Exchanged Secure Hashing Algorithm (ESHA-512) For Blockchain Technology

S. Jenifa Sabeena[1*], S. Antelin Vijila[2]

[1]Research Scholar,Department of Computer Science and Engineering, Manonmaniam Sundaranar University,Tirunelveli, Tamil Nadu, India, Email: jenifasabeena1996@gmail.com
[2]Assistant Professor,Department of Computer Science and Engineering, Manonmaniam Sundaranar University,Tirunelveli, Tamil Nadu, India, Email : antelinvijila@gmail.com
*Corresponding Author

Received: 05.07.2024 Revised: 07.08.2024 Accepted: 13.09.2024

ABSTRACT
The key strength of blockchain technology depends on the encryption algorithm as well as hashing technique. Hashing converts the transaction data of the block to a fixed length value to represents the original data. Hashing encryption algorithms such as MD5, SHA-1, SHA-512 etc., can be applied to different aspects apart from blockchain. This paper introduces a novel Exchanged SHA-512 algorithm which is an improved version of SHA-512 with lesser number rounds as 60 where in traditional SHA-512 it is 80 rounds. This proposed ESHA-512 algorithm computes the hashing functions with different types of register as related position of the registers chosen in SHA-512. The initial hash value arrangement is also depending on the length of the message either in odd or even. Due to this enhancement the proposed ESHA-512 improves the strength as well as reduces the overall computation of the algorithm considerably than the existing SHA-512. The proposed ESHA-512 algorithm also increases encryption data size per second than the SHA-512 algorithm. The evaluation of all kinds of hashing algorithms with different number of blocks in blockchain technology is clearly stated in this paper.

Keywords: ESHA-512, blockchain, algorithm, technology.

1. INTRODUCTION

People storing their data on cloud platforms are growing in number. Outsourcing data offers clients several advantages, including decreased expenditures for personnel upkeep, hardware/software maintenance, and heavy storage management. Nevertheless, a customer may lose control over data management if they store data in the cloud, which might result in security issues. Malicious attacks, technical faults, internal attacks, or human mistake are typically the causes of data loss or corruption in cloud servers [1-3]. These factors make it necessary for cloud users to regularly conduct data integrity audits on outsourced data using effective methods. A stimulating research topic is the combination of blockchain technology and cloud computing [4, 5, 6], as well as the use of the blockchain's security mechanism to improve the cloud's capacity for computation and safe data storage.

Certain technologies, including blockchain, AI, IoT and automation, are becoming essential to an organization's transition to a cognitive enterprise. Blockchain technology is a solution to the long-standing issue of human trust. It appeared on the market with the known crypto-currency Bitcoin. Instead of putting faith in any individual operating inside a decentralised system like the Internet or Web, it offers an architecture that enables us to do so. The peer-to-peer network upon which it operates is built, and it contains identical copies of the transaction ledger. The transaction process is completed by machine consensus, which helps to eliminate the need for an intermediary.

Distributed general ledger technology, or blockchain, is mostly utilised in the cryptocurrency space, where the most well-known examples are Bitcoin and Litecoin [7], Zcash [8] and Monroe [9]. Blockchain technology is able to reliably provide data confidentiality, authenticity, and dependability despite its fast expansion. Data allocation schemes [10], personal data protection [11], and medical data [12] have also made extensive use of it. A block, the fundamental building block of a blockchain, is made up of a partition header containing the original data and a block body containing transaction data. Among these, the data from the hash value of the range block is indexed and connected to the preceding block using block data.

Hashing is another essential component of blockchains. Data is transformed into a string of characters using a cryptographic technique called hashing. Hashing allows for more efficient information storage

since it creates a fixed-size hash, just like encryption does. This study presented a novel notion based on the cryptographic concept of hashing that attempts to give ownership protection in blockchain with the goal of improving security.

In order to hash data from a block and produce an output with a certain length, one must process the data through a mathematical relation. Using a fixed-length output improves security since it prevents someone attempting to decrypt the hash from determining the length of the input by merely looking at the result. The blockchain consists of sequences of blocks that are similar to linked lists that contain data about transactions and timestamps in addition to the encrypted hash value (function) of the previous block in its own network. The hash value from the previous block is written onto the next block to form a chain of following blocks. The two blocks get connected as a result of this. In this process, which is repeated in sequential sequence, the integrity of every block up to the genesis block is tested. To determine each block's hash value from the data it contains, the SHA-256 algorithm is employed. One unique code from that block is reflected in this value. Any modification to the data within a block also affects the block's hash value. Hence, any modification to the data of a block in the past would likewise need a modification to the hash values of every subsequent block.

The length of the output generated by SHA-2 hash algorithms sets them apart. The two fundamental variations are SHA-256 and SHA-512, which are really just various word length applications of the same algorithm. While SHA-512 acts on 64-bit words, SHA-256 operates on 32-bit words. The two variations use different starting values and have differences in several constant parameters and values. With a modified beginning value and an output truncated to a predetermined number of bits, the other four versions SHA-224, SHA-384, SHA-512/224, and SHA-512/256 have the same functionality as SHA-256 (the original version) or SHA-512 (the others). Because of this, the two fundamental varieties alone may be used to describe the whole family.

Longer hash values are produced by SHA-512 and SHA-256, which is the main distinction between them. The hash value produced by SHA-512 is 512 bits, while SHA-256 produces a 256-bit hash value. A larger hash value increases the difficulty of a "brute force attack," which is a method of attempting to guess the original message. Thus, it would seem that SHA-512 provides stronger security than SHA-256 at first glance.

2. Related works

Blockchain technology is widely applied in the following fields: e-commerce, energy, social applications [13,14], copyright protection, advertising, health, insurance, logistics, and cryptocurrency (financial). Some of the recent studies based on blockchain have been presented in this section.

