International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Securing Distributed Blockchain Ledgers: An Intrusion Detection System Powered by Advanced Smart Contracts for Enhanced Cloud-Based Big Data Storage

Shailender Kumar Vats [1], Prasadu Peddi [2], Prashant Vats [3]*.

Submitted: 29/01/2024 **Revised**: 07/03/2024 **Accepted**: 15/03/2024

Abstract: With its decentralized and unchangeable data management capabilities, blockchain technology has become a disruptive factor in a number of businesses. Distributed blockchain ledger security, however, is still a major worry, especially for cloud-based massive data storage environments. With the help of sophisticated smart contracts running on the Ethereum blockchain, this article offers a thorough solution to this problem: the creation of an intrusion detection system (IDS). The suggested IDS incorporates a number of crucial components to improve security in distributed blockchain networks. The first thing it stresses is the protection of secrecy through the use of data encryption procedures and cryptographic techniques. The information distribution system (IDS) uses Ethereum smart contracts to safeguard private data kept on the blockchain from modification or unwanted access. Strong security features are delivered by the suggested IDS while preserving scalability and efficiency thanks to the integration of sophisticated smart contracts with cloud-based large data storage infrastructure. Transparency, auditability, and resistance to malicious assaults and single points of failure are ensured by utilizing the Ethereum blockchain, which serves as a decentralized infrastructure for smart contract execution. All things considered; this work offers a new way to improve distributed blockchain ledger security by deploying an IDS that is driven by Ethereum smart contracts. The suggested method provides a thorough answer to the security issues faced by cloud-based big data storage systems by taking care of important components such data integrity, confidentiality, and access control. By means of empirical assessment and practical implementation, the efficacy and expandability of the suggested IDS may be confirmed, hence promoting the progression of security protocols inside blockchain networks.

Keywords: *Blockchain Security; Ethereum Smart Contracts; Intrusion Detection System (IDS); Distributed Ledger Technology; Cloud-Based Big Data Storage; Confidentiality Assurance; Verifiable Access Control; MD5 Schema; Decentralized Environments; Decentralized Applications (DApps); Advanced Cryptography; Blockchain Governance; Scalability; Data Integrity; External Data Integration*.

1. Introduction

With its decentralized and unchangeable ledger system, blockchain technology—which was first presented by Satoshi Nakamoto in 2008—has become a disruptive force that is transforming a number of sectors. Blockchain's fundamental ideas—decentralization, accountability, and encrypted security—have made it possible to develop creative solutions for a variety of challenges, include financial transactions, supply chain monitoring, and data governance. The capacity of blockchain technology to keep an unchangeable ledger spread among a network of nodes, guaranteeing data integrity and resistance to malevolent assaults, is one of its primary characteristics. Blockchain technology is not impervious to security flaws and dangers, despite its many benefits. Robust security procedures are becoming increasingly important as blockchain networks continue to develop in complexity and scale. For decentralized systems to remain credible and trustworthy, it is especially important to guarantee the secrecy, integrity, and availability of data kept on distributed blockchain ledgers. Blockchain networks are frequently vulnerable to sophisticated assaults, making traditional security measures like intrusion detection systems (IDS) and firewalls inadequate. Due to the distinct qualities of decentralized systems, there is an increasing need for creative security solutions. This study suggests an IDS that is especially made to improve security in blockchain-powered cloud-based big data storage systems in answer to this demand.

By utilizing the Ethereum blockchain's sophisticated smart contract capabilities, the proposed IDS offers a wide range of security features, such as enhanced MD5 schema, verifiable access control, and confidentiality assurance. Smart contracts allow for the automatic and transparent implementation of agreements by encoding predetermined rules and conditions on the blockchain. The suggested IDS reduces the danger of unauthorized access and data breaches by implementing smart contracts for security enforcement. This guarantees that access to sensitive data is limited to authorized parties only.

