CHAPTER I
INTRODUCTION

1. 1. Background

In the increasingly advanced digital era, information security has become a very important aspect, especially in computer networks and distributed systems that are often used as the main means of exchanging information. In the midst of these developments, threats to data security are also increasing, so that efforts to protect information from unauthorized access or manipulation are a major concern. In this context, encryption and hash algorithms play an important role because they are used to protect data integrity and authentication, prevent forgery, and ensure that data sent on the network cannot be read by unauthorized parties.

Hash functions in cryptography have many variants and each has its own advantages and disadvantages. The level of effectiveness of the algorithm used depends on the case to be solved. The more difficult and long the process to produce a hash value, the longer it will take, but the level of security is also high [1]. The hash algorithm specifically functions to convert data into a unique, fixed-length character string, called a hash value. This algorithm is widely used in network security, including for password encryption, file integrity verification, and digital authentication. In the context of cybersecurity, the use of a strong hash algorithm ensures that any change in the data, no matter how small, will result in a significant change in the hash value. This makes it difficult for unauthorized parties to manipulate data or create collision attacks, which is a situation where two different data produce the same hash value. In Secure Hash Algorithm (SHA) there are several variants, namely SHA-1, SHA-512, SHA-768, and SHA-512. The numbers 256, 385 and 512 are the length of the hash value generated. In this study the author uses SHA-512 which will be used to search for fingerprints or identities from docx files so that duplicate docx files can be deleted one by one and will ultimately optimize the use of storage space [1].

One of the hash algorithms that has long been relied on in network security is the Secure Hash Algorithm 2 (SHA-2), especially the SHA-512 variant. SHA-512 is an algorithm designed by the National Security Agency (NSA) and issued by the National Institute of Standards and Technology (NIST) as part of the United States

national security standard. SHA-512 was developed to replace SHA-1 which was considered no longer secure because it was vulnerable to collision attacks. This algorithm has been proven to provide higher security and is used in various applications, including secure communication protocols such as Transport Layer Security (TLS), Bitcoin, and other digital verification systems. With an output length of 256 bits, SHA-512 is capable of producing hashes that are very difficult to guess, making it one of the algorithms relied on in critical data security scenarios. However, as technology advances, the need for more efficient, faster, and resource-efficient hash algorithms is increasing. This is mainly due to changes in the technological landscape driven by increasingly sophisticated hardware and wider networks. In this context, new algorithms such as BLAKE3 emerge as promising alternatives. BLAKE3 is the successor to the BLAKE2 algorithm, developed to offer high speed without sacrificing security. BLAKE3 is designed to work optimally on modern hardware such as multicore CPUs, GPUs, and ARM devices, which are widely used in smartphones and IoT devices. This algorithm promises much higher speeds than SHA-512, enabling real-time applications in data security that require fast response and high efficiency.

While SHA-512 has long been regarded as a robust cryptographic standard, its limitations become increasingly evident in the context of modern decentralized applications (DApps), particularly those requiring high throughput and low latency. The algorithm's high computational overhead leads to slower hashing speeds, greater energy consumption, and increased strain on system resources such as CPU and memory. These characteristics are especially problematic in real-time blockchain environments where fast, lightweight, and efficient operations are critical. In large-scale or latency-sensitive systems, the use of SHA-512 may lead to performance bottlenecks that hinder the scalability and responsiveness of DApps. Moreover, as blockchain technology evolves toward more efficient and user-responsive platforms, relying solely on SHA-512 risks creating rigid infrastructures that are misaligned with current computational and regulatory demands. Thus, while SHA-512 offers strong security guarantees, its operational inefficiencies make it less suitable as a stand-alone hashing solution for next-generation DApps, highlighting the urgent need for a more adaptive, hybrid cryptographic approach.

Research on BLAKE3 is relatively new, and is still limited in practical applications. However, the performance claims offered by its developers suggest that BLAKE3 could be an effective algorithm choice to replace or complement existing conventional hash algorithms. BLAKE3 is optimized to address the performance needs of modern devices, with claims that this algorithm is able to provide higher hashing speeds than SHA-512, especially in data processing that requires intensive resources. Therefore, comparing the performance of BLAKE3 with SHA-512 in the context of network security is relevant to determine whether this algorithm is able to replace the role of SHA-512 in certain scenarios. Previous studies in the field of network security have more often focused on the analysis and testing of algorithms such as SHA-512, but have not included the same testing of BLAKE3. Previous studies that examine hash algorithms often focus on aspects of speed and efficiency in data security scenarios. For example, a comparative study involving the BLAKE2 and ChaCha20 algorithms showed that ChaCha20, which is commonly used for stream encryption, has a higher speed in the encryption and decryption process of large data. The study involved collecting data from encryption and decryption tests using various file types with varying sizes, such as documents, audio, and video. In this test, encryption and decryption speed, CPU usage, memory usage, energy efficiency, and security level were the main parameters analyzed.

