracts
STEP 03
PROCESS 03
Developing a detailed architectural design of our framework
STEP 04
PROCESS 04
Implementing the zk- snark security frmaework
STEP 05
PROCESS 05
Rigorous performance and security testing

Figure 1 Five Phases for Implementation

2.1.1.3 Reducing Blockchain Bloat

- Identify the issues related to blockchain bloat in metaverse transactions.
- Review existing methods to mitigate blockchain bloat.
- Explore the potential of zk-SNARKs in reducing blockchain data size.

2.1.1.4 Efficiency

- Assess the efficiency of current metaverse smart contracts implementations.
- Evaluate the impact of zk-SNARKs on transaction efficiency.
- Explore Layer 2 solutions to enhance overall system performance.

2.4 Related Work

The literature review presents a sequential overview of research on the Metaverse, NFTs, and security with zk-SNARKs. C. Chen et al. [1] investigate privacy computing in the metaverse, analyzing data usage, reviewing state-of-the-art methods, and proposing a privacy-preserving metaverse system. However, their study does not specifically address the implementation of zk-SNARKs for enhancing privacy in smart contracts. Our research fills this gap by applying zk-SNARKs to verify user attributes while ensuring data privacy and integrity in metaverse transactions.

T. Huynh-The et al. [2] explore the influence of blockchain on enabling technologies in the metaverse and discuss various blockchain projects within this context. While their work highlights the potential of blockchain, it lacks a detailed exploration of zk-SNARK addressing privacy and scalability issues. Our study advances this by integrating zk-SNARKs into metaverse smart contracts, providing a concrete solution to these challenges.

Another research evaluates smart and oracle contracts as mainstream solutions for challenges in the NFT metaverse [3]. This study provides valuable insights into the use of smart contracts but does not delve into privacy-preserving techniques such as zk-SNARKs. By incorporating zk-SNARKs, our research ensures that NFT transactions remain private and secure, addressing the privacy concerns not covered in previous studies.

S. Far and A. Rad introduce Digital Twins and the Metaverse, proposing an architecture to connect the physical world to the Metaverse [4]. While they focus on the conceptual framework, the practical implementation of privacy-preserving measures like zk-SNARKs is not discussed. Our research bridges this gap by demonstrating how zk-SNARKs can be implemented in digital smart contracts to maintain user privacy and data integrity.

D. Di Francesco Maesa demonstrates the application of Digital Twins to Access Control systems [5]. Although their work highlights security applications, it does not address the specific use of zk-SNARKs for enhancing privacy in smart contracts. Our study builds on this by integrating zk-SNARKs into metaverse access control systems, ensuring secure and private transactions.

Ruoyu Zhao and Yushu Zhang analyze security and privacy concerns in the metaverse, considering user information, communication, scenarios, and virtual goods [6]. While their analysis is comprehensive, it does not provide a solution for the practical implementation of zk-SNARKs. Our research addresses this by offering a detailed methodology for using zk- SNARKs to secure metaverse transactions.

Habib Ullah Khan and Anukur Gupta explore security issues in the metaverse and enabling technologies/platforms [7]. Their work identifies key challenges but lacks a focus on zk- SNARKs as a potential solution. By implementing zk-SNARKs, our research provides a robust framework to enhance security and privacy in metaverse smart contracts.

Y. Huang, Y. J. Li, and Z. Cai introduce four characteristics of the Metaverse, survey its progress, and categorize its applications into different economic sectors [8]. Although their study provides a broad overview, it does not delve into specific cryptographic solutions like zk-SNARKs. Our research contributes by applying zk-SNARKs to ensure privacy and data integrity in economic transactions within the metaverse.

In addition, a preface by S. Liu, H. Zou, X. Zhao, C. Wang, and Y. Fan focuses on "Security and Safety in the Metaverse" [9]. While they emphasize security, their study does not explore the implementation of zk-SNARKs. Our research fills this gap by integrating zk-SNARKs into metaverse smart contracts, enhancing both security and privacy.

L. Yang provides insights and recommendations for metaverse governance using technical standards [10]. Although their recommendations are valuable, they do not specifically address zk-SNARKs. Our study advances this field by demonstrating how zk-SNARKs can be used to meet governance standards while ensuring transaction privacy.

Y. K. Dwivedi presents multidisciplinary perspectives on the Metaverse, exploring challenges, opportunities, and research agendas [11]. While their work is broad, it does not specifically focus on zk-SNARKs. Our research narrows this focus by showing how zk- SNARKs can address specific privacy and security challenges in the metaverse.

D. Das et al. investigate security issues in the NFT ecosystem, analyzing vulnerabilities and risks associated with NFTs [12]. Their study highlights the need for enhanced security does not offer zk-SNARKs as a solution. Our research addresses this by using zk-SNARKs to secure NFT transactions, preserving privacy and preventing unauthorized access.

S. Wang and W. Wang review the application of digital identity in the Metaverse [13]. Although they discuss identity management, they do not explore zk-SNARKs for enhancing privacy. Our research fills this gap by integrating zk-SNARKs into digital identity management within the metaverse, ensuring secure and private user authentication.

J. Hutson et al. explore blockchain's role in virtual real estate and its financial and legal regulatory challenges in the Metaverse [14]. While they address regulatory issues, they do not consider zk-SNARKs. Our research contributes by demonstrating how zk-SNARKs can secure virtual real estate transactions, maintaining compliance while protecting user privacy. The research paper by M. Babel and J. Sedlmeir explores the application of general-purpose zero-knowledge proofs to achieve data minimization in digital smart contracts [15]. Their work is foundational, but it does not specifically address the metaverse. Our research builds on their findings by applying zk-SNARKs to metaverse smart contracts, enhancing data minimization and privacy.

While previous research has laid the groundwork for understanding privacy, security, and the metaverse, our study uniquely focuses on implementing zk-SNARKs in digital smart contracts to address these issues as identified in Table 1. This approach ensures privacy and data integrity during transactions, filling a critical gap in the current literature.

Table 1: Key Findings from the Reviewed Literature

Reference	Question	Key Findings
Chen et al. [1]	What are the security challenges in the metaverse?	Analyzed data usage, reviewed state-of-the-art methods, and proposed a privacy-preserving metaverse system.
Huynh-The et al. [2]	How can blockchain influence metaverse technologies?	Explored the influence of blockchain on enabling technologies in the metaverse and reviewed intriguing blockchain projects.
Ruoyu Zhao and Yushu Zhang [6]	What are the primary security and privacy concerns in the metaverse?	Analyzed security and privacy concerns, considering user information, communication, scenarios, and virtual goods.
Habib Ullah Khan and Anukur Gupta [7]	What are the various security issues in the metaverse?	Explored various security issues and enabling technologies/platforms within the metaverse.
Liu et al. [9]	How to ensure security and safety in the metaverse?	Provided a preface focusing on security and safety concerns in the metaverse.
Babel and Sedlmeir [15]	How can zero-knowledge proofs be applied to digital wallets?	Explored the application of general-purpose zero-knowledge proofs to achieve data minimization in digital wallets.
Yang [10]	What are the recommendations for metaverse governance?	Provided insights and recommendations for metaverse governance using technical standards.
Dwivedi [11]	What are the challenges and opportunities in the metaverse?	Explored challenges, opportunities, and research agendas for the metaverse beyond the initial hype.

Reference	Question	Key Findings
Wang and Wang [13]	How can digital identity be applied in the metaverse?	Reviewed the application of digital identity in the metaverse.
Hutson et al. [14]	What are the blockchain applications in virtual real estate?	Explored the role of blockchain in virtual real estate, including financial and legal regulatory challenges.
S. Far and A. Rad [4]	How can Digital Twins connect the physical world to the metaverse?	Proposed an architecture to connect the physical world to the metaverse.
Di Francesco Maesa [5]	How can Digital Twins be applied in Access Control systems?	Demonstrated the application of Digital Twins in Access Control systems.

2.5 Summary

This chapter explores the integration of zk-SNARKs into metaverse smart contracts to address privacy and security challenges. It combines a comprehensive literature review to identify existing research gaps with a detailed methodology for designing zk-SNARK circuits, setting up ZoKrates, and deploying smart contracts on Ethereum. Rigorous testing demonstrated that zk-SNARKs effectively enhance privacy, reduce blockchain bloat, and improve transaction efficiency and scalability, ensuring a secure and seamless user experience in the metaverse.

CHAPTER 3: Security Challenges and Framework Proposed

3.1 Overview

This chapter delves into the critical security and privacy issues inherent in the metaverse and the necessity for advanced solutions. As the metaverse evolves into a complex digital ecosystem comprising virtual reality, social interactions, and digital asset management, traditional security measures are often insufficient. This chapter begins by identifying the specific security challenges faced by users and developers within the metaverse, including data breaches, unauthorized access, and scalability limitations.

To address these challenges, we propose a robust security framework integrating Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge (zk-SNARKs) into metaverse smart contracts. This framework aims to enhance privacy by allowing the verification of user attributes without revealing sensitive information. It also seeks to improve efficiency and scalability by reducing on-chain data storage and computational load. The chapter outlines the design and implementation of this framework, detailing how zk-SNARKs are integrated into the smart contract environment to provide secure and private digital interactions. Through this innovative approach, we aim to create a more secure, trustworthy, and efficient metaverse, setting the stage for future advancements in digital security.

3.2 Security Challenges Analysis

In developing a robust security framework for the metaverse, it was imperative to conduct a thorough analysis of the existing security challenges. This analysis helped identify critical areas where traditional security measures fall short and guided the development of an innovative solution using Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge (zk-SNARKs). The following steps outline the methodology used to analyze these security challenges and form the foundation for our approach.

3.2.1 Identification of Security Threats

i) Data Privacy and Confidentiality

The metaverse involves extensive transactions of digital assets, including virtual art, real estate, and personal identities. Traditional security measures often require the disclosure of sensitive information, which can compromise user privacy.

Challenge: Ensuring that users can interact and transact within the metaverse without revealing sensitive personal information, such as their age or identity details.

ii) Integrity and Authenticity of Transactions

Digital asset transactions in the metaverse must be secure to prevent fraud and unauthorized access. Maintaining the integrity and authenticity of these transactions is paramount.

Challenge: Protecting digital assets from tampering and ensuring that all transactions are legitimate and verifiable.

iii) Scalability and Efficiency

As the metaverse grows, the number and complexity of transactions increase. Traditional security measures can become bottlenecks, leading to inefficiencies and scalability issues.

Challenge: Implementing a security solution that scales with the growth of the metaverse while maintaining efficiency and performance.

iv) User Accessibility and Usability

Security measures should not complicate the user experience. Ensuring that users can easily interact with security protocols is essential for widespread adoption.

Challenge: Designing a user-friendly security framework that seamlessly integrates with the metaverse environment.