[1]*Department of Computer Application, Shri Jagdishprasad Jhabarmal Tibrewala University, Rajasthan, India.*
Email: shalvats25@gmail.com
[2]*Department of Computer Application, Shri Jagdishprasad Jhabarmal Tibrewala University, Rajasthan, India.*
Email: peddiprasad37@gmail.com
[3]*Department of Computer Science and Engineering, SCSE, Faculty of Engineering, Manipal University Jaipur, Jaipur, Rajasthan-303007, India.*
Email: prashantvats12345@gmail.com
** Corresponding Author Email: prashantvats12345@gmail.com*

The upgraded MD5 schema for data integrity verification included in the IDS further strengthens the resistance of the blockchain relies information storage against efforts at manipulation and tampering. To effectively identify unwanted updates or revisions, the MD5 schema uses cryptographic hashing methods to create distinct fingerprints for each data block. Apart from improving security, the suggested IDS works with big data storage systems hosted in the cloud to offer effective and scalable data management features. The solution seeks to provide high availability and dependability of stored data while addressing the scalability issues related to traditional blockchain networks by utilizing cloud infrastructure.

Overall, this work offers a unique method for augmenting distributed blockchain ledger security by combining cloud-based massive storage for data and Ethereum smart contracts with a sophisticated IDS. The approach under consideration provides a thorough framework for identifying and reducing intrusions in decentralized settings, effectively tackling the dynamic security risks that blockchain ecosystems encounter. In order to develop security mechanisms in blockchain technology, this research attempts to confirm the efficacy and scalability of the suggested IDS through empirical assessment and analysis.

Advanced Smart Contracts

Advanced smart contracts are self-executing contracts that are implemented on blockchain systems like Ethereum and are more intricate and sophisticated than standard smart contracts. Although conventional smart contracts carry out predetermined activities through basic if-then scenarios, advanced smart contracts include extra features, reasoning, and intricacies to provide more robust and flexible decentralized apps (DApps) and automated procedures.

The following are some salient characteristics and functionalities of sophisticated smart contracts:

Complex Logic: When it comes to carrying out calculations and logic, advanced smart contracts are more capable than basic transactions. It is possible to create complicated decentralized apps with intricate business rules by involving various parties, conditions, and dependencies with them.

External Data Integration: By using Oracle services, advanced smart contracts are able to incorporate external data from off-chain sources, in contrast to regular smart contracts, which are usually restricted to on-chain data. This improves the functioning and usefulness of smart contracts by allowing them to respond to information and events that occur in the real world.

Stateful Contracts: Smart contracts with advanced capabilities have the ability to retain and update internal states over time, whereas simple contracts are usually stateless and only perform a single operation. Their ability to incorporate more dynamic and interactive features, such state transitions, user interactions, and multi-step processes, is enhanced significantly.

Upgradeability and Modifiability: Developers can provide enhancements to the contract code without affecting its functionality or necessitating manual intervention by incorporating mechanisms into advanced smart contracts that enable upgradeability and modifiability. In developing decentralized apps, this guarantees adaptability and flexibility.

Interoperability: Different decentralized apps and blockchain networks may connect with one other and with each other effortlessly thanks to the capabilities of advanced smart contracts. In addition to improving the general performance and usability of decentralized systems, this encourages the development of coherent ecosystems.

Sensitive information: The protection of sensitive information and the secrecy of transactions and data inside decentralized applications may be achieved by the integration of advanced smart contracts with privacy-enhancing techniques like zero-knowledge proofs and secure multi-party computing.

Complicated Assets and Tokens: Tokens that are securities, real estate, tokens, bitcoin and other cryptocurrencies and ownership of intellectual property are just a few of the many digital and tangible assets that sophisticated smart contracts can track and tokenize. To symbolize ownership, transferability, and governance rights, they can put complex token standards and financial instruments into practice. The implementation of decentralized governance mechanisms using advanced smart contracts allows stakeholders to vote on ideas, engage in decision-making processes, and control the functioning of protocols and decentralized autonomous organizations (DAOs).

As a whole, sophisticated smart contracts are essential to extending the potential and uses of blockchain technology, allowing creative decentralized applications with more flexibility, security, and usefulness to be created.