Digitalization has led to the development of E-commerce platforms, which have concerns including fraud, commission fees, buyer-seller payments, and unauthorised use of personal data. With smart contracts and payments, blockchain technology may provide dependability and transparency [15–16]. Blockchain technology can help prevent cyber attacks by mitigating the harm, loss, and misuse that arise from security flaws. The use of this technology can also prevent significant financial losses that cyber attacks can cause to organisations, people, or governments [17–18]. Suzen et al [19] created a blockchain-based safe storage architecture to guard against potential data leaks that might lead to the theft of credit card information from e-commerce apps.

A blockchain-based document managing system was created by Zhu et al. [20] to explicitly address the issue of simple document manipulation. Ren et al. [21] proposed the identity-based proxy aggregation signature (IBPAS) technique, which aims to reduce storage space compression and promote efficiency in signature verification while also decreasing communication bandwidth. TheDCOMB(Decentralization, Consensus, Ownership, Monetization, Blockchain) approach, a unique strategy for creating a blockchain-based query model for Internet of Things data, was described by Ren et al. [22]. This technique improves the generality and data interoperability of IoT database systems by mining hash computation to execute queries.

In response to a potential data breach, Taherdoost et al. [23] created a blockchain-based safe storage strategy to stop credit card information from being stolen in e-commerce apps. Additionally, by using this technology, significant financial losses caused by cyber attacks on organisations, people, or governments can be prevented [24]. A transaction processing system that offers safe online transactions and a model that guards against denial of service (DoS) attacks were introduced by Shaikh et al [25].

Liu et al. [26] created a scheme for supply chain management and cross-border e-commerce that uses blockchain technology to guard against fake product, clone and tag attacks as well as solve the key recovery problem. A multi-chain model, a data management model and a block structure model are the models included in the framework. Liang et al [27] proposed a secure Fabricblockchain-based data

transmission technique for industrial IoT using dynamic secret sharing mechanism and power data consensus mechanism.

In a non-trust setting, Hao et al. [28] presented a blockchain-based outsourced data integrity verification system. The key segment computation data hash is taken out of memory and applied to a high-performance hash algorithm in blockchain technology using pipelines [29]. Once the hash value is computed, and the result is wrapped and sent to the storage server to finish storing the blockchain.

During operation execution, the PRCA (Proactive Reconfigurable Computing Architecture) determines the best computing structure through self-perception and dynamic selection. Every variation of both hardware and software is dynamic. As a result, they may choose the best solutions throughout the application processing phase based on the program's independent variables to obtain variable optimum solution sets with comparable functions and varying computing efficiencies [30].When used with blockchain, it can boost the hash algorithm's security, efficiency of transmission, and overall performance. An algorithm for blockchain hashing that is based on PRCA was suggested by Fu et al. [31]. An efficient reconfigurable hash algorithm is built via a complete pipeline approach, with the goal of matching the structure of the blockchain hash algorithm.

Ensuring the integrity of the GPS location data and permitting anybody to follow them are necessary steps to increase dependability. Using the Ethereum blockchain network, a prototype system was created by Lee et al. [32] to store GPS data and the changes made throughout survey procedures.

Ranjit Kumar et al [33] proposed a unique solution called Chameleon Hashing Technique, which is a particularly regionalized peer-to-peer framework based on blockchain for privacy preservation in an e-government system. Chameleon hashing always employs trapdoor one-way hash functions in conjunction with a public key (hash key) and private key (trapdoor key) pair.

Panwar et al [34] developed a Cognitive method dubbed blockchain-based cryptographic curve hash signature (BC-CCHS) technology to safeguard patients' medical records and safely and conveniently communicate their sensitive health data. The suggested method is implemented in the hyperledger framework. Kosta et al. [35] introduced a strong and secure lightweight cryptographic hash algorithm that compresses each 512-bit of data to 256-bits. It is then separated into 8 blocks of 32 bits each. By executing an experimental setup for SHAs using FPGA (Field Programmable Gate Array) optimisation approaches, Zeyad et al. [36] proposed the benefits and drawbacks of the optimisation strategies and their effect on the performance level.

Using Hyperledger Fabric, Sharma et al. [37] created a unique blockchain architecture. They also used Hyperledger Composer to create a permissioned blockchain structure and a decentralised SWARM storage system to store multimedia data. Al-Ghuraybi et al. [38] presented a blockchain technology that emphasises the performance and security elements of CPS, especially in fending against external attacks, by combining blockchain technology with machine learning. Additionally, it looks into how blockchain technology combined with physically unclonable functions (PUF) might greatly improve the effectiveness of physical device authentication. A secure random DNA encoded key generation mechanism was used by Sanober et al. [39] to cartographical primitive for blockchain.

3. Overview of SHA-512

SHA-512 is a variant of SHA-256 which operates on eight 64-bit words which produces the message digest of 512-bit and block length is 1024 bit. Initially the message is padded to make it into 1024 bits long if required. Subsequently it is parsed into message blocks represented as $MB^{(1)}$,$MB^{(2)}$,……..,$MB^{(N)}$ of 1024 bits long. Each message blocks are processed one by one. The hashing process begin with a initially predefined hash value $Hash^{(0)}$. Further hashing values are computed as in Equation 3.1.

$$Hash^{(i)} = Hash^{(i-1)} + CF_{MB\,(i)}\left(Hash^{(i-1)}\right) \qquad (3.1)$$

Where $CF_{MB(i)}$ typically represents the Cryptographic Hash (CF) of thei-thmerkle branch. In a merkle tree or hash tree, a Merkle Branch (MB) is a set of hash values of nodes along the path from a leaf node to the root of the tree. The CF of this MB is a hash value that encapsulates the integrity of the branch, + represents the concatenation operation.