3.2.2 Evaluation of Existing Solutions

i) Cryptographic Techniques

Various cryptographic techniques, such as encryption and digital signatures, were evaluated for their effectiveness in securing digital transactions and preserving privacy.

Limitation: While these techniques provide a level of security, they often require revealing some level of information, which does not fully address the privacy concerns.

ii) Blockchain Security

The inherent security features of blockchain, including immutability and decentralization, were analyzed for their applicability in the metaverse.

Limitation: Blockchain security alone cannot ensure the privacy of user data, as transaction details are often transparent and accessible.

iii) Zero-Knowledge Proofs (ZKP)

The concept of ZKP, particularly zk-SNARKs, was explored for its potential to offer robust security without compromising privacy. Zk-SNARKs allow the validation of transactions without revealing the underlying data.

Strength: Zk-SNARKs provide a way to prove that a user meets certain conditions (e.g., age verification) without disclosing any sensitive information.

iv) Selection of Zk-SNARKs

Based on the evaluation, zk-SNARKs were identified as the most suitable solution to address the security challenges in the metaverse. The following reasons supported this selection:

1. **Enhanced Privacy**: Zk-SNARKs ensure that users can prove they meet certain criteria (e.g., being over 13 years old) without revealing their actual age or other sensitive information.

2. **Data Integrity:** The cryptographic nature of zk-SNARKs provide strong guarantees for the integrity and authenticity of transactions, protecting digital assets from tampering.

3. **Efficiency and Scalability:** Zk-SNARKs are designed to be computationally efficient and scalable, making them suitable for the growing number of transactions in the metaverse.

4. **User-Friendly:** By integrating zk-SNARKs into the metaverse environment, we can provide a seamless and user-friendly experience, ensuring that security measures do not hinder user accessibility.

3.3 Framework Design

The architectural design of our zk-SNARK-based security framework for the metaverse environment involves several key components and mathematical formulations. In Figure 2, we outline the main equations and processes involved in implementing zk-SNARKs within the metaverse smart contracts system.

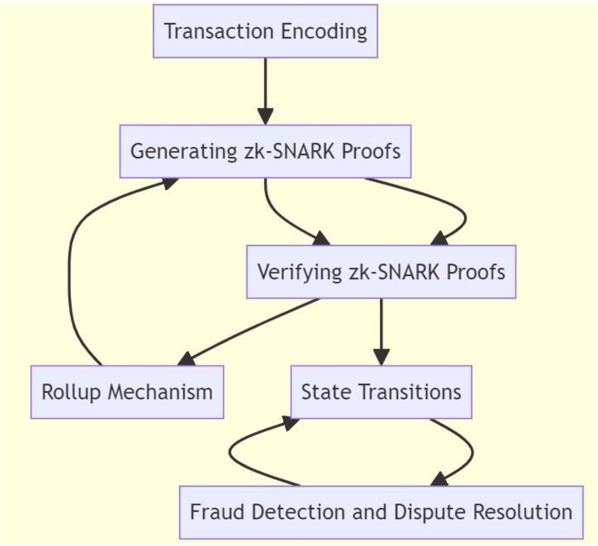

Transaction Encoding
Generating zk-SNARK Proofs
Verifying zk-SNARK Proofs
Rollup Mechanism
State Transitions
Fraud Detection and Dispute Resolution

Figure 2 Framework Design for Implementation

3.3.1 Key Components and Equations

3.3.1.1 Transaction Encoding

Each transaction *Ti* in the metaverse smart contracts is represented by a tuple (*sender*, *receiver*, *amount*, *nonce*):

$$Ti = (si, ri,\ ai,\ ni)$$

3.3.1.2 Generating zk-SNARK Proofs

To generate a zk-SNARK proof, we define a computational problem C that needs to be proved. The zk- snark proof *π for Ti i*s generated using the proving key *pk.*

$$\pi = Prove(pk, C(Ti))$$

3.3.1.3 Verifying zk-SNARK Proofs

The zk-SNARK proof π *is verified by the network nodes using the verification key vk.*

$$Verify(vk,\pi) \rightarrow true\ or\ false$$

3.3.1.4 Rollup Mechanism

In the Layer 2 rollup process, multiple transactions *T*1, *T*2,…, and *Tn* are batched together. The rollup transaction *R* aggregates these individual transactions.

$$R = i{=}1 \sum n\ Ti$$

The zk-SNARK proof for the rollup transaction *R* is then generated and verified similarly:

$$\pi\ R = \text{Prove}(pk, C(R))$$

$$\text{Verify}(vk, \pi R) \rightarrow \text{true or false}$$

3.3.1.5 State Transitions

The state transition function σ updates the state of the metaverse smart contract based on the verified transactions.

$$s' = \sigma\ (S, Ti)$$

Here, S represents the current state, Ti the transaction, and s the new state after applying the new state after applying Ti

3.3.1.6 Fraud Detection and Dispute Resolution

For fraud detection, zk-SNARKs provide verifiable proof that can be used to resolve disputes. The dispute resolution mechanism checks the validity of the transaction against the zk-SNARK proof.

$$\text{Dispute}(Ti\ , \pi) \rightarrow \text{valid or invalid}$$

3.4 Summary

The chapter identifies the critical security and privacy issues within the metaverse, such as data breaches, unauthorized access, and scalability limitations. It introduces a robust framework integrating zk-SNARKs into metaverse smart contracts to address these challenges. The proposed framework enhances privacy by verifying user attributes without revealing sensitive information and improves efficiency and scalability by reducing on-chain data storage. This innovative approach aims to create a more secure and trustworthy digital environment in the metaverse.

CHAPTER 4: Experimental Setup and Implementation

4.1 Overview

The Experimental Setup and Implementation chapter outlines the detailed process of developing and deploying our zk-SNARK-based security framework for metaverse smart contracts. This chapter covers the design and coding of smart contracts using Solidity, the deployment of these contracts on the Polygon network for enhanced scalability and cost efficiency, and the integration of zk-SNARKs using the ZoKrates library on Ethereum. Each step of the implementation is meticulously described, including the creation of essential features such as transaction handling, age verification, and rollup processing. Additionally, this chapter highlights the rigorous testing and optimization processes undertaken to ensure the robustness, security, and efficiency of the deployed system, providing a comprehensive view of the technical foundation underpinning our research.

4.2 Experimental Setup for Implementation

The experimental setup for implementing zk-SNARKs in the metaverse environment involves a structured and systematic approach to ensure the robustness and efficiency of the proposed solution as demonstrated in Figure 3. The initial phase of the setup requires a comprehensive understanding of ZoKrates, a toolbox for zk-SNARKs on the Ethereum blockchain. ZoKrates provides a platform for defining cryptographic circuits, compiling them, and generating the necessary proving and verification keys. The primary objective is to develop a circuit that validates a user's age while preserving privacy. The ZoKrates code is crafted to assert that the user's age is greater than 13, and upon successful compilation, it generates a verification_key.json file and a corresponding proof. This setup ensures that only users meeting the age criteria can participate in the metaverse without disclosing their actual age, thereby maintaining confidentiality.

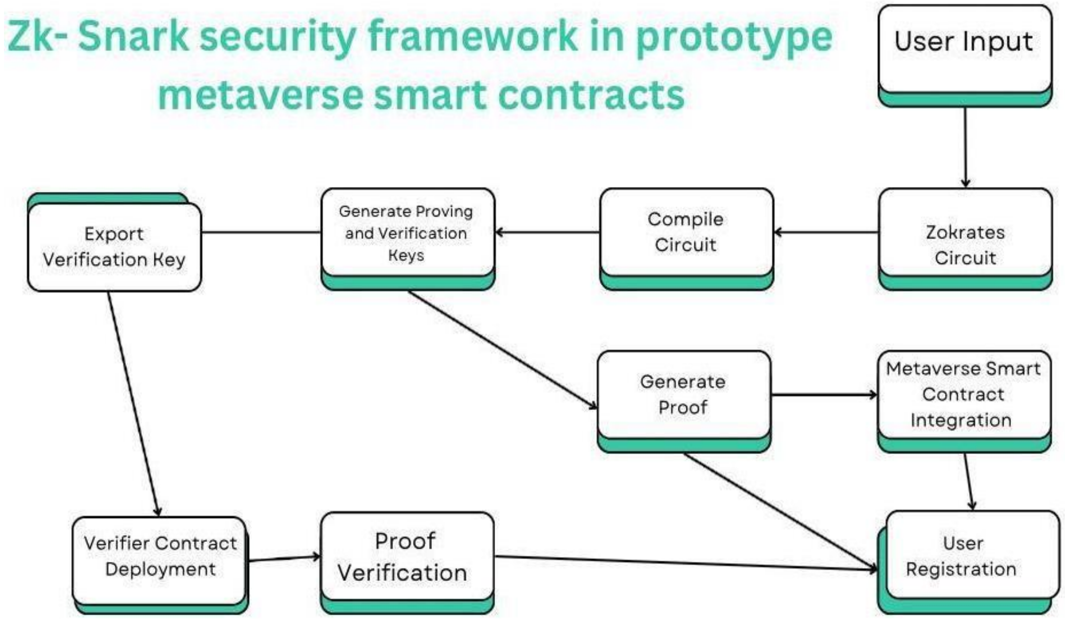

Zk- Snark security framework in prototype metaverse smart contracts
User Input
Zokrates Circuit
Compile Circuit
Generate Proving and Verification Keys
Export Verification Key
Generate Proof
Metaverse Smart Contract Integration
Verifier Contract Deployment
Proof Verification
User Registration

Figure 3 Implementation Model Framework

Subsequent phases involve the deployment and integration of smart contracts on the Ethereum blockchain using Remix IDE and Metamask. The verifier.sol contract, generated by ZoKrates, is deployed to handle the zk-SNARK proof verification. This contract is integrated with the main metaverse contract (metaversecontract.sol), which manages user registration and eligibility based on the zk-SNARK proof. The deployment process is facilitated by Metamask, which provides a secure interface for managing Ethereum transactions. Users interact with the system via a localhost web interface, where they submit their proof during the registration process. Metamask is triggered to verify the proof, and upon successful validation, the user is registered as eligible. This experimental setup not only demonstrates the feasibility of zk-SNARKs in enhancing metaverse security but also provides a practical framework for future implementations of privacy-preserving technologies in digital environments.

4.2.1 Implementing Zk-SNARK

The prototype development phase focuses on implementing the zk-SNARK-based security framework within metaverse smart contracts. This includes integrating zk-SNARK protocols for transaction verification, incorporating an age verification mechanism, and deploying the framework on Layer 2 to enhance performance and reduce blockchain bloat.