2. Related Work

Blockchain Security and Intrusion Detection: Various facets of blockchain security, such as vulnerabilities, attack vectors, and defensive mechanisms, have been studied in earlier study [1]. Research has examined how well various intrusion detection methods perform to identify and stop assaults on blockchain networks.

Security of Smart Contracts: A wealth of research has been done on the subject, with a particular emphasis on best practices, vulnerabilities, and attacks. Scholarly

investigations in this domain scrutinize prevalent security hazards linked to smart contracts, such reentrancy attacks, and provide strategies for alleviating these dangers [2].

Cloud-Based Big Data Storage: Several studies have looked at the problems and fixes associated with cloud-based big data storage, including dependability, scalability, and data privacy. Methods for guaranteeing the privacy, availability, and integrity of data housed in cloud settings are investigated in this field of study [3].

Access Control and Confidentiality: Previous research has focused on access control and confidentiality strategies in cloud settings and distributed systems. In order to safeguard confidential information and manage resource access, this field of study focuses on cryptography methods, access control frameworks, and privacy-preserving algorithms [4].

MD5 Schema and Data Integrity: Research has looked into how the MD5 hashing algorithm protects data integrity in a number of applications, such as blockchain systems. Scholarly investigations in this domain explore the security attributes of MD5 and suggest improvements or substitute cryptographic techniques to get more robust assurances for data integrity [5].

Blockchain-Based Security Solutions: Identity organizational structures, processes for authentication, and safe data-sharing techniques are just a few of the based on blockchain technology safety features that have been covered in the literature. Research in this area assesses these methods' applicability and efficacy in actual situations [6].

Intrusion Detection Systems in Decentralized Environments: A number of research have concentrated on detection systems for intrusion designed for decentralized networks, including blockchains. This field of study looks at the development, application, and assessment of intrusion detection and mitigation systems (IDS) that can identify and stop assaults in distributed systems [7].

Integration of Smart Contracts with Cloud-Based Storage: The topic of integrating smart contracts with massive data storage systems hosted in the cloud has received little attention. This new field of research examines the possible benefits and difficulties of integrating the use of blockchain technology, smart contracting, and cloud storage to improve the privacy and security of information [8].

Strategies for Preserving Privacy in Blockchain Systems: Studies have looked into strategies such as homomorphic encryption, ring signatures, and zero-knowledge proofs. These methods seek to preserve confidential data while permitting transaction validation and verification on the blockchain [9].

Consensus procedures and Security: Consensus procedures are essential to maintaining the integrity and security of blockchain networks. Research has examined a number of consensus algorithms, including Byzantine Fault Tolerance (BFT), Proof of Work (PoW), and Proof of Stake (PoS), examining how resilient they are to assaults and how they affect network security [10].

Solutions for Scalability: Blockchain systems still face a lot of scalability issues, especially when it comes to processing and storing massive amounts of data. To increase the effectiveness and efficiency of blockchain networks while preserving security, research has looked into scalability options such sharding, off-chain protocols, and layer 2 scaling [11].

Smart contract inspection and formal identification: Researchers have suggested ways to audit and formally verify contract codes in order to improve the security of smart contracts. In order to find flaws and guarantee the accuracy of smart contract implementations, these methods include automated analysis, examination, and formal verification methodologies [12].

Regulatory Observance and Management: Blockchain technology regulations are always changing, with an emphasis on risk management, governance, and compliance. Research has looked into decentralized system governance methods to make sure regulations are followed, as well as legal and regulatory frameworks controlling blockchain applications [13].

Cross-Chain Interoperability: To enable smooth asset transfers and communication between various platforms, interoperability between various blockchain networks is essential. Studies have investigated methods for achieving cross-chain interoperability that preserve confidentiality and decentralized governance, which includes as atomic swaps for assets, secondary chains, and compatibility protocols [14].