The equation 3.2 represents a cryptographic primitive, often used in cryptographic algorithms and functions. Chf represents the choice function. The symbol Λ represents logical AND, ┐ represents logical OR, ⊕ represent logical XOR operation

$$Chf(u, v, w) = (u \wedge v) \oplus (\neg u \wedge w) \qquad (3.2)$$

The Chf function takes three 64-bit words (u, v, and w) as input and performs bitwise logical operations on them to produce a 64-bit output. It is designed to ensure that the output bit is influenced by the values of the input bits in a way that adds complexity and diffusion to the hashing process.

$$Majf(u, v, w) = (u \wedge v) \oplus (u \wedge w)) \oplus (v \wedge w) \qquad (3.3)$$

Majf represent majority function, it is one of the logical functions used to process the message blocks and update the hash values in each iteration.

$\sum_0(u) = RS^{28}(u) \oplus RS^{34}(u) \oplus RS^{39}(u)$ (3.4)

$\sum_1(u) = RS^{14}(u) \oplus RS^{18}(u) \oplus RS^{41}(u)$ (3.5)

$Sigma_0(u) = RS^{1}(u) \oplus RS^{8}(u) \oplus RR^{7}(u)$ (3.6)

$Sigma_1(x) = RS^{19}(u) \oplus RS^{61}(u) \oplus RR^{6}(u)$ (3.7)

In the equation 3.4 &3.5, RS represents a right shift operation, the $RS^{28}(u)$ represents a right shift of 28 positions, $RS^{34}(u)$ represents a right shift of 34 positions and $RS^{39}(u)$ represents a right shift of 39 positions.

Both $Sigma_0(u)$ and $Sigma_1(x)$ serve similar purposes in terms of introducing non-linearity and diffusion in the equations 3.6 &3.7, notations RS and RR, which typically represent right shift and right rotate operations, they differ in the specific shift and rotation constants applied to their respective input words.

$RS^1(u)$ represents the right shift of the bits of u by 1 position, $RS^8(u)$ represents the right shift of the bits of u by 8 position, $RR^7(u)$ represents the right rotate of the bits of u by 7 positions, in a right rotate, the bits that are shifted out from the right are brought back in from the left.

The initial hash value $Hash^{(0)}$ (64-bit) is derived from the first eight prime numbers. Square root is estimated for the first eight prime numbers from the value of fractional part is used as initial hash values. In the traditional SHA-512 the following are the initial hash values.

$Hash_1^{(0)} = 6a09e667f3bcc908$

$Hash_2^{(0)} = bb67ae8584caa73b$

$Hash_3^{(0)} = 3c6ef372fe94f82b$

$Hash_4^{(0)} = a54ff53a5f1d36f1$

$Hash_5^{(0)} = 510e527fade682d1$

$Hash_6^{(0)} = 9b05688c2b3e6c1f$

$Hash_7^{(0)} = 1f83d9abfb41bd6b$

$Hash_8^{(0)} = 5be0cd19137e2179$

As already stated after the padding process, the message is parsed into N blocks with 512-bit length. Each block $MB^{(i)}$ further split into 64 bits of message which is represented as $MB_0^i, MB_1^i, \ldots, MB_{15}^i$. The expanded message block weights for each 80 rounds are estimated as

$W_j = M_j^{(i)}$ (3.8)

Where j = 0 to 15, W_j is a weight associated with the index j (where j=0 to 15), $M_j^{(i)}$ is a matrix or a set of matrices associated with indices j. The superscript (i) suggests that there is a sequence or iteration involved.

$W_j = Sigma_1(W_{j-2}) + W_{j-7} + Sigma_0(W_{j-15}) + W_{j-16}$ (3.9)

In equation (3.9) the index j varies between 16 to 79, Where j = 16 to 79.

Algorithm 1 Secure Hash Algorithm 512

Input: Text Message
Output: Hash value
Procedure:
BEGIN

Repeat steps 1 to 3 while (i <= N):

Step 1: Registers a; b; c; d; e; f ; g; h initialized with initial predefined hash values for the first iteration and with previous hash value for the current iteration i

a $\leftarrow \boldsymbol{Hash_1^{(i-1)}}$, b $\leftarrow \boldsymbol{Hash_2^{(i-1)}}$ ..... h $\leftarrow \boldsymbol{Hash_8^{(i-1)}}$

Step 2: Compression Function Module: Registers a,b,...h

Updating progress

Set j = 0

while(j <=79) do:

begin

Estimate Chf(e,f,g), Majf(a,b,c),$\sum_0(\boldsymbol{a})$,$\sum_1(\boldsymbol{e})$, and Wjusing the Equations,

$$\boldsymbol{Chf}(\boldsymbol{u},\boldsymbol{v},\boldsymbol{w}) = (\boldsymbol{u} \wedge \boldsymbol{v}) \oplus (\neg\, \boldsymbol{u} \wedge \boldsymbol{w})$$

$$\boldsymbol{Majf}(\boldsymbol{u},\boldsymbol{v},\boldsymbol{w}) = (\boldsymbol{u} \wedge \boldsymbol{v}) \oplus (\boldsymbol{u} \wedge \boldsymbol{w})) \oplus (\boldsymbol{v} \wedge \boldsymbol{w})$$

$$\sum_{0}(\boldsymbol{u}) = \boldsymbol{RS^{28}}(\boldsymbol{u})\oplus\boldsymbol{RS^{34}}(\boldsymbol{u})\oplus\boldsymbol{RS^{39}}(\boldsymbol{u})$$
$$\sum_{1}(\boldsymbol{u}) = \boldsymbol{RS^{14}}(\boldsymbol{u})\oplus \boldsymbol{RS^{18}}(\boldsymbol{u})\oplus\boldsymbol{RS^{41}}(\boldsymbol{u})$$

estimated values are used to update the registers,
T1 ⟵h + $\sum_1(\boldsymbol{e})$ + Chf(e, f ,g) + K_j+ W_j
T2⟵$\sum_0(\boldsymbol{a})$+ Majf(a, b,c)
Register updating as h⟵g, g⟵ f, f ⟵ e, e ⟵ d + T_1
d⟵c,c⟵b,b⟵a,a⟵ T1 + T2
end
Step 3: Hash value estimation at the level I

$$\boldsymbol{H_1^{(i)}} \longleftarrow a+\boldsymbol{H_1^{(i-1)}},\ \boldsymbol{H_2^{(i)}} \longleftarrow b + \boldsymbol{H_2^{(i-1)}},\ \dots,\ \boldsymbol{H_8^{(i)}} \longleftarrow h + \boldsymbol{H_8^{(i-1)}}$$

i⟵ i + 1
Output Hash of M as $\boldsymbol{H^N} = (\boldsymbol{H_1^{(N)}}, \boldsymbol{H_2^{(N)}}, \dots\dots, \boldsymbol{H_8^{(N)}})$