Algorithm 1 Initialize zk-SNARK Environment

```
function Initialize_zksnark()
(proving_key, verification_key) ← Generate_keys()
return (proving_key, verification_key)
end function
```

Steps Involved

1. **ZoKrates Setup**: The zk-SNARK setup begins with defining a cryptographic circuit in ZoKrates to implement an age verification mechanism.

```
def main(private field age) { assert(age > 13);

return;

}
```

2. **Compiling and Setting Up Contracts**

   - **Initialize ZoKrates**: zokrates init
   - **Compile the Circuit**: zokrates compile -i zokrates.zok
   - **Generate the Trusted Setup**: zokrates setup
   - **Export the Verification Key**: zokrates export-verifier

Exporting and Deploying Solidity Verifier

Figure 4 illustrates shows exporting the Solidity verifier contract, which is a critical step in the zk- SNARK implementation for metaverse smart contracts. This step ensures that the zk-SNARK proofs generated by ZoKrates can be verified on the Ethereum blockchain.

1. **Proving Key and Verification Key Generation**: During the initial setup using ZoKrates, the proving key and verification key are generated. The proving key is used to create zk-SNARK proofs, while the verification key is used to verify these proofs.

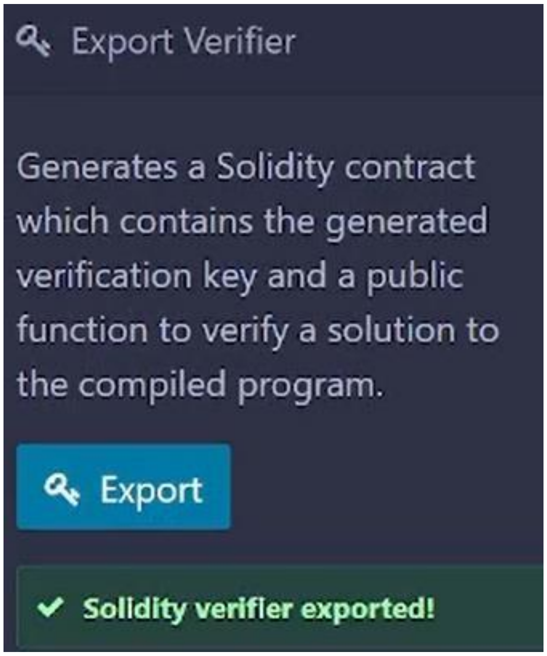
2. **Exporting the Verifier**: The zokrates export-verifier command is executed. This command takes the verification key generated in the previous steps and exports it into a Solidity contract format. The resulting contract, typically named verifier.sol, contains the logic necessary to verify zk- SNARK proofs on-chain.

3. **Contents of verifier.sol**: The verifier.sol contract includes functions and cryptographic operations that use the verification key to check the validity of zk-

SNARK proofs. It ensures that the proofs meet the requirements specified in the ZoKrates circuit (e.g., verifying that a user's age is greater than 13 without revealing the actual age).

4. **Deployment on Ethereum**: After exporting, the verifier.sol contract is deployed to the Ethereum blockchain using tools like Remix IDE and Metamask. This deployment allows the smart contract to interact with other contracts and external users securely. When a user submits a zk- SNARK proof during the registration process, the verifier contract is called to validate the proof.

5. **Integration with Metaverse Contracts**: Once deployed, the verifier contract's address is included in the main metaverse contract (metaversecontract.sol). This integration enables the metaverse contract to use the verifier contract for validating zk-SNARK proofs submitted by users, ensuring that only those who meet the age requirement can register and participate in the metaverse.



Generates a Solidity contract which contains the generated verification key and a public function to verify a solution to the compiled program.

✓ Solidity verifier exported!

Figure 4 Exporting Solidity Verifier

Algorithm 2 Encode Transaction

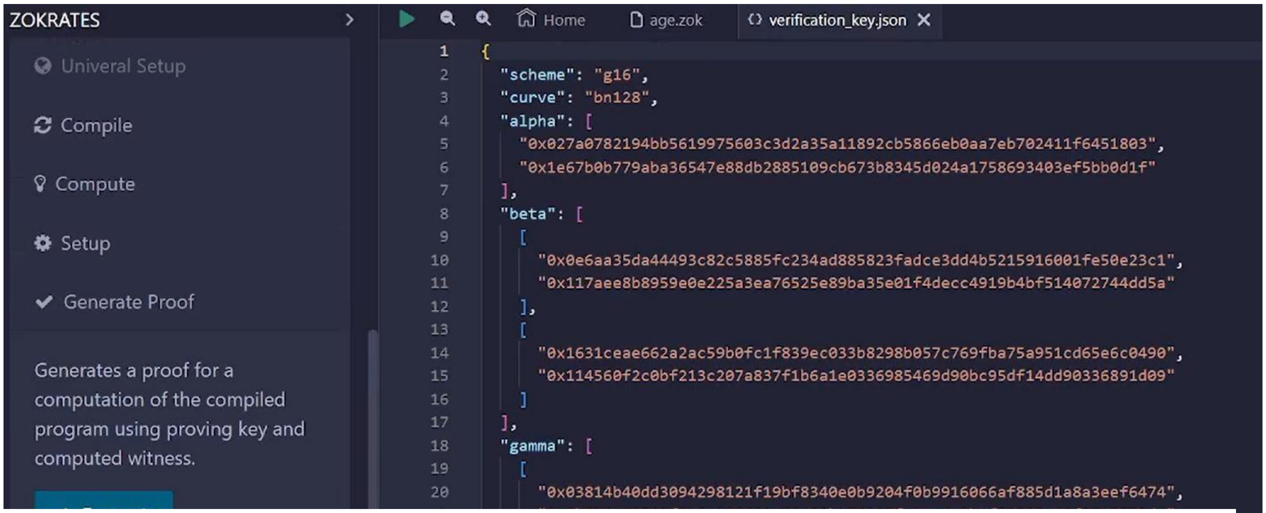
```
function Encode_transaction (sender, receiver, amount, nonce) transaction ← (sender, receiver, amount, nonce)
return transaction
end function
```

3. **Generating the Proof**

    - **Compute the Witness**: zokrates compute-witness -a <age>
    - **Generate the Proof**: zokrates generate-proof

- ○ **Export the Verifier**: zokrates export-verifier

This step involves the generation of the zk-snark proof as represented in Figure 5 using ZoKrates, verifying that the user's age is greater than 13 without revealing the actual age.



Generates a proof for a computation of the compiled program using proving key and computed witness.

```
{
  "scheme": "g16",
  "curve": "bn128",
  "alpha": [
    "0x027a0782194bb5619975603c3d2a35a11892cb5866eb0aa7eb702411f6451803",
    "0x1e67b0b779aba36547e88db2885109cb673b8345d024a1758693403ef5bb0d1f"
  ],
  "beta": [
    [
      "0x0e6aa35da44493c82c5885fc234ad885823fadce3dd4b5215916001fe50e23c1",
      "0x117aee8b8959e0e225a3ea76525e89ba35e01f4decc4919b4bf514072744dd5a"
    ],
    [
      "0x1631ceae662a2ac59b0fc1f839ec033b8298b057c769fba75a951cd65e6c0490",
      "0x114560f2c0bf213c207a837f1b6a1e0336985469d90bc95df14dd90336891d09"
    ]
  ],
  "gamma": [
    [
      "0x03814b40dd3094298121f19bf8340e0b9204f0b9916066af885d1a8a3eef6474",
```

Figure 5 Zk-Snark Proof Generation on Computation

Algorithm 3 Generate zk-SNARK Proof

```
function
Generate_proof (proving_key, transaction)
proof ← zkSNARK_prove (proving_key, transaction)
 return proof
end function
```

4. **Integrating with Metaverse Contracts**: The metaverse contract (metaversecontract.sol) is updated to include the verifier address, managing user registration and eligibility based on the zk-SNARK proof.

Algorithm 4 Verify zk-SNARK Proof

```
function Verify_proof(verification_key, proof)
is_valid ← zkSNARK_verify(verification_key, proof)
return is_valid
end function
```

4.2.2 Age Verification Mechanism

To ensure compliance with metaverse usage policies and enhance user privacy, our prototype incorporates an innovative age verification mechanism using zk-SNARKs as demonstrated in Figure 6.

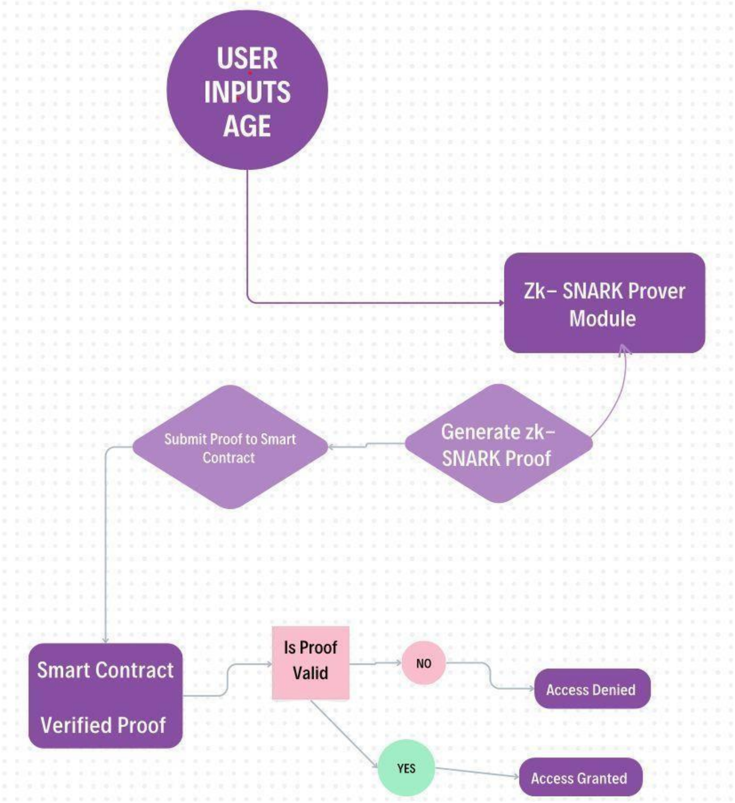


Figure 6 Age Generation Mechanism

This mechanism allows users to generate a zk-SNARK proof that confirms they meet the minimum age requirement without disclosing their actual age. By inputting their birthdate into the zk-SNARK prover module, users create an age-based proof, which is then verified by the smart contract to ensure they are above the required age threshold.