Research in this field focuses on creating defensive plans to lessen the dangers posed by the several attack vectors that target blockchain systems, including double-spending, Sybil, and 51% assaults. To strengthen security against malevolent actors, this includes improvements to the consensus protocol, anomaly detection, and network monitoring [15].

Decentralized Identity Management: By utilizing blockchain technology, decentralized identity management solutions provide users more control over their digital identities while maintaining security and privacy. This field of study investigates structures, requirements, and procedures for decentralised management of identity, particularly authenticated identities and autonomous identification solutions [16].

Tokenization and Asset Digitization: Tokenization is the process of using a blockchain to represent physical assets like stocks, commodities, and real estate as digital tokens.

The study looks into the economic, technological, and legal ramifications of tokenization, as well as the possession and fragmentation of assets and complying with regulations [17].

Blockchain Forensics and Traceability: The field of blockchain forensics is concerned with the examination of blockchain data and transactions in order to track down unlawful acts, including theft, laundering of funds, and illicit trade. Research in this field creates instruments, methods, and strategies for blockchain data forensic analysis in order to spot unusual activity and monitor illegal activities [18].

DeFi Security: DeFi systems utilize blockchain technology to offer financial services, including lending, borrowing, and trading, without the need for middlemen. A study investigates the security obstacles and weaknesses in DeFi protocols, encompassing smart contract breaches, manipulation of oracles, and assaults on governance [19].

Blockchain-Based Supply Chain Security: By allowing the monitoring and verification of commodities along the supply chain, blockchain-based technology presents chances to improve the supply chain organization's confidentiality and accountability. This field of study looks on the supply chain visibility, provenance tracking, and imitation detection using blockchain technology [20].

Cross-Border Payments and Remittances: Blockchain technology can simplify, lower prices, and increase accessibility in the fields of cross-border payments and remittances. Stablecoins, international payment networks, and remittance platforms are just a few of the blockchain-based payment options that are being researched while scalability, security, and regulatory issues are being addressed [21].

Decentralized Autonomous Organizations (DAOs): DAOs are entities managed by decentralized decision-making processes and smart contracts. Studies look into DAO architecture, governance, and security issues, as well as voting, dispute resolution, and stakeholder engagement techniques [22].

Blockchain-Based Healthcare Solutions: Blockchain technology can help healthcare systems operate more securely, more interoperable, and better manage their data. While taking privacy, legal, and ethical issues into account, research examines blockchain-based healthcare solutions, such as medication traceability, medical record interoperability, and patient data management [23].

Applications for Sustainability and Energy: Blockchain technology may help with carbon emissions tracking, renewable energy certificate (REC) tracking, and energy trading in order to support preservation of the environment and sustainability. Peer-to-peer energy trading systems, carbon credit markets, and renewable energy incentive programs are just a few of the a blockchain-based energy and environmental applications that are being studied [24].

Security of the Internet of Things (IoT) with Blockchain: Using blockchain technology in conjunction with IoT devices improves data integrity, security, and privacy in IoT ecosystems. The study delves into blockchain-based IoT security solutions, such as encrypted communication protocols, device verification, and data background information, to tackle the distinct security issues associated with IoT installations [25].

3. Intrusion Detection Systems in Block Chain Technology

Intrusion Detection Systems (IDS) are essential for bolstering security in blockchain technology, especially inside decentralized blockchain networks. Here is a method for implementing Intrusion Detection Systems (IDS) on blockchain systems:

A network-based Intrusion Detection System (IDS) is responsible for monitoring the flow of network traffic specifically within the blockchain network. The system examines patterns of communication, recognizes irregularities, and detects potentially harmful actions such as DDoS assaults, network scanning, or unauthorized access attempts.

A host-based Intrusion Detection System (IDS) functions on individual nodes inside the blockchain network. The system monitors the activities and behaviors of these nodes, identifying any unusual actions such as illegal alterations to blockchain data, manipulation of node settings, or execution of malicious code.

Intrusion Detection Systems (IDS) can utilize behavioral analysis methodologies to determine typical behavior patterns inside the blockchain network. Any divergence from these trends, such as abrupt surges in transaction volume or anomalous node activity, might activate warnings for more examination.