END

The following Figure 1 shows the work process of existing SHA-512 algorithm and its corresponding functions are noted by equations 3.14 to 3.21.

$$Chf(e,f,g) = (e \wedge f)\oplus(\neg\, e \wedge g) \quad (3.14)$$
$$Majf(a,b,c) = (a \wedge b)\oplus(a \wedge c))\oplus(b \wedge c) \quad (3.15)$$
$$\sum_0(a) = RS^{28}(a)\oplus RS^{34}(a)\oplus RS^{39}(a) \quad (3.16)$$
$$\sum_1(e) = RS^{14}(e)\oplus RS^{18}(e)\oplus RS^{41}(e) \quad (3.17)$$

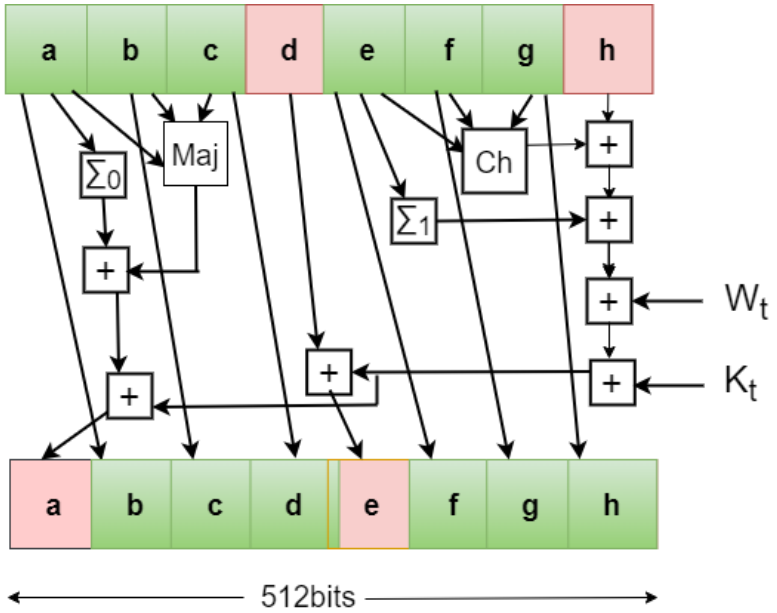

Figure 1. SHA-512 Algorithm (80 Rounds)

$$T1 = h + \sum_1 + chf(e,f,g) + Kt + Wt \quad (3.18)$$
$$T2 = \sum_0 + Majf(a,b,c) \quad (3.19)$$
$$W_t = h \quad (3.20)$$
$$K_t = d + T2 \quad (3.21)$$

The hash code is found using the equations (3.18 – 3.21), where K_t represent the round constant ($0 \le t \le 79$), $W_t$represent the word. T1 and T2 are used to find the hash code.Register updating as h⟵g, g⟵ f, f ⟵ e, e ⟵ d + T_1, d ⟵ c, c⟵ b, b ⟵ a
and a⟵ T1 + T2.

4. Proposed Exchanged SHA-512

This proposed Exchanged SHA-512 algorithm is an improved version of traditional SHA-512 with 60 rounds of operation which will reduce the overall computation of ESHA-512. Due to the reduction of 20 rounds, it saves the computation time as well as improves the security level with the help of different mode of operations. In the proposed ESHA-512 algorithm the total 60 rounds divided into three set of

progress with 20 rounds in each subdivision progress. Each set of twenty rounds, the registers values taken for the operation is in different form. In the approach the actual operation carried out in the SHA-512 is done as it is in the second set of 20 rounds that is from 21 to 40 rounds. But in the first and third set of twenty rounds the operation is derived with the relative position of registers used in the actual SHA-512 algorithm. In this proposed the initial hash value Hash(0)(64-bit) is derived from the first sixteen prime numbers. Square root is estimated for the first sixteen prime numbers from the value its fractional part is used as initial hash values. In this work the hash value is assigned in an alternate form as odd position hash value is assigned for the block with odd number of message length. Similarly, the even position hash value is assigned for the block which has message length in even.
The following Figure 2, shows the work progress of the first 20 rounds of the proposed ESHA-512.

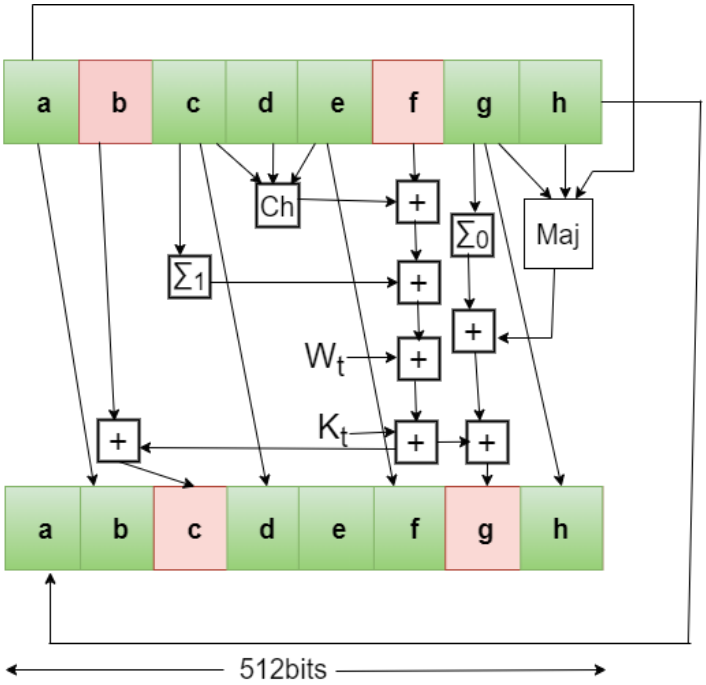

Figure 2. ESHA-512 Algorithm (For first 20 Rounds)