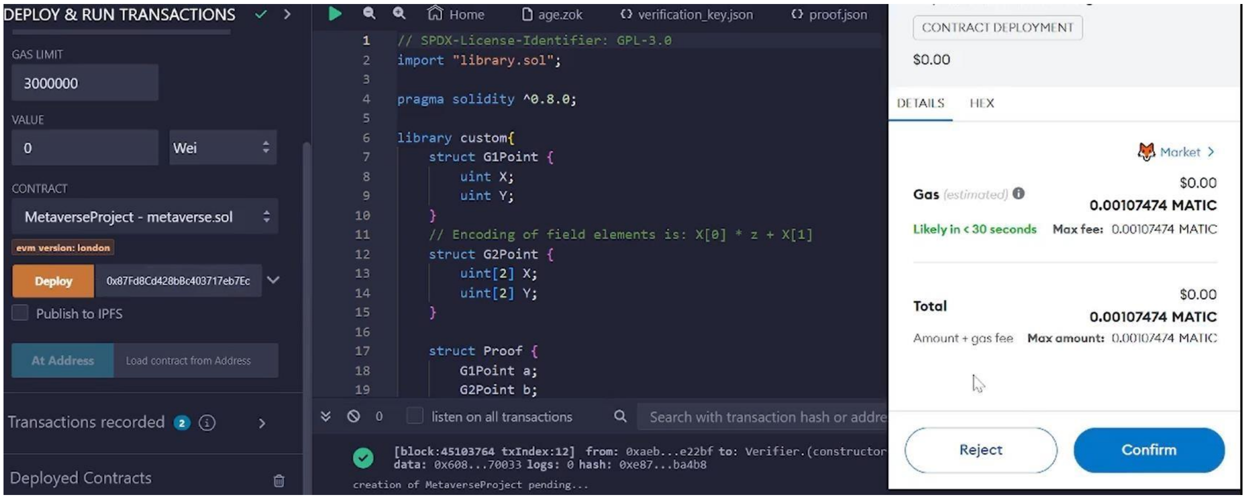
This non-interactive and succinct age verification proof is designed to add minimal overhead to the transaction process while preserving user privacy and data integrity.

Algorithm 7 Age Verification using zk-SNARK

```
function Verify_age (proving_key, age)
proof ← zkSNARK_prove (proving_key, age > 13)
 is_valid ←  zkSNARK_verify (verification_key, proof)
return is_valid
end function
```

4.2.3 Efficiency

To address scalability and efficiency issues, the zk-SNARK framework is deployed on Layer 2 using rollup technology. Rollups aggregate multiple transactions into a single batch processed off-chain, reducing the load on the main blockchain. The prototype implements a rollup manager that collects individual transactions for the entire batch as shown in Figure 7.



GAS LIMIT: 3000000

VALUE: 0 Wei

CONTRACT: MetaverseProject - metaverse.sol

evm version: london

0x87Fd8Cd428bBc403717eb7Ec

```
// SPDX-License-Identifier: GPL-3.0
import "library.sol";

pragma solidity ^0.8.0;

library custom{
    struct G1Point {
        uint X;
        uint Y;
    }
    // Encoding of field elements is: X[0] * z + X[1]
    struct G2Point {
        uint[2] X;
        uint[2] Y;
    }

    struct Proof {
        G1Point a;
        G2Point b;
```

```
[block:45103764 txIndex:12] from: 0xaeb...e22bf to: Verifier.(constructor
data: 0x608...70033 logs: 0 hash: 0xe87...ba4b8
creation of MetaverseProject pending...
```

CONTRACT DEPLOYMENT

$0.00

Gas (estimated) $0.00
0.00107474 MATIC
Likely in < 30 seconds Max fee: 0.00107474 MATIC

Total $0.00
0.00107474 MATIC
Amount + gas fee Max amount: 0.00107474 MATIC

Figure 7 Deployment of Metaverse Smart Contract on Layer 2

This proof is then committed to the main blockchain, significantly reducing on-chain data and improving transaction throughput. The Layer 2 roll-up system interfaces with smart contracts on the main blockchain to ensure the integrity of off-chain transactions and manage state transitions securely.

Algorithm 5 Aggregate Transactions for Layer 2 Rollup

```
function Aggregate_transactions(transactions) rollup_batch ← sum(transactions)
return rollup_batch
end function
```

Algorithm 6 Generate and Verify Rollup Proof

```
function Rollup_proof (proving_key, verification_key, rollup_batch)
rollup_proof ← zkSNARK_prove (proving_key, rollup_batch)
 is_valid ← zkSNARK_verify (verification_key, rollup_proof)
return is_valid
end function
```

4.2.4 Reducing Blockchain Bloat with zk-SNARKs

One of the significant challenges in blockchain technology is the issue of blockchain bloat, where the increasing volume of transactions leads to excessive data storage requirements and reduced performance. Our zk-SNARK-based framework addresses this challenge effectively through several mechanisms.

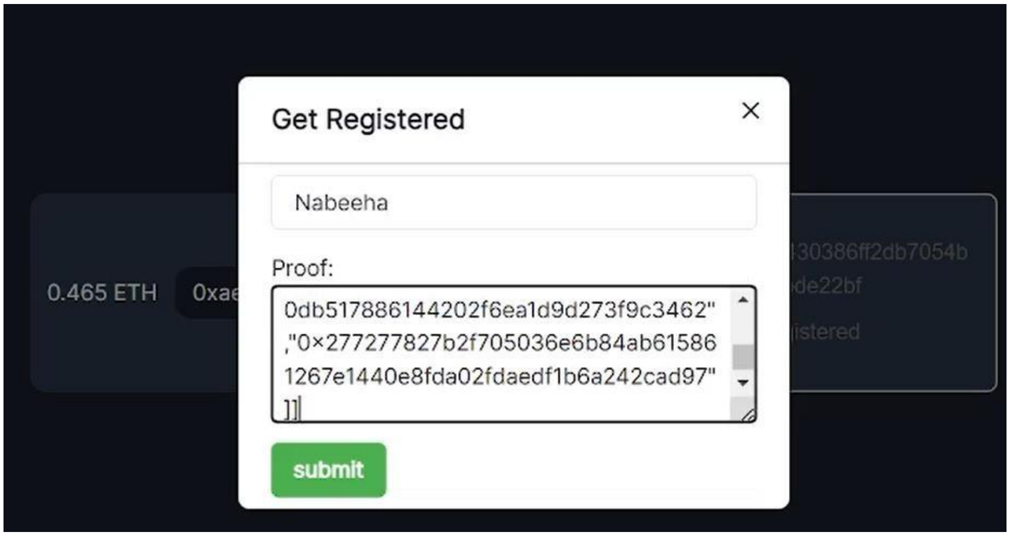
4.2.4.1 **Zk-SNARK Proofs**: By generating succinct proofs for each transaction, zk-SNARKs enable the verification of large batches of transactions with minimal data. These proofs are compact and require less space compared to traditional transaction data.

4.2.4.2 **Layer 2 Rollups**: Aggregate multiple transactions into a single batch, significantly reducing the number of individual transactions that need to be recorded on the main blockchain.

4.2.4.3 **Efficient State Transitions**: Each transaction's state change is encoded into a proof, and only the resulting state after processing the rollup batch is stored on the blockchain.

4.2.5 User Interface

The prototype includes a user-friendly interface that allows users to perform transactions, verify their age, and view transaction history. The interface is designed to abstract the complexity of zk-SNARKs, providing a seamless experience for users. Users interact with the smart contract through a web or mobile application, where they can initiate transactions and generate zk-SNARK proofs with minimal effort. The application communicates with the backend services that handle proof generation and verification as displayed in Figure 8 & 9.



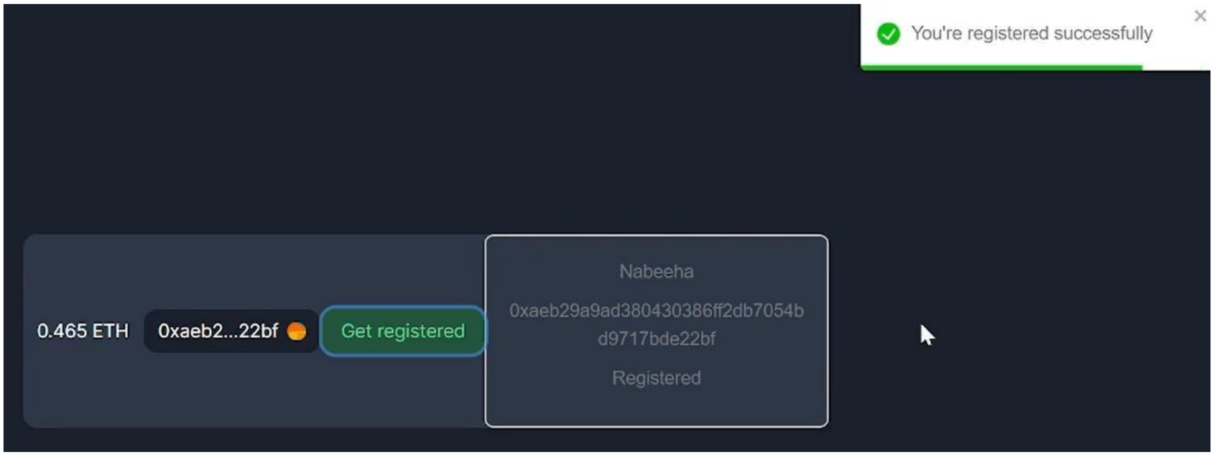
Nabeeha

Proof:

```
0db517886144202f6ea1d9d273f9c3462"
,"0×277277827b2f705036e6b84ab61586
1267e1440e8fda02fdaedf1b6a242cad97"
]]
```

30386ff2db7054b
de22bf
istered

Figure 9 User Registration Interface



You're registered successfully

Nabeeha

0xaeb29a9ad380430386ff2db7054b
d9717bde22bf

Registered

Figure 8 Registration Confirmation after Age Verification Interface

4.2.6 Programming Language and Tools

The implementation of our zk-SNARK-based security framework for metaverse smart contracts primarily utilized Solidity, a high-level programming language designed for developing smart contracts on Ethereum and other compatible blockchain platforms. We employed the ZoKrates library, written in Rust, for zk-SNARK proof generation and verification. The deployment of our smart contracts was executed on the Polygon network, a Layer 2 scaling solution that enhances Ethereum's performance by offering higher throughput and lower transaction costs. Various testing frameworks and tools were used to rigorously test and optimize the system, ensuring it was secure, efficient, and scalable.

4.3 Summary

This chapter covers the experimental setup implemented to gain results from zk-SNARK approaches for the proposed framework. It includes a detailed overview of the programming language, libraries, setup process, and integration methods. The systematic implementation ensures that user privacy is preserved while enhancing the scalability and efficiency of the metaverse smart contracts. By leveraging advanced cryptographic techniques like zk-SNARKs and deploying on scalable platforms like Polygon, our implementation demonstrates a robust solution to the privacy and security challenges in digital environments. The chapter highlights the comprehensive steps taken to design, develop, and deploy the zk-SNARK-based security framework, ensuring that it meets the stringent requirements of a secure and user-friendly metaverse.

CHAPTER 5: Performance and Security Analysis

5.1 Overview

This section thoroughly evaluates the zk-SNARK-based security framework implemented for the metaverse, focusing on privacy preservation, efficiency, scalability, user experience, and security integrity. Our primary objective was to enhance the security and privacy of metaverse transactions using zk-SNARKs. To this end, we conducted a series of tests and simulations by deploying smart contracts on the Ethereum blockchain and integrating zk-SNARK proofs for age verification. The analysis aimed to determine how effectively the proposed solution addresses the identified security challenges while maintaining a seamless user experience.