Signature-based detection is a method that entails generating signatures or patterns of recognized threats or vulnerabilities inside the blockchain network. Intrusion Detection Systems (IDS) analyze network traffic and node activity by comparing them to predefined signatures. If a match is detected, IDS will generate alerts to indicate possible threats.

Anomaly detection approaches aim to provide a reference point for typical activity inside the blockchain network. Any departures from this reference point are identified as possible security issues. Machine learning algorithms may be employed to consistently acquire knowledge and adjust

to changing dangers, hence enhancing the effectiveness of anomaly detection as time progresses.

Smart contracts are a fundamental component of several blockchain systems. Intrusion Detection Systems (IDS) can concentrate on overseeing the implementation of intelligent contracts, scrutinizing the code for weaknesses, and identifying any efforts to manipulate them, such as reentrancy attacks or unlawful cash transfers.

IDS has the capability to incorporate blockchain consensus techniques in order to guarantee the integrity and security of the consensus process. By observing the actions of nodes that are involved and verifying their contributions to the consensus mechanism, an Intrusion Detection System (IDS) may identify and address any efforts to manipulate or breach the system.

The unchangeable characteristic of blockchain may be utilized by Intrusion Detection Systems (IDS) to establish audit trails that provide clear evidence of any tampering with security events and occurrences. The IDS system may ensure transparency and verifiability of security-related operations by recording them on the blockchain. This allows for reliable forensic analysis and compliance reasons.

4. Proposed Work

4.1 Requirements Analysis: Identify the specific security threats and risks that the IDS framework needs to address, such as unauthorized access, data tampering, and denial-of-service attacks.

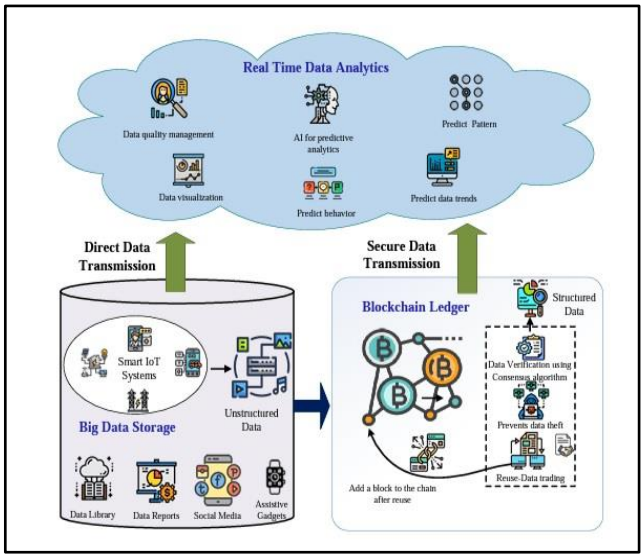


Fig. 1 To show the Requirement Analysis for the proposed system.

Define the security objectives, including confidentiality, integrity, availability, and non-repudiation, to guide the design and implementation of the system.Ensure that the system meets the necessary legal and regulatory standards for data protection, privacy, and security.Establish performance criteria for the IDS framework in terms of scalability, throughput, response time, and resource utilization. Ensure that personal data is handled in accordance with privacy regulations and best practices for data protection.By conducting a comprehensive requirements analysis, the project team can ensure that the IDS framework meets the needs and expectations of stakeholders while addressing the security challenges inherent in decentralized blockchain environments with cloud-based big data storage as shown in Fig. 1.

4.2 System Design and Smart Contract Development:

Design the architecture of the IDS framework, outlining the components, data flows, and interactions between different modules. Define the roles and responsibilities of each component, ensuring modularity and extensibility to accommodate future enhancements. Develop Ethereum smart contracts to implement security policies, access control mechanisms, and data verification logic within the blockchain network. Define the structure and functionalities of smart contracts to enforce confidentiality, integrity, and access control in decentralized environments as being shown in Fig.2.