In this first 20 rounds the register the b and c is taken for operation as the register d and e taken in the actual SHA-512 algorithm. The registers used for the $Chf, Majf, \sum_0$and $\sum_1$ is chosen based on the relative position of the registers d and e. For the Chf function in the SHA-512, subsequent two registers (f, g) including 'e' is taken for the computation, hence in this ESHA-512 for the first round, the subsequent two registers (d, e) including c is taken for the Chf function. Similarly for the $Majf$ function the three preceding registers (a, b, c) from the dth register position is used for the computation. Hence in this round the three preceding registers (a, h, g) from the bthregister position. This is shown in the Eqn., 4.1 and 4.2.

$$Chf(c,d,e) = (c \wedge d) \oplus (\neg c \wedge e) \quad (4.1)$$

$$Majf(g,h,a) = (g \wedge h) \oplus (g \wedge a)) \oplus (h \wedge a) \quad (4.2)$$

For the $\sum_1$ function the register c is taken as 'e' in the SHA-512 and for $\sum_0$ register g is selected which is in the third position from left to right from the register 'b'. Simila1r to the register 'a' is selected in SHA-512 which is in the third position from left to right from the register 'd' as shown in the equations 4.3 to 4.4.

$$\sum_0(g) = RS^{28}(g) \oplus RS^{34}(g) \oplus RS^{39}(g) \quad (4.3)$$

$$\sum_1(c) = RS^{14}(c) \oplus RS^{18}(c) \oplus RS^{41}(c) \quad (4.4)$$

From the estimated value of these functions are further used to find$T1$ and $T2$ and update the registers similar to the relative position to the register 'b' and 'c' in the SHA-512. It is clearly shown in the following equations 4.5 to 4.8 as well as clearly depicted in figure 2.

$$T1 = f + \sum_1 + chf(c,d,e) + K[t] + W[t] \quad (4.5) \qquad T2 = \sum_0 + Majf(g,h,a) \quad (4.6)$$

$$W_t = f \quad (4.7)$$

$$K_t = b + T\,2 \quad (4.8)$$

Register updating ash⟵g, g⟵ f, f ⟵ e, d ⟵ c, c⟵ b+T1, b ⟵ a and g⟵ T1 + T2.

In this final 20 rounds, similar to the first 20 rounds all the registers are taken based on the relative position of the register 'f' and 'g' instead of 'd' and 'e' in traditional SHA-512. The subsequent two registers (h, a) in cycle form including g is taken for the Chf function. For $Majf$ function the three preceding registers (c, d, e) from the f^{th} register position is used for the computation. For the $\sum_1$ function the register g is taken as 'e' in the SHA-512 and for $\sum_0$ register c is selected which is in the third position from left to right from the register 'g'. The corresponding equations 6.31 to 3.38 show the whole progress of the final 20 rounds of ESHA-512. Figure 3 illustrates the proposed ESHA-512 working procedure.

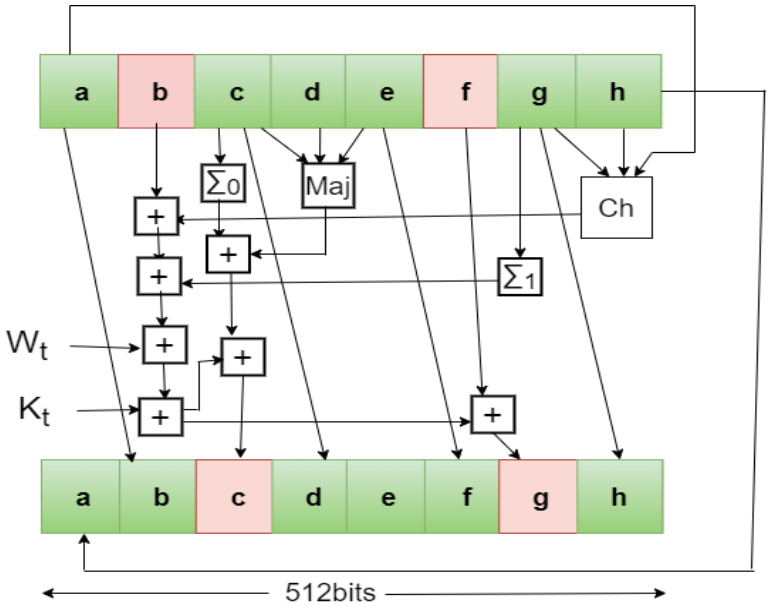

Figure 3. ESHA-512 Algorithm (For Final 20 Rounds)

$$Chf(g, h, a) = (g \wedge h) \oplus (\neg g \wedge a) \quad (4.9)$$

$$Majf(c, d, e) = (c \wedge d) \oplus (c \wedge e)) \oplus (d \wedge e) \quad (4.10)$$

$$\sum_0(c) = RS^{28}(c) \oplus RS^{34}(c) \oplus RS^{39}(c) \quad (4.11)$$

$$\sum_1(g) = RS^{14}(g) \oplus RS^{18}(g) \oplus RS^{41}(g) \quad (4.12)$$

$$T1 = b + \sum_1 + chf(g, h, a) + K[t] + W[t] \quad (4.13)$$

$$T2 = \sum_0 + Majf(c, d, e) \quad (4.14)$$

$$W_t = b \quad (4.15)$$

$$K_t = f + T2 \quad (4.16)$$

Register updating as h⟵ g, g⟵ b+T1, f ⟵ e, d ⟵ c, c⟵ T1+T2, b⟵ a.