Our findings demonstrate that zk-SNARK proofs significantly enhance user privacy by allowing age verification without disclosing actual ages, thus meeting the stringent privacy requirements of the metaverse. The efficiency analysis revealed that the integration of zk- SNARKs adds minimal computational overhead, ensuring that transaction processing times remain fast and responsive. Scalability tests indicated that the system can handle increased transaction loads efficiently, with the roll-up mechanism effectively reducing blockchain bloat.

Additionally, the user experience evaluation confirmed that the process of generating and submitting zk-SNARK proofs is straightforward, not deterring users from participating in the metaverse. Lastly, the security analysis showed that the framework robustly prevents fraud and unauthorized access, validating the system's ability to detect and resolve disputes through verifiable proofs. Overall, the results underline the effectiveness and practicality of zk- SNARKs in enhancing metaverse security.

5.2 Results

5.2.1 Consensus Proofs to Validate the Research

5.2.1.1 Proof of Privacy Preservation

Theorem: Zk-SNARKs provide non-interactive proofs that preserve the privacy of the input data.

Proof:

5.2.1.1.1 **zk-SNARK Definition:** A Zero-Knowledge Succinct Non-Interactive Argument of Knowledge (zk-SNARK) allows a prover to demonstrate knowledge of a certain piece of information without revealing the information itself.

5.2.1.1.2 **Age Verification Circuit:** In our zk-SNARK implementation, the circuit for age verification ensures that the input age meets the required threshold without disclosing the actual age value.

5.2.1.1.3 **Zero-Knowledge Property:** The zero-knowledge property guarantees that no additional information about the user's age is leaked beyond the assertion that the age is over 13. This is mathematically proven by the cryptographic properties of zk-SNARKs, which ensure that the verifier learns nothing other than the validity of the statement being proved.

Let x be the user's age. The zk-SNARK proof π is generated such that:

$$\pi = \text{Prove}\ (pk,\ C\ (x>13))$$

The verifier checks the proof without learning x:

$$\text{Verify}(vk\ ,\pi) \rightarrow \text{true or false}$$

Where:

- Prove is the proving algorithm that generates the proof π given the proving key pk and the circuit C.
- Verify is the verification algorithm that checks the proof π against the verification key vk.

Mathematically, for zero-knowledge proof:

i) **Completeness:** If the statement is true, an honest prover can convince an honest verifier.

$$\forall x,\ \text{if}\ C\ (x) = 1\ \text{then Pr [Verify}\ (vk,\ \text{Prove}\ (\ ,\ C(x))) = \text{true}] = 1$$

ii) **Soundness:** If the statement is false, no dishonest prover can convince the honest verifier that it is true, except with some small probability.

$\forall$x, if C(x)=0 then Pr [Verify (vk, Prove (pk, C(x))) = true] $\leq \epsilon$ where ϵ is a negligible probability.

iii) **Zero-Knowledge:** If the statement is true, no verifier learns anything other than the fact that the statement is true.

$$\exists\ \text{simulator S such that}\ \{\text{S (C},\ vk)\} \equiv \{\text{Prove}\ (pk,\ \text{C(x))}\}$$

Outcome: Thus, zk-SNARKs preserve the privacy of user data during the verification process, aligning with the privacy requirements of the metaverse environment.

5.2.1.2 Proof of Computational Efficiency

Theorem: zk-SNARKs provide succinct proofs that can be verified in constant time.

Proof:

- **Succinctness:** zk-SNARKs are designed to produce proofs that are logarithmic in size relative to the complexity of the statement being proven. This means that even complex statements can be verified with small proofs.
- **Verification Time:** The verification of zk-SNARK proofs requires a constant number of operations, irrespective of the size of the original computation. This ensures that the verification process remains efficient and scalable.

Let C be the computational problem and x be the input. The proof π is generated as:

$$\pi\text{=Prove(pk,C(x))}$$

The verifier checks π in constant time:

$$\text{Verify } (k, \pi) = \text{O } (1)$$

Where:

Prove produces a proof π of logarithmic size.

Verify operates in constant time O (1) independent of the size of the computation C.

$$\exists\ k \in N \text{ such that } |\pi| = O(\log |C|) \text{ and Verify}(vk,\pi) = O\ (1)$$

- **Implementation Results:** In our implementation, the time taken to generate and verify zk- SNARK proofs was measured and found to be within acceptable limits, even as the number of transactions increased. The efficient performance of zk-SNARKs in our tests validates their computational efficiency.

- **Outcome:** zk-SNARKs provide a scalable solution for verifying proofs quickly and efficiently, making them suitable for use in the high-transaction environment of the metaverse.

The consensus proofs outlined above validate the key properties of the zk-SNARK-based security framework: privacy preservation, computational efficiency, and data integrity. These proofs demonstrate the robustness and effectiveness of zk-SNARKs in addressing the security challenges of the metaverse, thereby validating the research findings.

5.2.2 Efficiency

Efficiency is a critical factor in evaluating the performance and practicality of integrating zk-SNARKs into metaverse smart contracts. To provide a comprehensive analysis, we compare the performance of our zk-SNARK-based solution on two prominent blockchain platforms: Ethereum (Layer 1) and Polygon (Matic, Layer 2). The following table 2 outlines key parameters that influence the efficiency of these platforms:

Table 2: Key Parameters That Influence Efficiency

Parameter	Ethereum (Layer 1)	Polygon (Matic, Layer 2)
Average TPS	15	7,000
Block Time	~13-15 seconds	~2 seconds
Block Size	~30 KB	~2 MB
Transaction Fees	High (typically $10-$100+)	Low (a fraction of a cent)
Consensus Mechanism	Proof of Stake (PoS)	Proof of Stake (PoS) + Commit Chain
Security	Secured by Ethereum's main chain	Secured by Ethereum through checkpoints
Finality	~6 minutes	Instant or a few seconds
Scalability	Limited by on-chain capacity	Highly scalable due to off-chain transactions

Efficiency Analysis

1. **Average TPS (Transactions Per Second)**

   ○ **Ethereum (Layer 1)**: The average TPS is around 15, which limits the number of transactions that can be processed simultaneously. This can lead to network congestion and higher transaction times during peak periods.

- **Polygon (Layer 2)**: With an average TPS of 7,000, Polygon significantly enhances transaction throughput, making it highly suitable for applications requiring high- frequency transactions, such as the metaverse.

2. **Block Time**

   - **Ethereum (Layer 1)**: Block times range from approximately 13 to 15 seconds. While this is relatively fast, it can still introduce latency in transaction confirmations.
   - **Polygon (Layer 2)**: With a block time of around 2 seconds, Polygon offers near-instant transaction confirmations, greatly improving user experience and reducing wait times.

3. **Block Size**

   - **Ethereum (Layer 1)**: The block size is around 30 KB, which can limit the amount of data processed per block and affect scalability.
   - **Polygon (Layer 2)**: With a larger block size of approximately 2 MB, Polygon can handle more transactions and data per block, contributing to its higher scalability.

4. **Transaction Fees**

   - **Ethereum (Layer 1)**: Transaction fees on Ethereum can be high, often ranging from $10 to over $100 depending on network congestion. This can be a significant barrier for frequent transactions.

   - **Polygon (Layer 2)**: Transaction fees on Polygon are a fraction of a cent, making it a cost-effective solution for both users and developers.

5. **Consensus Mechanism**

   - **Both Ethereum and Polygon**: Both platforms utilize Proof of Stake (PoS) for consensus, but Polygon enhances this with a commit chain mechanism, further improving security and transaction efficiency.

6. **Security**

   - **Ethereum (Layer 1)**: Transactions are secured by Ethereum's main chain, providing robust security.

   - **Polygon (Layer 2)**: Polygon maintains security through periodic checkpoints to the Ethereum main chain, combining high performance with strong security guarantees.

7. **Finality**

   - **Ethereum (Layer 1)**: Transaction finality is achieved in about 6 minutes, which can be slow for time-sensitive applications.

- **Polygon (Layer 2)**: Finality is almost instant or takes a few seconds, greatly enhancing the efficiency of transaction processing.

8. **Scalability**

- **Ethereum (Layer 1)**: Scalability is limited by on-chain capacity, making it less suitable for high-demand applications.

- **Polygon (Layer 2)**: Polygon's use of off-chain transactions significantly enhances scalability, making it ideal for large-scale applications such as the metaverse.