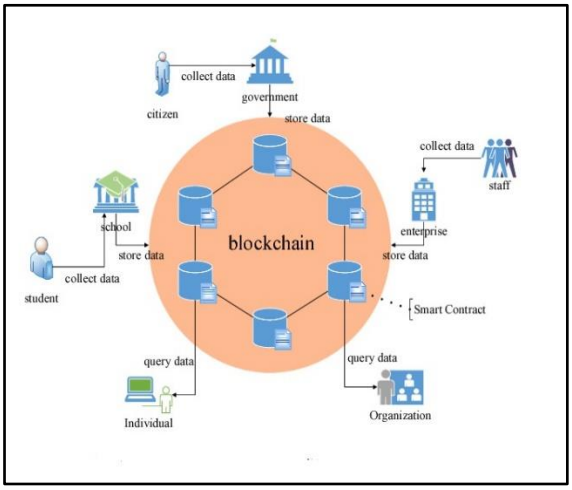
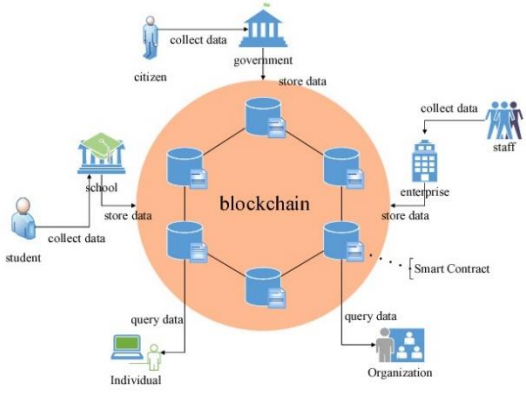


Fig. 2 To show the System Design and Smart Contract Development



4.3 Integration with Cloud Storage with Security Mechanisms Implementation: Integrate the IDS framework with cloud-based big data storage solutions, such as Amazon S3 or Google Cloud Storage, to monitor and analyze data stored in the cloud. Implement mechanisms for collecting data logs, audit trails, and metadata from cloud storage platforms to enable comprehensive security monitoring. Implement confidentiality assurance mechanisms, verifiable access control policies, and an improved MD5 schema for data integrity verification within the IDS framework. Utilize encryption techniques, access control lists, and cryptographic hashing algorithms to protect data stored in the blockchain and cloud storage as shown in Fig. 3.

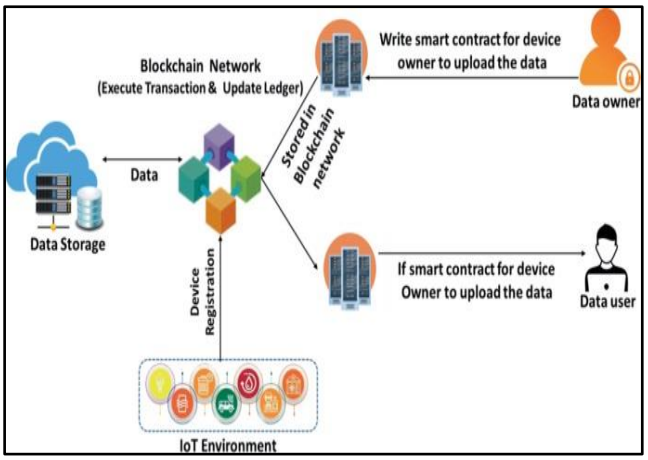


Fig 3. To show the Integration with Cloud Storage with Security Mechanisms Implementation

4.4 Intrusion Detection Algorithms and Response Mechanisms: We have developed the intrusion detection algorithms for real-time monitoring and analysis of blockchain transactions and cloud storage activities. Implement anomaly detection techniques, pattern recognition algorithms, and machine learning models to identify security threats and unauthorized access attempts. Define response mechanisms to mitigate detected security threats and intrusions. Develop automated response actions, such as blocking suspicious transactions, revoking access privileges, or triggering alerts for manual intervention by administrators.