The results of this study show that ChaCha20 excels in processing speed on big data and supports more secure and stable network connections. However, BLAKE2 has an advantage in data hashing for high-level security, where data integrity is a priority. Efficient use of resources is also an important concern in this study, because efficient data security can reduce system load and increase network responsiveness.

BLAKE2's superiority in data hashing that is resistant to attacks is the basis for the development of its successor algorithm, BLAKE3, which overcomes BLAKE2's performance weaknesses in speed without sacrificing its security level. This algorithm is designed to be compatible with modern architectures that allow parallel processing, making it more efficient in resource usage than conventional algorithms. In the context of network security, comparative testing between BLAKE3 and SHA-512 has

important value because the more devices connected to the network, the higher the need for fast, secure, and resource-efficient algorithms.

In addition to speed, one important aspect that distinguishes BLAKE3 from SHA-512 is the algorithm's efficiency in utilizing CPU and memory, especially on modern hardware. BLAKE3 is designed to perform hashing with fewer CPU cycles and lower energy consumption compared to previous algorithms. This advantage is very relevant in modern network scenarios that use low-power devices, such as IoT devices, which have limited power and processing capabilities. With this capability, BLAKE3 has great potential in optimizing encryption performance in real-time applications that require fast response and high accuracy.

This study aims to conduct a comparative analysis of the performance of the SHA-512 and BLAKE3 algorithms in the process of encrypting and decrypting data in network security systems. This analysis includes testing several important parameters such as encryption and decryption speed, CPU and memory resource usage, and energy efficiency. By comparing these two algorithms, it is hoped that information can be obtained regarding which algorithm is more efficient and secure for use in network security systems. In addition, this study will also examine the potential for implementing BLAKE3 in applications that require high speed and energy efficiency, especially on modern devices connected to the network.

Hash algorithm performance testing involves several key metrics that affect the quality of the algorithm in securing data on the network. First, encryption and decryption speed determine how quickly an algorithm can process data before and after it is sent over the network. This metric is very important in applications that require real-time responses, such as streaming services or financial applications that require secure transactions. Second, CPU and memory usage indicate how efficiently an algorithm utilizes hardware resources. An efficient algorithm can reduce system load, allowing users to run other applications simultaneously without reducing overall performance. Third, energy efficiency is a crucial factor, especially in low-power devices with limited battery power such as smartphones and IoT devices. Energy-efficient algorithms can extend the life of devices and reduce operational costs, making them ideal for long-term use in large-scale networks. In this test, the study will compare the measurement results of SHA-512 and BLAKE3, to evaluate which

algorithm best suits the needs of modern networks. SHA-512, which has a good reputation in terms of security, will be evaluated based on its performance in processing data on networks that require high security. On the other hand, BLAKE3 as a new algorithm optimized for performance on modern hardware will be tested to determine whether its speed claim is indeed better than SHA-512, especially in large data processing. Thus, this study will identify the advantages and disadvantages of each algorithm in network security scenarios. Based on the explanation above, this study is very relevant in an effort to fill the research gap regarding BLAKE3 performance in network security, which currently has not been tested comprehensively. The results of this study are expected to provide a significant contribution to the development of network security, by providing alternative algorithms that are more efficient in data management and system security.

Furthermore, this study not only focuses on the technical aspects of hashing but also aims to explore the broader implications of adopting new hash algorithms like BLAKE3 in diverse network environments. With the increasing complexity of modern digital systems, network security protocols must adapt to the rising demands for both efficiency and resilience. BLAKE3, which is optimized for multicore processing, may offer distinct advantages in systems that require rapid data handling, such as cloud-based services or IoT networks with thousands of connected devices.