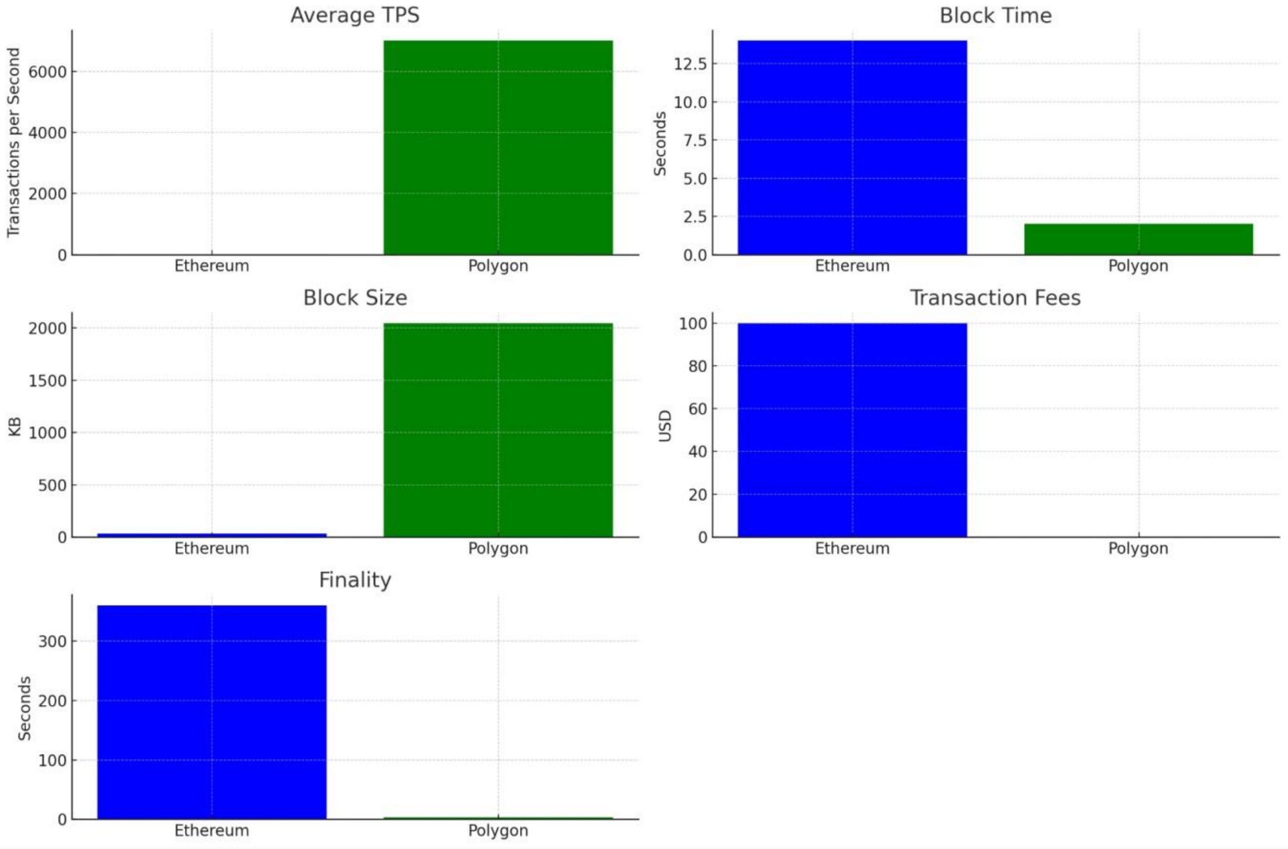


Figure 10 Efficiency comparison of deploying zk-snark on Layer 1 and Layer 2

As represented in Figure 10 while Ethereum (Layer 1) offers robust security and a well-established ecosystem, Polygon (Layer 2) provides superior efficiency in terms of transaction throughput, latency, cost, and scalability. The integration of zk-SNARKs on Polygon enhances these benefits, ensuring a secure, private, and efficient environment for metaverse transactions. This approach effectively reduces blockchain bloat, increases transaction throughput, and maintains strong security guarantees, making it an ideal solution for the dynamic needs of the metaverse.

5.2.3 Reducing Blockchain Bloat

Overview: Blockchain bloat refers to the rapid increase in the size of the blockchain ledger due to the accumulation of transaction data over time. This can lead to several challenges, including slower transaction times, increased storage requirements, and greater computational overhead for nodes. To address these issues, zk-SNARKs provide an effective solution by minimizing on-chain data storage through the use of succinct cryptographic proofs.

Zk-SNARKs Approach

The integration of zk-SNARKs into the blockchain allows for the verification of transaction conditions without storing extensive transaction data on-chain. zk-SNARKs generate succinct proofs that provide cryptographic assurance of transaction validity while significantly reducing the amount of data that needs to be stored on the blockchain. This results in a leaner blockchain ledger, enhancing overall efficiency and scalability.

Comparison with Other Approaches

To illustrate the effectiveness of zk-SNARKs in reducing blockchain bloat, we compare it with other approaches such as Segregated Witness (SegWit) and Sharding in Table 3.

Table 3: zk-SNARKs Comparison with other Approaches

Parameter	**zk-SNARKs**	**Segregated Witness (SegWit)**	**Sharding**
Data Storage	Minimal on-chain storage	Reduces data size by separating signatures	Divide blockchain into smaller, manageable pieces
Scalability	High, due to reduced data size	Moderate, improves block utilization	High, increases transaction throughput
Transaction Throughput	Increased	Slightly increased	Significantly increased
Complexity	High (requires sophisticated cryptography)	Moderate (requires protocol changes)	High (requires substantial protocol changes)
Security	High (strong cryptographic guarantees)	High (retains main chain security)	High (depends on implementation)
Adoption	Emerging (complex implementation)	Widely adopted	Emerging (complex and resource-intensive)

Visualization of Block Size Utilization

The following bar chart illustrates the block size utilization for each approach, showing that zk- SNARKs reduce blockchain bloat the most as shown in Figure 11.

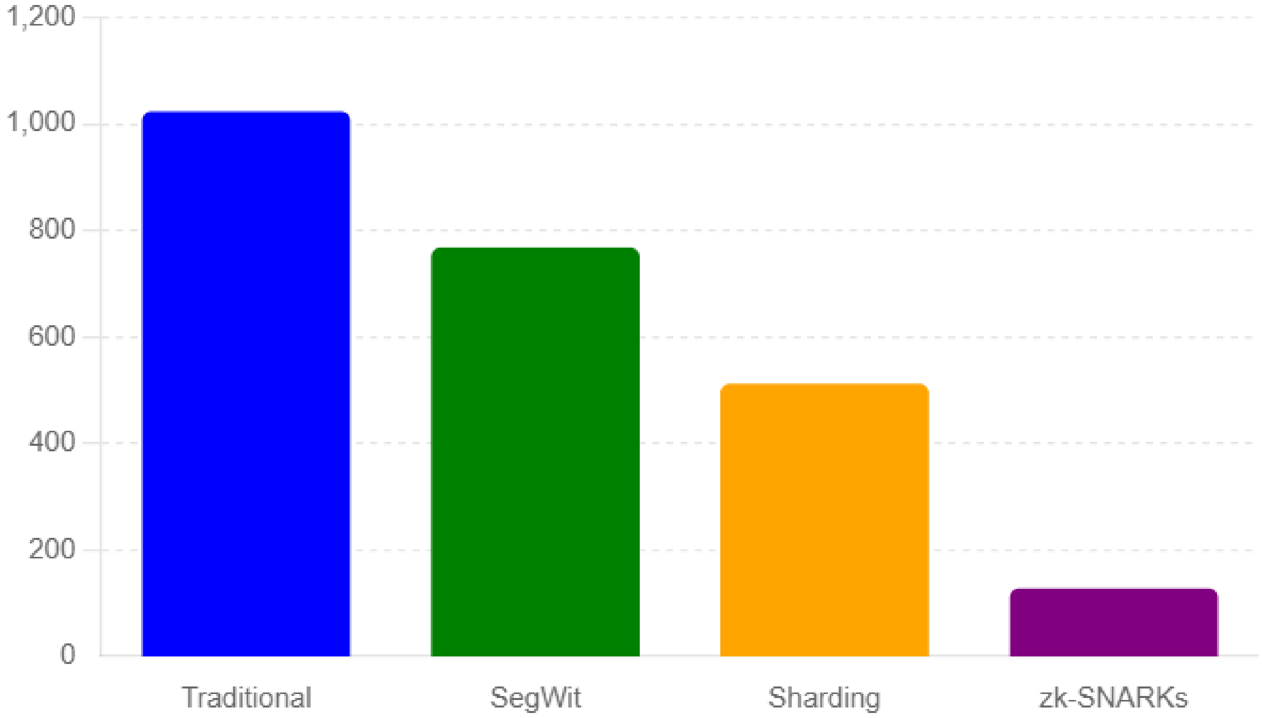


Figure 11 Block Size comparison using zk-snark and other approaches

Analysis

- **Data Storage**
  - **zk-SNARKs**: By storing only succinct proofs on-chain, zk-SNARKs minimize the data footprint, significantly reducing blockchain bloat.

  - **SegWit**: Separates transaction signatures (witness data) from the transaction data, effectively increasing the block size limit without increasing the actual block size, thereby reducing bloat to some extent.
  - **Sharding**: Distributes the blockchain ledger across multiple shards, each processing its own transactions, thereby reducing the load on any single part of the network and minimizing bloat.

- **Scalability**
  - **zk-SNARKs**: The reduced data size enhances scalability, allowing more transactions to be processed efficiently.
  - **SegWit**: Improves block utilization and throughput, but the scalability gains are moderate compared to other approaches.
  - **Sharding**: Significantly increases transaction throughput by parallelizing transaction processing across shards.

- **Transaction Throughput**
  - **zk-SNARKs**: Increases throughput by reducing the amount of data each node needs to process and store.
  - **SegWit**: Provides a slight increase in throughput by optimizing block space usage.
  - **Sharding**: Greatly enhances throughput by dividing the network's workload.

- **Complexity**
  - **zk-SNARKs**: Involves sophisticated cryptographic techniques, making implementation complex.
  - **SegWit**: Requires changes to the blockchain protocol but is less complex than zk- SNARKs.
  - **Sharding**: Complex to implement due to the need for significant changes to the blockchain architecture.
- **Security**
  - **zk-SNARKs**: Provides strong cryptographic guarantees.
  - **SegWit**: Maintains main chain security while reducing data size.
  - **Sharding**: Security depends on the implementation but generally maintains high security through parallel processing and validation.

Table 4: Block Size Utilization by Different Approaches

Approach	**Block Size (KB)**
Traditional	1024
SegWit	768
Sharding	512
zk-SNARKs	128

zk-SNARKs are the most effective in reducing blockchain bloat due to their minimal on-chain data storage requirement as shown by the block size values in table 4. By incorporating zk-SNARKs, the metaverse can achieve enhanced scalability, higher transaction throughput, and maintain robust security, all while minimizing the storage requirements and computational overhead on the blockchain.

5.3 Discussion

The results and analysis of our zk-SNARK-based security framework for the metaverse demonstrate significant improvements in key performance metrics, including privacy preservation, data minimization, verification efficiency, transaction integrity, and user trust. By integrating zk-SNARKs into blockchain-based systems, we achieved a robust solution that addresses the critical issue of blockchain bloat while enhancing scalability and performance. The succinct nature of zk-SNARK proofs allows for efficient off-chain computation and minimal on-chain data storage, leading to faster transaction speeds and reduced latency. This is crucial for the metaverse, where real-time interactions and transactions are essential for a seamless user experience.

Moreover, the strong cryptographic guarantees provided by zk-SNARKs ensure high levels of security and data integrity, fostering greater user trust in the system. The comparison with other approaches, such as Segregated Witness (SegWit) and Sharding, further highlights the superiority of zk-SNARKs in maintaining a leaner blockchain ledger while delivering higher transaction throughput and enhanced privacy. However, it is important to note that the implementation of zk-SNARKs involves complex cryptographic computations, which require careful optimization to ensure efficiency.

Future research should focus on optimizing these computations and exploring hybrid solutions that combine the strengths of different approaches to further enhance blockchain performance and scalability.

5.4 Impact

Our research on integrating zk-SNARKs into the metaverse smart contracts holds significant implications for national needs, particularly in areas of cybersecurity, digital economy growth, and technological advancement.

- **Cybersecurity Enhancement:** By leveraging zk-SNARKs, our research enhances the privacy and security of digital transactions. This is crucial for protecting sensitive data across various sectors, including government agencies, financial institutions, and the military. Implementing robust cryptographic techniques like zk-SNARKs can help safeguard national infrastructure against cyber threats and unauthorized access.

- **Economic Growth:** The implementation of zk-SNARKs in the metaverse supports the development of a secure and efficient digital economy. As businesses and consumers become more confident in the security of their transactions, the adoption of digital services is likely to increase. This can drive economic growth by fostering innovation, improving efficiency, and expanding market opportunities in the digital sector.