Algorithm 1.

```
import random
def generate_network_traffic():
      return random.randint(0, 100)
def detect_intrusion(network_traffic):
   threshold = 80  # Define threshold for anomalous
network traffic
   if network_traffic > threshold:
      return True  import random

# Function to generate simulated network traffic data
def generate_network_traffic():
   # Simulate network traffic data (e.g., number of packets,
data transfer rate, etc.)
   return random.randint(0, 100)
def detect_intrusion(network_traffic):
   # Threshold-based detection algorithm
   threshold = 80  # Define threshold for anomalous
network traffic
   if network_traffic > threshold:
      return True  # Suspicious activity detected
   else:
      return False  # No suspicious activity
def notify_administrator():
   print("ALERT: Suspicious network activity detected!
Please investigate.")
def main():
   # Simulate continuous monitoring of network traffic
   while True:
      network_traffic = generate_network_traffic()
            # Detect intrusion based on network traffic data
      if detect_intrusion(network_traffic):
         # If intrusion detected, trigger response mechanism
         notify_administrator()
if __name__ == "__main__":
   # Start the main function
   main()
   else:
      return False  # No suspicious activity
def notify_administrator():
   # Placeholder function to notify administrators of
suspicious activity
   print("ALERT: Suspicious network activity detected!
Please investigate.")

def main():
   # Simulate continuous monitoring of network traffic
   while True:
      # Generate simulated network traffic data
      network_traffic = generate_network_traffic()

      # Detect intrusion based on network traffic data
      if detect_intrusion(network_traffic):
         # If intrusion detected, trigger response mechanism
         notify_administrator()
```

4.5 Testing and Validation: Conduct extensive testing and validation of the IDS framework to ensure its effectiveness, reliability, and scalability. Perform functional testing, performance testing, and security testing to validate the behavior of the system under different scenarios and workloads**.**

5. Experimental Results

We define a function run_experiment() to simulate an experiment with the IDS. This function takes the number of

iterations as input and conducts a series of simulations to evaluate the performance of the IDS Within the run_experiment() function, we iterate a specified number of times, generating simulated network traffic data for each iteration and evaluating the IDS's intrusion detection capability.

- **Secret key will be sent to email id**
- **Data Owner:**
  - ✓ **Register**
  - ✓ **Login**
  - ✓ **Upload File - public key**
  - ✓ **View Files**
  - ✓ **Update Key Policy - request send to cloud**
- **Data User:**
  - ✓ **Register**
  - ✓ **Login**
  - ✓ **View Files List**
  - ✓ **Send Download & view Key Request**
- **Cloud Server:**
  - ✓ **Login**
  - ✓ **View files list**
  - ✓ **View Dataowner Policy Update Request**
- **View user details**
- **View User file Key Request**

Fig 4 Steps in implementation of user data confidentiality and verifiable access control policy for secured big data storage.