In addition, this research examines the potential trade-offs between security robustness and operational efficiency. While SHA-512 provides a proven level of security against various forms of cryptographic attacks, BLAKE3's faster processing speed could benefit environments where lower latency is crucial, such as real-time financial transactions and live data streams. This comparison will highlight whether the slight reduction in security, if any, with BLAKE3 is offset by the significant gains in performance, thereby providing actionable insights into which algorithm is preferable for different types of security infrastructures.

However, complying with data privacy regulations is not without challenges. The cost of implementing the necessary security technologies can be significant, and companies also need to invest in employee training and regularly update their privacy policies. Furthermore, businesses must adjust their processes to meet regulatory requirements, which can be a complex and time-consuming endeavor. Nevertheless,

the long-term benefits of adhering to data privacy regulations cannot be overlooked. By enhancing data security, companies not only protect their customers' personal information but also build trust and loyalty. Ultimately, this can improve the company's reputation and provide a competitive advantage in the market [2].

In the rapidly evolving landscape of blockchain technology, the efficiency, security, and scalability of cryptographic algorithms play a pivotal role in shaping the robustness of decentralized systems. The Secure Hash Algorithm family, particularly SHA-256 and SHA-512, has long been the cornerstone of many blockchain protocols, including Bitcoin and Ethereum. However, the increasing computational demands and energy consumption associated with these algorithms have prompted researchers and practitioners to explore alternative solutions that balance performance and security.

BLAKE3, a cryptographic hash function introduced in 2020, has garnered attention for its remarkable speed, low resource consumption, and strong security guarantees. Designed as a successor to the BLAKE2 algorithm, BLAKE3 leverages a unique tree-based structure that enables parallel processing, making it highly efficient for modern hardware architectures. Unlike SHA-256 and SHA-512, which are deeply embedded in traditional blockchain systems, BLAKE3 offers the potential to redefine how hashing is utilized in blockchain by addressing critical challenges related to scalability and energy efficiency.

Despite its technical superiority in certain aspects, BLAKE3’s adoption in the blockchain ecosystem remains limited. Questions surrounding its compatibility with existing protocols, long-term security assurance, and integration costs have hindered its widespread implementation. This study aims to evaluate the applicability of BLAKE3 in blockchain environments, analyzing its performance, security, and potential benefits compared to established algorithms like SHA-256 and SHA-512.

Through this study, we hope to contribute to the development of network security by demonstrating the practical benefits and limitations of emerging hash algorithms. By thoroughly analyzing SHA-512 and BLAKE3 across various parameters, including speed, resource efficiency, and adaptability to modern hardware, this research will guide practitioners in choosing algorithms that balance security with

performance, ensuring that network systems remain both protected and efficient as technology continues to evolve.

However, both SHA-512 and BLAKE3 possess distinct advantages and limitations. SHA-512 has been proven secure under various security standards, but it requires substantial computational resources and has limitations in terms of speed. On the other hand, BLAKE3 excels in time efficiency and resource utilization, yet it has not been extensively tested in large-scale blockchain implementations. Therefore, a combination of these two algorithms—utilizing SHA-512 as the backbone for security and BLAKE3 as a performance accelerator—may serve as a viable solution to bridge the need for high security with operational efficiency in DApps implementation.

The development of Decentralized Applications (DApps) demands a robust security structure, as these applications operate without a central server and facilitate direct peer-to-peer transactions on the blockchain. Consequently, each hashing process within smart contracts must be safeguarded against cryptographic attacks such as collision, pre-image attacks, or hash manipulation. The proposed combination of SHA-512 and BLAKE3 offers an innovative algorithmic approach to address these challenges: SHA-512 ensures the robustness of the system against attacks, while BLAKE3 enhances execution speed and reduces resource consumption—particularly beneficial in IoT environments or low-power devices. Thus, this combination not only strengthens system performance but also broadens the practical and efficient implementation of blockchain technology.

1. 2. Problem Identification

The performance of the SHA-512 and BLAKE3 algorithms in terms of data encryption and decryption speed in network security systems shows significant differences. SHA-512, as a more conventional hashing algorithm, is designed with a high level of security but tends to be slower when compared to newer algorithms such as BLAKE3. BLAKE3 was developed to be more efficient and faster, so it generally has an advantage in terms of processing speed. In the context of networks, BLAKE3 is able to perform encryption and decryption in a shorter time, making it more optimal for use in applications that require high speed. In terms of resource efficiency, BLAKE3 is generally superior to SHA-512, especially in terms of CPU, memory, and

energy usage. SHA-512 has relatively high computational requirements, so it can burden the system, especially on a large scale or networks that require real-time responses. BLAKE3, on the other hand, is designed to be more efficient and resource-efficient, allowing for lower CPU and memory consumption. This efficiency makes BLAKE3 a better choice for network applications that require fast data processing without sacrificing too many resources, which ultimately supports overall network performance.