5. Experimental Results and Analysis

The performance evaluation of the proposed ESHA-512 is conducted through a comparative analysis with traditional hashing algorithms such as SHA-512, SHA-256, and MD5, using blockchain simulations with varying block sizes. The assessment includes comprehensive metrics such as overall running time and the Number of Bytes Encrypted per unit Time (NBET). The experiments were executed on a Windows 10 platform featuring an Intel(R) Core (TM) i5-8300H CPU @ 2.30GHz, 8.00GB RAM, and a 64-bit operating system. The implementation of the blockchain concept was facilitated using Python 3.7.

In this study, each hashing algorithm is individually employed for blockchain hashing, and its running time performance is examined across different numbers of blocks. The analysis specifically focuses on job card management within the blockchain architecture, incorporating a fixed number of transactions per block (set at 2000 for analytical purposes). This investigation provides insights into the efficiency and efficacy of the proposed ESHA-512 algorithm compared to established hashing techniques in the context of blockchain technology.

5.1 Performance Analysis of SHA-512 and ESHA-512 Algorithms

Table 1 shows the outcomes of applying the SHA-512 (Secure Hash Algorithm 512-bit) and the proposed ESHA-512 (Exchanged SHA-512) algorithms to two different input strings along with the elapsed time for each computation. The elapsed time represents the time taken to compute the hash for each algorithm.

Table 1. Outcome of SHA-512 and the proposed ESHA-512

Input	SHA-512		ESHA-512	
	Outcome	Elapsed time (ms)	Outcome	Elapsed time (ms)
'Hello, world!'	c1527cd893c124773d811911970c8fe6e857d6df5dc9226bd8a160614c0cd963a4ddea2b94bb7d36021ef9d865d5cea294a82dd49a0bb269f51f6e7a57f79421	0.00040429999999958	4f45aa965e6a2ebff0bbbc62f4fed869a862ec8fae81a11c6ffb056944acebd44b109835d8ae9db39507ae4256067db3ef2479ea349d4e37411c91477e2b9ebf	0.00028800000001183435
'Transaction:Full and Oil Service, timestamp: 1683534359.0, billamt: 5689, branch:TVLN'	56059ff95e162fdbdab14c2baa868ae97dd28eef9a1c8a959314c08e70898d80d1340da86188d9cfa489ac35d3af1df86e1ea00d444bd6385bf5706ab875e5e1	0.0012462999999343083	430a1d712bc0956624df145eac85a4bd2c6624913c73597b02d7b09c452fce7d407c4382ab8ca2818a73598f28e48281d5b8a113cbfebd7334e104224b820566	0.0005897000000913977

Both SHA-512 and ESHA-512 generate unique hash values for each input. The hash is a fixed-length string of characters that represents the original data. The ESHA-512 algorithm demonstrates faster computation times for both input strings compared to the traditional SHA-512. This improvement in speed suggests that the proposed ESHA-512 algorithm more efficient in terms of computational performance.

5.2 Performance Analysis of various Hashing Algorithms in Blockchain

Table 2, compares the performance of different hashing algorithms (MD5, SHA256, SHA512, and ESHA-512) in terms of processing time for varying numbers of blocks in a blockchain.

Table 2. Analysis of Overall Running time (sec) vs Number of Blocks

Number of Blocks	Overall Running time (sec)			
	MD5	SHA256	SHA512	ESHA-512
100	0.40549	1.05464	2.96082	2.02672
200	1.17417	3.33096	6.28272	4.66912
300	1.21462	3.81088	7.81975	6.07085
400	2.27512	4.50335	12.77907	8.68160
500	2.36354	5.29840	15.85401	10.60720
600	2.97615	7.06000	19.18670	12.80208
700	3.18440	7.98800	20.00213	15.01920
800	3.56408	10.68960	25.98597	18.59438
900	3.82664	12.54156	27.26160	18.42490
1000	5.18260	13.58720	26.86219	20.76568

The overall running time for the MD5 is lower than all the approaches and the SHA-256 in the next level. MD5 exhibits relatively lower efficiency and slower performance, making it less suitable for modern cryptographic applications due to vulnerabilities. While SHA512 generally requires more processing time than SHA256, both remain widely used for their cryptographic strength. The proposed ESHA-512 algorithm exhibits improved efficiency, often outperforming SHA512 and providing a balance between security and computational speed. ESHA-512 consistently demonstrates superior efficiency compared to MD5, SHA256, and SHA512 across different block sizes.

Table 3 describes the time complexity of four cryptographic hash algorithms—MD5, SHA256, SHA512, and ESHA-512. All four algorithms exhibit linear time complexity, making them suitable for scenarios where the computational time is expected to increase linearly with the size of the input data.

Table 3. Time complexity analysis of the various algorithms

Algorithm	Time Complexity
MD5	θ(N)
SHA256	θ(N)
SHA512	θ(N)
ESHA-512	θ(N)

5.3 Performance Comparison of Bytes Encrypted per Second for Various Hashing Algorithms

Table 4 presents an analysis of the number of bytes encrypted per second for different hashing algorithms, including MD5, SHA256, SHA512, and the proposed ESHA-512, across varying numbers of blocks in a blockchain.