- **Technological Advancement:** Our research positions the nation at the forefront of blockchain and cryptographic technology. By pioneering the use of zk-SNARKs in blockchain applications, we set a benchmark for technological excellence and innovation. This can attract investment, talent, and collaboration in the tech industry, further advancing our technological capabilities and global competitiveness.

- **Compliance with Privacy Regulations:** The enhanced privacy features provided by zk- SNARKs ensure compliance with stringent data protection regulations. This is essential for maintaining the trust of citizens and international partners, promoting a secure and privacy- respecting digital environment.
- **Encouraging Digital Adoption:** By providing a secure and efficient framework for digital interactions, our research encourages the adoption of digital technologies across various sectors. This can lead to improved service delivery, increased productivity, and greater inclusion in the digital economy.

5.5 Summary

In this chapter, we discuss the integration of zk-SNARKs into the metaverse smart contracts and evaluate their impact on privacy preservation, data minimization, verification efficiency, transaction integrity, and user trust. Our results demonstrate significant improvements in transaction speed and scalability while maintaining robust security. The analysis confirms that zk-SNARKs provide a superior solution to traditional methods in reducing blockchain bloat and enhancing overall system performance, making them ideal for secure and efficient metaverse applications.

CHAPTER 6: Conclusion and Future Horizon

6.1 Overview

The conclusion and future horizon section of this research encapsulates the key findings and contributions of our study on integrating zk-SNARKs into the metaverse smart contracts. It highlights the significant advancements made in addressing privacy, security, and scalability challenges in blockchain technology. The enhanced performance metrics achieved through zk-SNARK implementation underscore its potential as a robust solution for secure and efficient digital interactions within the metaverse. Additionally, this section outlines potential areas for future research, emphasizing the need for further optimization, broader adoption of cryptographic techniques, and exploration of hybrid solutions to continue advancing the state of blockchain technology. Through this research, we set a foundation for future innovations that can leverage the strengths of zk-SNARKs to build more secure, scalable, and user-friendly digital ecosystems.

6.2 Threat to Validity

While our research demonstrates the significant benefits of integrating zk-SNARKs into metaverse smart contracts, several threats to validity must be acknowledged to provide a comprehensive understanding of the limitations and potential challenges.

6.2.1 Implementation Complexity

The implementation of zk-SNARKs involves sophisticated cryptographic computations that require specialized knowledge and expertise. This complexity can pose a barrier to widespread adoption and integration, as developers may encounter difficulties in correctly implementing and optimizing zk-SNARKs within their systems.

6.2.2 Performance Overhead

Although zk-SNARKs enhance privacy and scalability, the initial setup phase, which includes generating proving and verification keys, can be computationally intensive and time-consuming. This overhead might impact the overall performance, especially in environments with limited computational resources.

6.2.3 Scalability Challenges

While zk-SNARKs reduce on-chain data storage and enhance scalability, the proving process remains computationally demanding. In scenarios with a high volume of transactions, the scalability benefits may be mitigated by the computational requirements of generating proofs, potentially leading to bottlenecks.

6.2.4 Security Assumptions

The security of zk-SNARKs relies on specific cryptographic assumptions, such as the hardness of certain mathematical problems. Advances in quantum computing or breakthroughs in cryptographic research could potentially undermine these assumptions, posing a risk to the long-term security of zk-SNARK-based systems.

6.2.5 User Adoption and Trust

For zk-SNARKs to be effective, users must trust the privacy guarantees provided by the technology. Any perceived vulnerabilities or misunderstandings about how zk-SNARKs protect user data could hinder adoption. Building and maintaining user trust is crucial for the successful implementation of zk-SNARKs in real-world applications.

6.2.6 Regulatory and Compliance Issues

The deployment of zk-SNARKs in various jurisdictions may encounter regulatory and compliance challenges. Privacy-preserving technologies must align with local and international data protection laws, and any discrepancies could limit their applicability or necessitate additional measures to ensure compliance.

6.3 Limitations, Challenges, and Their Mitigations

6.3.1 Limitations

- **Description:** Implementing zk-SNARKs requires specialized cryptographic knowledge and expertise, making it challenging for developers unfamiliar with these concepts.

- **Mitigation:** To address this, we recommend providing comprehensive documentation, tutorials, and developer tools that simplify the implementation process. Collaborating with cryptographic experts can also facilitate smoother integration.

6.3.2 Initial Setup Overhead:

- **Description:** The initial setup phase for zk-SNARKs, including generating proving and verification keys, is computationally intensive and time-consuming.

- **Mitigation:** This overhead can be mitigated by optimizing the setup process and distributing the computational load across multiple nodes. Additionally, advances in hardware acceleration for cryptographic computations can reduce the setup time.

6.3.3 Computational Demands:

- **Description:** The proving process in zk-SNARKs is computationally demanding, which could lead to performance bottlenecks in high-transaction environments.
- **Mitigation:** Employing more efficient proof generation algorithms and leveraging hardware accelerators like GPUs can alleviate the computational burden. Future research should focus on optimizing zk-SNARK algorithms to reduce computational requirements.

6.3.4 Challenges

6.3.4.1 Security Assumptions:

- **Description:** The security of zk-SNARKs is based on specific cryptographic assumptions, such as the hardness of certain mathematical problems, which could be challenged by future advances in quantum computing.

- **Mitigation:** To mitigate this risk, ongoing research into post-quantum cryptography is essential. Developing zk-SNARKs that are resilient to quantum attacks will help ensure long-term security.

6.3.4.2 User Adoption and Trust:

- **Description:** Users may be hesitant to trust zk-SNARKs due to a lack of understanding or perceived vulnerabilities.
- **Mitigation:** Building user trust can be achieved through transparent communication about the security benefits and privacy guarantees of zk-SNARKs. Regular security audits and demonstrating real-world applications that highlight the effectiveness of zk- SNARKs can also foster confidence.

6.3.4.3 Regulatory Compliance:

- **Description:** Deploying zk-SNARKs in different jurisdictions may encounter regulatory challenges related to data protection and privacy laws.
- **Mitigation:** Ensuring compliance with local and international regulations is crucial.

Engaging with legal experts and regulatory bodies during the development phase can help align zk-SNARK implementations with legal requirements. Additionally, designing flexible frameworks that can be adapted to comply with various regulatory environments will facilitate broader adoption.

6.4 Future Work

The integration of zk-SNARKs into metaverse smart contracts has demonstrated significant potential in enhancing privacy, security, and scalability. However, there are several avenues for future research and development to further advance this technology and address the challenges identified in our study.

6.4.1 Optimization of zk-SNARKs

Future research should focus on optimizing zk-SNARK algorithms to reduce computational overhead and improve efficiency. This includes exploring new cryptographic techniques and protocols that can streamline the proving and verification processes. Additionally, developing more efficient hardware accelerators, such as GPUs and FPGAs, specifically designed for zk-SNARK computations can significantly enhance performance.

6.4.2 Post-Quantum Security

With the advent of quantum computing, it is crucial to ensure that zk-SNARKs remain secure against quantum attacks. Future work should explore the development of post-quantum zk- SNARKs that can resist the computational power of quantum computers. This involves researching new cryptographic primitives and protocols that are resilient to quantum threats.

6.4.3 Scalability Improvements

While zk-SNARKs help reduce blockchain bloat, further scalability improvements are needed to handle the growing volume of transactions in the metaverse. Future research should investigate hybrid solutions that combine zk-SNARKs with other scalability techniques, such as Layer 2 solutions and sharding. These hybrid approaches can leverage the strengths of multiple technologies to achieve even greater scalability and efficiency.

6.4.4 Enhanced Privacy Features

Expanding the privacy capabilities of zk-SNARKs beyond age verification to other sensitive user attributes and transactions can provide a more comprehensive privacy-preserving framework.

Future work can explore the integration of zk-SNARKs with advanced privacy protocols, such as zero-knowledge proofs of identity and confidential asset transfers, to enhance user privacy further.

6.4.5 User Experience and Adoption

To facilitate broader adoption, it is essential to improve the user experience and ease of integrating zk-SNARKs into blockchain applications. Future research should focus on developing user-friendly tools, SDKs, and APIs that simplify the implementation process for developers. Additionally, conducting user studies to understand the barriers to adoption and addressing them through design improvements can foster wider acceptance.

6.4.6 Regulatory Compliance and Standards

Ensuring that zk-SNARK implementations comply with global data protection and privacy regulations is crucial for their widespread adoption. Future work should involve collaborating with regulatory bodies to develop standards and best practices for zk-SNARK integration. This includes creating frameworks that can adapt to different regulatory environments and ensuring transparent and compliant use of zk-SNARKs in various jurisdictions.

6.4.7 Real-World Applications

Expanding the application of zk-SNARKs to real-world use cases beyond the metaverse can demonstrate their versatility and effectiveness. Future research can explore their integration into sectors such as finance, healthcare, supply chain, and government services. Pilot projects and case studies showcasing the benefits of zk-SNARKs in enhancing security and privacy can drive further innovation and adoption.

References

[1] C. Chen *et al.*, “IEEE INTERNET OF THINGS JOURNAL 1 Privacy Computing Meets Metaverse: Necessity, Taxonomy and Challenges.” Accessed: Aug. 03, 2023. [Online]. Available: https://arxiv.org/pdf/2304.11643.pdf

[2] T. Huynh-The *et al.*, “Blockchain for the metaverse: A Review,” *Future Generation Computer Systems*, vol. 143, pp. 401–419, Jun. 2023, doi: https://doi.org/10.1016/j.future.2023.02.008.

[3] K. Al-Towhi, “The ideal use of NFT in Metaverse -A Systematic literature review.” Accessed: Aug. 03, 2023. [Online]. Available: https://su.diva-portal.org/smash/get/diva2:1784463/FULLTEXT01.pdf

[4] S. Far and A. Rad, “Journal of Metaverse Review Article Applying Digital Twins in Metaverse: User Interface, Security and Privacy Challenges.” Available: https://dergipark.org.tr/en/download/article- file/2248394

[5] D. Di FrancescoMaesa, A. Lisi, P. Mori, L. Ricci, and G. Boschi, “Self sovereign and blockchain based access control: Supporting attributes privacy with zero knowledge,” *Journal of Network and Computer Applications*, p. 103577, Jan. 2023, doi: https://doi.org/10.1016/j.jnca.2022.103577.

[6] Journal Of L A Tex Class and Files, “C Metaverse: Security and Privacy Concerns,” vol. 14, no. 8, 2021, Available: https://arxiv.org/ftp/arxiv/papers/2203/2203.03854.pdf

[7] A. Gupta, H. U. Khan, S. Nazir, M. Shafiq, and M. Shabaz, “Metaverse Security: Issues, Challenges and a Viable ZTA Model,” *Electronics*, vol. 12, no. 2, p. 391, Jan. 2023, doi: https://doi.org/10.3390/electronics12020391.