Complying with data privacy regulations is not without challenges. Inevitably, significant issues arise, such as the high costs associated with implementing security technologies. Companies also need to make substantial investments, including employee training and regular updates to privacy policies on a scheduled basis. Businesses must adapt their processes to align with regulatory requirements as they evolve over time. The primary challenges in adhering to data privacy regulations are a combination of high costs, the need for investments in training and policy updates, and the complexity of operational adjustments. All of these require a well-thought-out strategy to ensure that companies can comply with regulations without sacrificing productivity or financial stability [2].

The dominance of SHA-256 and SHA-512 in blockchain protocols has created a reliance on algorithms that, while secure, are less optimized for modern computational demands. With blockchain applications expanding into areas requiring high throughput and low latency, the need for a more efficient cryptographic solution becomes evident. BLAKE3's advanced features present an opportunity to enhance blockchain performance, yet its feasibility as a replacement or complement to existing standards remains underexplored.

The primary challenge in developing decentralized applications (DApps) lies in designing a hashing mechanism that can simultaneously balance security and efficiency. The SHA-512 algorithm has a strong reputation for security but demonstrates significant weaknesses in execution speed and resource efficiency, especially in environments that demand real-time responsiveness and high throughput. Conversely, BLAKE3 excels in performance and energy efficiency but lacks the same level of adoption and proven resilience as SHA-512 in critical blockchain systems. This raises an important question: is it feasible to combine the strengths of both

algorithms into a single hashing model that is more adaptive to the demands of modern DApps?

Moreover, there has been no practical approach that explicitly integrates both algorithms into a single smart contract as a cryptographic combination strategy. Most previous studies have only compared their performances separately, without exploring the synergistic potential of combining them. In fact, the current challenges in DApps development are not limited to fast and low-cost transaction processing, but also involve ensuring data integrity and compliance with regulations such as the GDPR and Indonesia's Government Regulation No. 71 of 2019. Therefore, this research aims to address the need for a cryptographic system that is not only secure and fast but also aligned with applicable legal standards for data protection.

By integrating SHA-512 and BLAKE3 algorithms into the hashing process of smart contracts, this study seeks to establish a new foundation for more optimized Ethereum-based DApps development. Experimental evaluations will be conducted to assess how the combination affects transaction speed, gas efficiency, CPU and memory usage, and resistance to cryptographic attacks. This approach aims not only to improve technical performance but also to contribute to the development of a relevant and applicable DApps security architecture for the future of blockchain-based industries.

Although SHA-512 is a hash algorithm that relies heavily on security, it has significant limitations in terms of speed and resource efficiency. SHA-512 is designed with a robust cryptographic architecture and has been proven to withstand various attacks, including collision and pre-image attacks. However, the complexity of its hashing process results in high CPU and memory consumption, as well as slower execution times. In a blockchain environment that increasingly demands high-speed transactions and system scalability, these limitations can be a hindrance, especially for applications that require real-time response, such as financial DApps and intensive smart contracts.

Furthermore, a study on collision attacks on reduced SHA-512 found a collision attack (two different inputs producing the same hash) on SHA-512, shortened

to 23 and 24 steps with significantly lower complexity. This indicates a potential structural weakness in SHA-512 when the number of steps is reduced [2].

Furthermore, a study on quantum collisions against SHA-512 found a quantum algorithm (Grover). With this algorithm, an attacker can effectively create collisions on SHA-512 in up to 38-39 steps, more efficiently than classical methods. Although not a full SHA-512, this indication suggests that cryptographic systems relying on SHA-512 could become vulnerable when practical quantum computing becomes available [3].