Table 4. Analysis of Number of bytes encrypted per sec vs Number of blocks

	Number of bytes encrypted per sec			
Number of Blocks	MD5	SHA256	SHA512	ESHA-512
100	1257719	483577	142249	211638
200	868695	306218	142350	218456
300	1259648	401482	145658	222024
400	896656	452996	149636	224979
500	1078890	481277	149842	230402
600	1028172	433427	159485	236023
700	1121090	446920	159487	237695
800	1144755	381679	167007	239421
900	1199486	365983	178368	249119
1000	1284062	375353	189857	255597

The results reveal the efficiency of each algorithm in handling different block sizes. it is found that the number of encrypted bytes for the MD5 is more efficient than the SHA family. But on the basis of security based on the number of operation and rounds the proposed ESHA-512 is better than all other approaches. The 1000blocks reveals that MD5 encryption rate is reported at 1,284,062 bytes per second, while SHA-256 achieves a rate of 375,353 bytes per second. SHA-512 exhibits a rate of 189,857 bytes per second, and the proposed ESHA-512 surpasses this with an impressive rate of 245,597 bytes per second. Particularly noteworthy is the achievement of ESHA-512, which outperforms SHA-512 by 55,739 bytes per second, highlighting its superior efficiency. However, according to the number of characters and encryption time of hash value, ESHA-512 is better than SHA-512, and SHA-256 is better than MD5, if security is considered.

6. CONCLUSION

The proposed Exchanged SHA-512 (ESHA-512) algorithm introduces a notable advancement in cryptographic hashing, striking a balance between enhanced security and computational efficiency. Through a reduction in the number of rounds and strategic register optimizations, ESHA-512 outperforms traditional SHA-512 in terms of overall running time and bytes encrypted per second. When comparing the average overall running time, ESHA-512 exhibits a reduction of 5 seconds compared to SHA-512. When comparing the average number of bytes encrypted per second, ESHA-512 shows a reduction of 65740 bytes per seconds compared to SHA-512 for the block size 1000. The algorithm's innovative approach, utilizing initial hash values derived from prime numbers and alternating their assignment based on message length, contributes to its unique strengths.In the context of blockchain simulations, ESHA-512 exhibits superior performance compared to established algorithms, showcasing its potential for secure and efficient data hashing. While MD5 and SHA-256 offer competitive computation times, ESHA-512 stands out as a compelling choice for applications where security is paramount.Overall, the proposed algorithm not only addresses computational efficiency concerns but also highlights the importance of security considerations in modern cryptographic applications. ESHA-512 emerges as a promising solution for blockchain implementations, offering a robust and efficient hashing mechanism for safeguarding sensitive data.

REFERENCES

[1] M. Antonio, M. Antonio, and G. Javier, ‘‘Dynamic security properties monitoring architecture for cloud computing,’’ in Security Engineering for Cloud Computing: Approaches and Tools. Pennsylvania, PA, USA: IGI Gobal, 2012, pp. 1–18.

[2] J. Toutouh, A. Muñoz, and S. Nesmachnow, ‘‘Evolution oriented monitoring oriented to security properties for cloud applications,’’ in Proc. 13th Int. Conf. Availability, Rel. Secur., Hamburg, Germany, Aug. 2018, pp. 1–7

[3] M. Li and P. P. C. Lee, ‘‘STAIR codes,’’ ACM Trans. Storage, vol. 10, no. 4, pp. 1–30, Oct. 2014.

[4] S. Cao, G. Zhang, P. Liu, X. Zhang, and F. Neri, ‘‘Cloud-assisted secure eHealth systems for tamper-proofing EHR via blockchain,’’ Inf. Sci., vol. 485, pp. 427–440, Jun. 2019.

[5] Y. Yuan, J. Zhang, W. Xu, and Z. Li, ‘‘Identity-based public data integrity verification scheme in cloud storage system via blockchain,’’ J. Supercomput., vol. 78, no. 6, pp. 8509–8530, Apr. 2022.

[6] G. Xie, Y. Liu, G. Xin, and Q. Yang, ‘‘Blockchain-based cloud data integrity verification scheme with high efficiency,’’ Secur. Commun. Netw., vol. 2021, pp. 1–15, Apr. 2021.

[7] M. Padmavathi and R. M. Suresh, “Secure P2P intelligent network transaction using Litecoin,” Mobile Networks and Applications, vol. 24, no. 2, pp. 318–326, 2018.

[8] P. Katsiampa, “Volatility estimation for Bitcoin: a comparison of GARCH models,” Economics Letters, vol. 158, pp. 3–6, 2017.

[9] I. Bentov and R. Kumaresan, “How to use Bitcoin to design fair protocols,” Lecture Notes in Computer Science, vol. 8617, pp. 421–439, 2017.

[10] W. Pennington and J. Evans, “Blockchain-enabled, subscriber-based capital markets index data distribution,” ;e Journal of Index Investing, vol. 7, no. 4, pp. 83–87, 2017

[11] G. Liang, S. R. Weller, F. Luo, J. Zhao, and Z. Dong, “Distributed blockchain-based data protection framework for modern power systems against cyber attacks,” IEEE Transactions on Smart Grid, vol. 10, no. 3, pp. 162–173, 2019.

[12] Q. Xia, E. B. Sifah, K. O. Asamoah, J. Gao, X. Du, and M. Guizani, “MeDShare: trust-less medical data sharing among cloud service providers via blockchain,” IEEE Access, vol. 5, no. 99, pp. 14757–14767, 2017.

[13] W. Chen, Z. Xu, S. Shi, Y. Zhao, and J. Zhao, “A survey of blockchain applications in different domains.” In Proceedings of the 2018 International Conference on Blockchain Technology and Application, pp. 17-21, 2018.

[14] M. Tekin, D. Öztürk, İ. Bahar, “AkıllıLojistikFaaliyetlerinde Blokzincir Teknolojisi”, Kent Akademisi, vol. 13(3), p. 570-583, 2020.

[15] X. Zhu, D. Wang, “Research on Blockchain Application for E-Commerce, Finance and Energy” In IOP Conference Series: Earth and Environmental Science, vol. 252, no. 4, p. 042126, IOP Publishing, 2019.

[16] L. Ismanto, H. S. Ar, A. N. Fajar, S. Bachtiar, “Blockchain as E-Commerce Platform in Indonesia”, In Journal of Physics: Conference Series, vol. 1179, p. 012114. IOP Publishing, 2019.