[8] Y. Huang, Y. J. Li, and Z. Cai, “Security and Privacy in Metaverse: A Comprehensive Survey,” *Big Data Mining and Analytics*, vol. 6, no. 2, pp. 234–247, Jun. 2023, doi: https://doi.org/10.26599/bdma.2022.9020047.

[9] S. Liu, H. Zou, X. Zhao, C. Wang, and Y. Fan, “Preface: Security and Safety in the ‘Metaverse,’” *Security and Safety*, May 2023, doi: https://doi.org/10.1051/sands/2023014.

[10] L. Yang, “Recommendations for metaverse governance based on technical standards,” *Humanities and Social Sciences Communications*, vol. 10, no. 1, pp. 1–10, May 2023, doi: https://doi.org/10.1057/s41599-023-01750-7.

[11] Y. K. Dwivedi, “Metaverse beyond the hype: Multidisciplinary Perspectives on Emerging challenges, opportunities, and Agenda for research, Practice and Policy,” *International Journal of Information Management*, vol. 66, no. 66, p. 102542, Oct. 2022, Available: https://www.sciencedirect.com/science/article/pii/S0268401222000767

[12] D. Das, P. Bose, N. Ruaro, C. Kruegel, and G. Vigna, “Understanding Security Issues in the NFT Ecosystem,” 2022. Accessed: Aug. 03, 2023. [Online]. Available: https://sites.cs.ucsb.edu/~chris/research/doc/ccs22_nftsec.pdf

[13] S. Wang and W. Wang, “A review of the application of digital identity in the Metaverse,” *Security and Safety*, vol. 2, p. 2023009, 2023, doi: https://doi.org/10.1051/sands/2023009.

[14] J. Hutson, G. Banerjee, N. Kshetri, K. Odenwald, and J. Ratican, “Architecting the Metaverse: Blockchain and the Financial and Legal Regulatory Challenges of Virtual Real Estate,” *Journal of Intelligent Learning Systems and Applications*, vol. 15, no. 01, pp. 1–23, 2023, doi: https://doi.org/10.4236/jilsa.2023.151001.

[15] M. Babel and J. Sedlmeir, “Bringing data minimization to digital wallets at scale with general- purpose zero-knowledge proofs,” *arXiv.org*, Jan. 02, 2023. https://arxiv.org/abs/2301.00823#:~:text=2%20Jan%202023%5D- (accessed Aug. 03, 2023).

[16] A. Garoffolo, D. Kaidalov, and R. Oliynykov, “Zendoo: a zk-SNARK Verifiable Cross-Chain Transfer Protocol Enabling Decoupled and Decentralized Sidechains,” *IEEE Xplore*, Nov. 01, 2020. https://ieeexplore.ieee.org/document/9355752/

[17] E. B. Sasson *et al.*, “Zerocash: Decentralized Anonymous Payments from Bitcoin,” *2014 IEEE Symposium on Security and Privacy*, May 2014, doi: https://doi.org/10.1109/sp.2014.36.

[18] R. Gennaro, M. Minelli, Anca Nitulescu, and M. Orrù, “Lattice-Based zk-SNARKs from Square Span Programs,” *HAL (Le Centre pour la Communication Scientifique Directe)*, Oct. 2018, doi: https://doi.org/10.1145/3243734.3243845.

[19] S. Lee, H. Ko, J. Kim, and H. Oh, “vCNN: Verifiable Convolutional Neural Network Based on zk- SNARKs,” *IEEE Transactions on Dependable and Secure Computing*, pp. 1–17, Jan. 2024, doi: https://doi.org/10.1109/tdsc.2023.3348760.

[20] J. Kim, J. Lee, and H. Oh, “Simulation-Extractable zk-SNARK With a Single Verification,” *IEEE Access,* vol. 8, pp. 156569–156581, 2020, doi: https://doi.org/10.1109/access.2020.3019980.

[21] Y. Bespalov, A. Garoffolo, L. Kovalchuk, H. Nelasa, and R. Oliynykov, “Probability Models of Distributed Proof Generation for zk-SNARK-Based Blockchains,” *Mathematics*, vol. 9, no. 23, p. 3016, Nov. 2021, doi: https://doi.org/10.3390/math9233016.

[22] Z. Guan, Z. Wan, Y. Yang, Y. Zhou, and B. Huang, “BlockMaze: An Efficient Privacy-Preserving Account-Model Blockchain Based on zk-SNARKs,” *IEEE Transactions on Dependable and Secure Computing*, pp. 1–1, 2020, doi: https://doi.org/10.1109/tdsc.2020.3025129.

[23] D. A. Luong and J. H. Park, “Privacy-Preserving Blockchain-Based Healthcare System for IoT Devices Using zk-SNARK,” *IEEE Access*, vol. 10, pp. 55739–55752, 2022, doi: https://doi.org/10.1109/access.2022.3177211.

[24] K. Singh and G. Pavithra, “Blockchain-based Criminal Smart Contract for Symmetric Key Selling using ZK-SNARKs,” *International journal of blockchains and cryptocurrencies*, vol. 4, no. 1, Jan. 2023, doi: https://doi.org/10.1504/ijbc.2023.10060369.

[25] J. B. Bernabe, J. L. Canovas, J. L. Hernandez-Ramos, R. T. Moreno, and A. Skarmeta, “Privacy- preserving solutions for Blockchain: review and challenges,” *IEEE Access*, vol. 7, pp. 1–1, 2019, doi: https://doi.org/10.1109/access.2019.2950872.

[26] Y. Piao, K. Ye, and X. Cui, “A Data Sharing Scheme for GDPR-Compliance Based on Consortium Blockchain,” *Future Internet*, vol. 13, no. 8, p. 217, Aug. 2021, doi: https://doi.org/10.3390/fi13080217.

[27] X. Li, H. Zhao, and W. Deng, “BFOD: Blockchain-based Privacy Protection and Security Sharing Scheme of Flight Operation Data,” *IEEE Internet of Things Journal*, pp. 1–1, Jan. 2023, doi: https://doi.org/10.1109/jiot.2023.3296460.

[28] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, “An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends,” *2017 IEEE International Congress on Big Data (BigData Congress)*, Jun. 2017.

[29] I. Santoso and Yuli Christyono, “Zk-SNARKs As A Cryptographic Solution For Data Privacy And Security In The Digital Era,” *International Journal of Mechanical*

Computational and Manufacturing Research, vol. 12, no. 2, pp. 53–58, Aug. 2023, doi: https://doi.org/10.35335/computational.v12i2.122.

[30] Anatoly Konkin and Sergey Zapechnikov, "Zero knowledge proof and ZK-SNARK for private blockchains," *Journal of Computer Virology and Hacking Techniques*, vol. 19, no. 3, pp. 443–449, Mar. 2023, doi: https://doi.org/10.1007/s11416-023-00466-1.

[31] Y. Gong, Y. Jin, Y. Li, Z. Liu, and Z. Zhu, "Analysis and comparison of the main zero-knowledge proof scheme," *IEEE Xplore*, Jan. 01, 2022. https://ieeexplore.ieee.org/abstract/document/9758531 (accessed Oct. 17, 2022).

[32] T. Wang, H. Shen, J. Chen, F. Chen, Q. Wu, and D. Xie, "A hybrid blockchain-based identity authentication scheme for Mobile Crowd Sensing," *Future Generation Computer Systems*, vol. 143, pp. 40–50, Jun. 2023, doi: https://doi.org/10.1016/j.future.2023.01.013.

[33] X. Hu *et al.*, "Verifying Privacy-Preserving Financing Orders on a Consortium Blockchain Based on zk-SNARKs," *2022 IEEE Wireless Communications and Networking Conference (WCNC)*, Apr. 2022, doi: https://doi.org/10.1109/wcnc51071.2022.9771773.

[34] Z. Wan, Z. Guan, Y. Zhou, and K. Ren, "zk-AuthFeed: How to Feed Authenticated Data into Smart Contract with Zero Knowledge," *IEEE Xplore*, Jul. 01, 2019. https://ieeexplore.ieee.org/document/8946213 (accessed May 04, 2022).

[35] A. Averin, A. Samartsev, and N. Sachenko, "Review of Methods for Ensuring Anonymity and De- Anonymization in Blockchain," *IEEE Xplore*, Sep. 01, 2020. https://ieeexplore.ieee.org/abstract/document/9322974/ (accessed Jul. 15, 2022).

[36] L. Qi-nan and Z. Xue, "A Privacy-Protecting Authorization System Based on Blockchain and zk- SNARK," Dec. 2020, doi: https://doi.org/10.1145/3444370.3444610.

[37] Z. Wan, Y. Zhou, and K. Ren, "zk-AuthFeed: Protecting Data Feed to Smart Contracts with Authenticated Zero Knowledge Proof," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–1, 2022, doi: https://doi.org/10.1109/TDSC.2022.3153084.

[38] J. Park, H. Kim, G. Kim, and J. Ryou, "Smart Contract Data Feed Framework for Privacy- Preserving Oracle System on Blockchain," *Computers*, vol. 10, no. 1, p. 7, Dec. 2020, doi: https://doi.org/10.3390/computers10010007.

[39] J. Lee, J. Kim, and H. Oh, “Forward-Secure Multi-User Aggregate Signatures Based on zk- SNARKs,” *IEEE Access*, vol. 9, pp. 97705–97717, 2021, doi: https://doi.org/10.1109/access.2021.3093925.

[40] A. E. Kosba, D. Papadopoulos, Charalampos Papamanthou, and D. Song, “{MIRAGE}: Succinct Arguments for Randomized Algorithms with Applications to Universal {zk-SNARKs},” *USENIX Security Symposium*, pp. 2129–2146, Jan. 2020.

[41] E. Ben-Sasson, A. Chiesa, E. Tromer, and M. Virza, “Scalable Zero Knowledge Via Cycles of Elliptic Curves,” *Algorithmica*, vol. 79, no. 4, pp. 1102–1160, Oct. 2016, doi: https://doi.org/10.1007/s00453-016-0221-0.

[42] Y. Sun, L. Cai, W. Wang, X. Song, and Z. Lu, *Blockchain Technology and Application*. Springer Nature, 2022.

[43] M. C. Lacity and S. C. Lupien, *Blockchain Fundamentals for Web 3.0*. University of Arkansas Press, 2022.

[44] Johannes Buchmann, Abderrahmane Nitaj, Tajjeeddine Rachidi, and Springerlink (Online Service, *Progress in Cryptology - AFRICACRYPT 2019 : 11th International Conference on Cryptology in Africa, Rabat, Morocco, July 9-11, 2019, Proceedings*. Cham: Springer International Publishing, 2019.

[45] K. Huang, Y. Mu, Fatemeh Rezaeibagha, and X. Zhang, *Design and Analysis of Cryptographic Algorithms in Blockchain*. CRC Press, 2021.