Several data breach incidents (such as those at LinkedIn, Dropbox, and Adobe) have shown that even passwords hashed with strong algorithms like SHA-1/SHA-512 can still be compromised. If the hash is not salted, it can easily be matched against a rainbow table. In 2012 and 2016, 117 million accounts were leaked with unsalted password hashes (SHA-512), and thousands of accounts could be reverse-hashed using public databases.On the other hand, BLAKE3 presents a far superior algorithm in terms of speed and energy efficiency. With its highly parallel design and efficient memory optimization, BLAKE3 is capable of significantly faster hashing than SHA-512, even on devices with limited resources. However, BLAKE3's adoption in blockchain systems is still low, and its resistance to long-term attacks has not been as comprehensive as SHA-512's. This gap creates a dilemma between choosing a proven strong but slow algorithm or a fast but unproven algorithm for critical distributed systems [4].

Given the weaknesses of each algorithm, this study proposes a hybrid approach combining SHA-512 and BLAKE3 as a more adaptive and balanced solution. The hybrid model is expected to maintain the security strengths of SHA-512 while leveraging the performance efficiency of BLAKE3, creating a hashing system that is robust to attacks yet computationally lightweight. In the context of DApp development, this combination aims not only to improve technical performance but also to address regulatory challenges by providing security mechanisms that comply with data protection provisions such as the GDPR and Government Regulation No. 71 of 2019. Thus, the hybrid approach is not only a technical solution but also a strategic step towards a faster, more secure, and more legally compliant blockchain future.

1. 3. Objective and Contributions

This research aims to develop and evaluate a hybrid cryptographic model that combines SHA-512 and BLAKE3 algorithms in the context of data security within Ethereum-based smart contracts. The main objectives of this study are as follows:

1. To analyze and compare the performance of each algorithm, as well as their combination, in the processes of data encryption and decryption within DApps environments.
2. To measure execution efficiency (execution time), gas consumption, and resource usage (CPU and memory) of the SHA-512 and BLAKE3 combination in blockchain transactions.
3. To assess the resilience of the combined algorithm against cryptographic attacks such as collision attacks and pre-image attacks in decentralized financial application scenarios.
4. To build and implement a smart contract integrating the combined algorithm using the Ethereum platform and supporting tools such as Truffle, Ganache, Web3.js, and MetaMask.
5. To evaluate the impact of this algorithmic combination on user experience, transaction cost efficiency, and system scalability in real-world financial DApps.
6. To provide a technical recommendation foundation for blockchain developers and policymakers regarding the selection of optimal and regulation-compliant hashing algorithms (e.g., GDPR and Indonesian Government Regulation No. 71 of 2019).

1. 4. Hypothesis

This study is structured around several hypotheses to be tested through experiments within an Ethereum-based blockchain environment:

1. The combination of SHA-512 and BLAKE3 algorithms improves execution time efficiency and resource usage (CPU and memory) in data encryption and decryption processes within smart contracts, compared to the use of SHA-512 alone.

2. The combined SHA-512 and BLAKE3 algorithm demonstrates equal or better resilience against cryptographic attacks (e.g., collision and pre-image attacks) compared to using either algorithm separately.
3. Integrating SHA-512 and BLAKE3 in smart contracts reduces gas consumption during blockchain transactions, making it more cost-effective for implementing financial DApps.
4. This algorithmic combination complies with efficiency and security standards required by data protection regulations, such as the GDPR and Government Regulation No. 71 of 2019, making it suitable for broader adoption in decentralized blockchain systems.

1. 5. Assumption

The research is based on several conceptual and technical assumptions:

1. The combination of SHA-512 and BLAKE3 can be functionally implemented within a single hashing system in smart contracts without logical or compatibility conflicts.
2. The Ethereum platform used in this study provides sufficient infrastructure to test algorithm performance representatively for real-world blockchain applications.
3. The testing environment using Ganache and Ethereum testnets (e.g., Rinkeby) adequately reflects the mainnet ecosystem in terms of transaction architecture and smart contract execution.
4. The test dataset (comprising English words and their hashes) is considered a valid representation of common input types in financial DApp data transactions.
5. Parameters such as execution time, gas consumption, CPU and memory usage, and resistance to attacks can be objectively and consistently measured to evaluate algorithm performance.