[17] S. Demirkan, I. Demirkan, A. McKee, “Blockchain technology in the future of business cyber security and accounting”, Journal of Management Analytics, vol. 7(2), p. 189-208, 2020.

[18] Ö. Aydın, S. Yükçü, “Siber Saldırı Önlemede Blokzinciri Teknolojisinin Fayda Maliyet Açısından Değerlendirilmesi”. MANAS Sosyal Araştırmalar Dergisi, vol. 9(4), p. 2519-2530, 2020.

[19] Süzen, Ahmet&Duman, Burhan. (2021). Blockchain-Based Secure Credit Card Storage System for E-Commerce. Sakarya University Journal of Computer and Information Sciences. 4. 204-215. 10.35377/saucis.04.02.895764.

[20] L. Zhu, Y. Wu, K. Gai, and K.-K. R. Choo, ‘‘Controllable and trustworthy blockchain-based cloud data management,’’ Future Gener. Comput. Syst., vol. 91, pp. 527–535, Feb. 2019

[21] Y. Ren, Y. Leng, J. Qi, P. K. Sharma, and A. Tolba, ‘‘Multiple cloud storage mechanism based on blockchain in smart homes,’’ Future Gener. Comput. Syst., vol. 115, pp. 304–313, Feb. 2021.

[22] Y. Ren, F. Zhu, P. K. Sharma, T. Wang, J. Wang, O. Alfarraj, and A. Tolba, ‘‘Data query mechanism based on hash computing power of blockchain

[23] Taherdoost, H.; Madanchian, M. Blockchain-Based E-Commerce: A Review on Applications and Challenges. Electronics 2023, 12, 1889. https://doi.org/10.3390/electronics12081889

[24] S. Demirkan, I. Demirkan, A. McKee, “Blockchain technology in the future of business cyber security and accounting”, Journal of Management Analytics, vol. 7(2), p. 189-208, 2020.

[25] J. R. Shaikh, G. Iliev, “Blockchain based confidentiality and integrity preserving scheme for enhancing e-commerce security” In 2018 IEEE Global Conference on Wireless Computing and Networking (GCWCN), pp. 155-158, 2018.

[26] Z. Liu, Z. Li, "A blockchain-based framework of cross-border e-commerce supply chain", International Journal of Information Management, vol. 52, 2020

[27] Liang, W.; Tang, M.; Long, J.; Peng, X.; Xu, J.; Li, K.C. A Secure Fabric Blockchain-based Data Transmission Technique for Industrial Internet-of-Things. IEEE Trans. Ind. Inform. 2019, 15, 358–3592

[28] K. Hao, J. Xin, Z. Wang, and G. Wang, ''Outsourced data integrity verification based on blockchain in untrusted environment,'' World Wide Web, vol. 23, no. 4, pp. 2215–2238, Jul. 2020, doi: 10.1007/s11280-019- 00761-2.

[29] Q. Wen, D. Wang, S. Feng, Y. Zhang, and G. Yu, "A novel cross-modal hashing algorithm based on multimodal deep learning," Science China (Information Sciences), vol. 60, no. 9, pp. 50–63, 2017.

[30] S. X. Xi, W. N. Zhang, Q. L. Zhou, S. XueMing, and B. Li, "High-throughput implementation of SHA512 algorithm based on mimetic computer," Computer Engineering and Science, vol. 40, no. 8, pp. 1344–1350, 2018.

[31] Fu, Jinhua&Qiao, Sihai& Huang, Yongzhong& Si, Xueming& Li, Bin & Yuan, Chao. (2020). A Study on the Optimization of Blockchain Hashing Algorithm Based on PRCA. Security and Communication Networks. 2020. 1-12. 10.1155/2020/8876317.

[32] Lee, S.; Seok, H.-W.; Lee, K.-r.; In, H.P. B-GPS: Blockchain-Based Global Positioning System for Improved Data Integrity and Reliability. ISPRS Int. J. Geo-Inf. **2022**, 11, 186. https://doi.org/10.3390/ijgi11030186

[33] Ranjith Kumar, M.V., Bhalaji, N. Blockchain Based Chameleon Hashing Technique for Privacy Preservation in E-Governance System. Wireless PersCommun **117**, 987–1006 (2021). https://doi.org/10.1007/s11277-020-07907-w

[34] Panwar, A., Bhatnagar, V. A cognitive approach for blockchain-based cryptographic curve hash signature (BC-CCHS) technique to secure healthcare data in Data Lake. Soft Comput (2021). https://doi.org/10.1007/s00500-021-06513-7

[35] B.P Kosta, and P.S. Naidu " Design and Implementation of a Strong and Secure Lightweight Cryptographic Hash Algorithm using Elliptic Curve Concept: SSLHA-160 ",(IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 12, No. 2, 2021

[36] Zeyad A. Al-Odat, Mazhar Ali, Assad Abbas, and Samee U. Khan. 2020. Secure Hash Algorithms and the Corresponding FPGA Optimization Techniques. ACM Comput. Surv. 53, 5, Article 97 (October 2020), 36 pages. doi:https://doi.org/10.1145/3311724

[37] Sharma, P., Jindal, R. & Borah, M.D. Blockchain-based distributed application for multimedia system using Hyperledger Fabric. Multimed Tools Appl (2023). https://doi.org/10.1007/s11042-023-15690-6

[38] Al-Ghuraybi, H.A., AlZain, M.A. &Soh, B. Exploring the integration of blockchain technology, physical unclonable function, and machine learning for authentication in cyber-physical systems. Multimed Tools Appl (2023). https://doi.org/10.1007/s11042-023-16979-2

[39] Sanober, A., Anwar, S. Crytographical primitive for blockchain: a secure random DNA encoded key generation technique. Multimed Tools Appl **81**, 40413–40430 (2022). https://doi.org/10.1007/s11042-022-13063-z.