1. 6. Scope of Work

The scope of this research is designed to stay focused on the core issue of hashing algorithm efficiency and security in DApp development. The specific boundaries of this study include:

1. The research is limited to two cryptographic algorithms—SHA-512 and BLAKE3—and their combination, excluding other hash algorithms such as SHA-256, Keccak, or Argon2.
2. Implementation is conducted within the context of Ethereum smart contracts using the Solidity programming language, with Ganache serving as a local network simulator and Rinkeby/Testnet as the public blockchain test environment.
3. The research focuses on backend performance (smart contract and hashing mechanisms) and does not deeply evaluate frontend UI/UX aspects of DApps.
4. Performance evaluation focuses on execution time, gas usage, CPU and memory consumption, and resilience against basic cryptographic attacks (collision and pre-image), excluding asymmetric or complex encryption mechanisms.
5. The DApp developed in this study serves a prototypical and experimental purpose and is not intended for commercial release but rather to test cryptographic algorithm performance in blockchain systems.

1. 7. Research Methodology

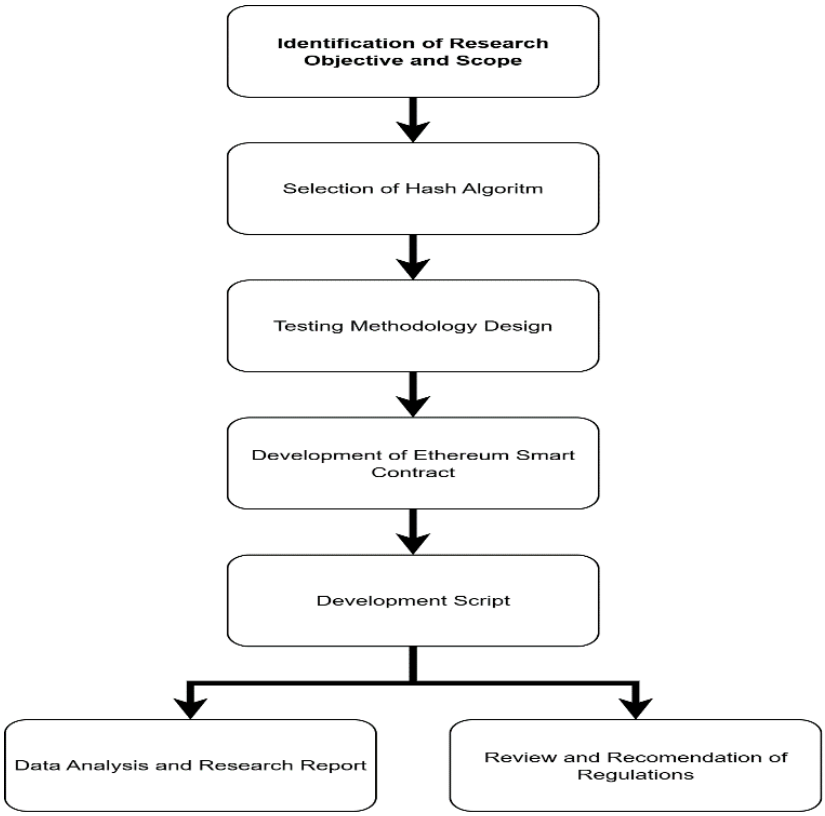

Figure 1. 1. Research Methodology

1. Identification of Research Objectives and Scope

The primary objective of this research is to evaluate the performance and security of two hashing algorithms, SHA-512 and BLAKE3, within decentralized finance applications (dApps) on the Ethereum blockchain. The scope of the research focuses on comparing the two algorithms in terms of execution speed, gas efficiency, security (resistance to attacks like collision and pre-image), and their impact on the integration of smart contracts. This study will limit itself to smart contracts written in Solidity, running on the Ethereum blockchain, using Ganache for local testing and Rinkeby or Mainnet for testnet deployment. Furthermore, the study will not consider other factors such as memory usage or scalability on a large-scale blockchain network

2. Selection of Hash Algorithms

In this research, the two hashing algorithms to be evaluated are SHA-512 and BLAKE3. These algorithms were selected due to their relevance and widespread use

in cryptographic applications. SHA-512, a member of the Secure Hash Algorithm family, is well-established in blockchain applications, including Ethereum, while BLAKE3 is a newer algorithm designed to offer faster hashing speeds without compromising security. The comparison between these two algorithms will focus on their performance, security, and integration with smart contracts in a dApp finance context.

3. Testing Methodology Design and Dataset Selection

The testing methodology will involve a controlled experimental design using blockchain transactions. To ensure accuracy and consistency, we will use Ganache to simulate a local Ethereum network, allowing us to test the execution speed, gas consumption, and transaction success rates of both algorithms. The dataset will consist of a series of mock financial transactions, where each transaction will be hashed using both SHA-512 and BLAKE3. These transactions will simulate real-world dApp finance activities, such as token transfers, loan requests, and asset tokenization. The data will be measured in terms of transaction times, gas usage, and failure rates for each hashing algorithm.

4. Development of Ethereum Smart Contract

The smart contracts will be developed in Solidity and will implement basic financial operations within the context of a dApp finance application. These contracts will interact with Ethereum-based assets and require hashing for security purposes, such as for verifying the integrity of transactions and preventing unauthorized data modifications. The smart contracts will incorporate both SHA-512 and BLAKE3 for hashing and will be deployed on Ganache for local testing and Rinkeby or Mainnet for deployment on the Ethereum testnet. These contracts will also be designed to interact with Web3.js for integration with the frontend of the dApp.

5. Development Script

To facilitate testing, a development script will be written using Truffle for deploying the smart contracts, interacting with the Ethereum network, and conducting the necessary tests. This script will automate the process of invoking smart contract functions, executing transactions, and measuring performance. It will

also collect data on execution time, gas usage, and any errors or anomalies encountered during the testing phase. MetaMask will be used to simulate the user experience by enabling interaction with the Ethereum network via a browser extension.

6. Testing

In the testing phase, both SHA-512 and BLAKE3 will be used in smart contracts to handle various blockchain transactions. The primary focus will be on evaluating the execution speed of each algorithm by measuring the time it takes to process transactions. Additionally, the gas efficiency will be assessed by tracking the amount of gas consumed for each transaction, which directly impacts transaction costs on the Ethereum network. Security testing will also be conducted to identify how resistant each algorithm is to common cryptographic attacks, such as collision and pre-image attacks, ensuring that the integrity of data within smart contracts is maintained. Furthermore, the transaction success rate will be recorded, focusing on the percentage of successful transactions for each hashing algorithm, to assess how well each handles real-world use cases in dApp finance applications.

7. Data Analysis and Research Report

The data collected from the testing phase will be thoroughly analyzed to compare the performance and security of SHA-512 and BLAKE3. First, the execution times for both algorithms will be compared to determine which algorithm is faster in processing blockchain transactions. Gas usage will be evaluated by analyzing the gas costs for each algorithm, which will help identify which one is more cost-efficient for decentralized finance applications. The transaction success rate will be calculated for each algorithm, identifying any discrepancies or failures in transaction integrity. Security analysis will also be conducted to evaluate the vulnerability of each algorithm to cryptographic attacks. Based on these findings, the research report will provide a detailed comparison, highlighting the strengths and weaknesses of each hashing algorithm. The report will conclude with recommendations on the most suitable algorithm for different types of dApp finance applications, offering insights for future optimizations and further research in this area.

8. Review and Recommendation of Regulation

This study examines the implementation and evaluation of SHA-512 and BLAKE3 hashing algorithms in decentralized financial applications (dApps) using the Ethereum blockchain, with Ganache as a local blockchain simulation. The results indicate that both algorithms possess unique characteristics influencing performance, resource efficiency, and security. SHA-512, as a well-established algorithm, offers high security levels but demands more computational resources compared to BLAKE3. On the other hand, BLAKE3, with its modern design, demonstrates faster performance and greater efficiency in resource management, making it an attractive alternative for blockchain-based systems.

Testing revealed that BLAKE3 has significant advantages in terms of hashing speed and CPU and memory resource consumption. This makes BLAKE3 more suitable for blockchain applications that require real-time processing of large amounts of data, such as in financial dApps. However, despite its efficiency, this study also emphasizes the importance of thorough security testing, such as resistance to collision and pre-image attacks, to ensure that the algorithm can be utilized without compromising data integrity.

Additionally, this study highlights the importance of regulations supporting the use of blockchain technology and efficient hashing algorithms. Existing regulations need to be updated to accommodate newer technologies like BLAKE3, especially in the financial sector. Data security standards and transaction cost efficiency are critical aspects regulators must consider to promote the wider adoption of blockchain technology. This study also recommends that regulators provide guidelines and incentives to support the development and implementation of emerging technologies.

Overall, this study offers valuable insights into the performance comparison between SHA-512 and BLAKE3 in blockchain applications. With these findings, developers and policymakers can make more informed decisions regarding which hashing algorithm to use based on specific application needs and resource constraints. This research also paves the way for further studies to test other hashing algorithms or explore implementations in more complex blockchain scenarios.