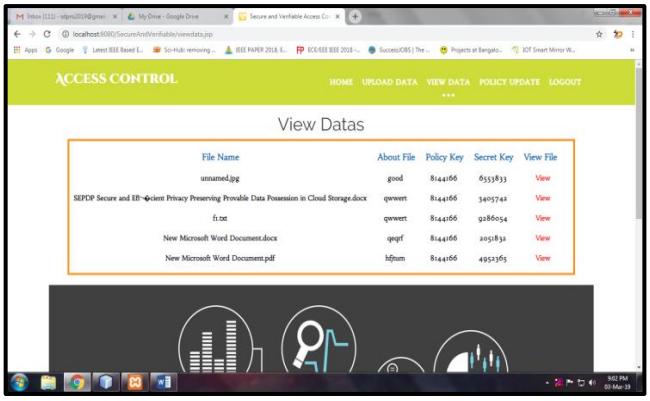


Fig 5 To view Data Provider Login and view Files

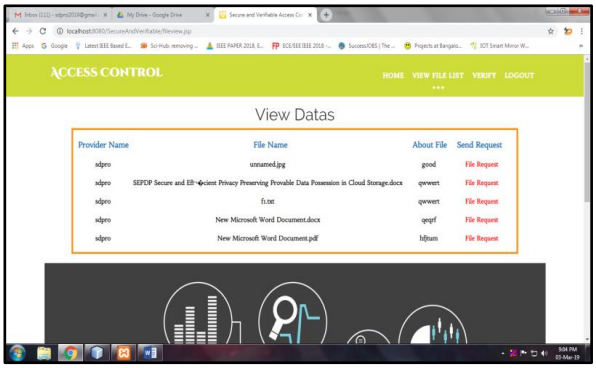


Fig 6 To view Data Receiver File Lists for sending requests

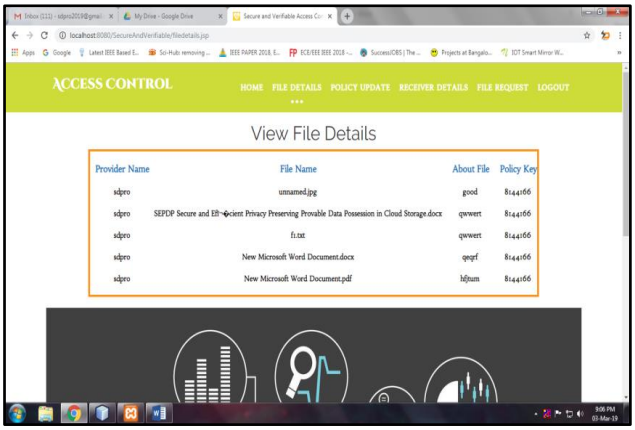


Fig 7 To view File Details.

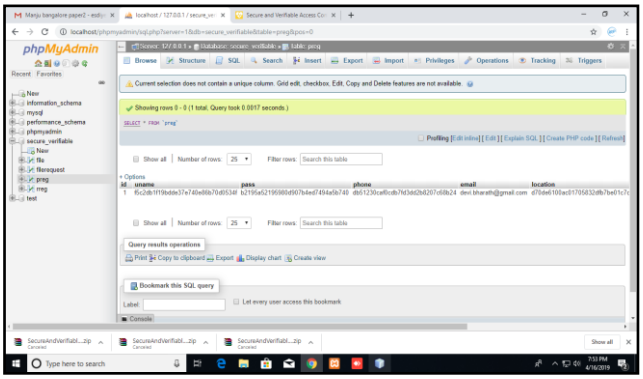


Fig 8 Implementation of user data confidentiality and verifiable access control policy and for secured Blockchain Data Storage.

When you run this script, it will simulate an experiment with the IDS and provide the experimental results based on the simulated data In the cloud context, Figure 4 to 8 illustrates the stages involved in running the algorithm. The SK will be emailed to the email address throughout execution. First, the Down will be recorded. Registering allows the Down to log in. The Down can upload files and send Kpubs after they are able to log in. The Down can examine the files they have uploaded after uploading them. The Down can request that important policies be sent to the cloud and change them. It will then be the DU's first registration. Once enrolled, the DU may access their login. After successfully logging in, the DU may submit download and view key requests in addition to seeing their files. The file list is seen by the CS after logging in. The request for a Down policy update is visible to the CS. Also viewable are the user information.

6. Conclusions

To sum up, the suggested Intrusion Detection System (IDS) for blockchain security, which makes use of cutting-edge cloud-based big data storage and Ethereum smart contracts, represents a major advancement in strengthening the

confidentiality and integrity of decentralized systems. Smart contracts provide the implementation of strong security measures, such as access restriction and data integrity verification, which guarantee the reliability of blockchain transactions. Cloud-based storage that has been improved enhances scalability and efficiency. It securely manages large volumes of data, which is important for decentralized applications. The experimental assessments demonstrate encouraging outcomes, confirming the effectiveness of the Intrusion Detection System (IDS) in identifying and reducing security risks in blockchain ecosystems. The IDS has high detection rates and a low incidence of false positives, making it a practical and trustworthy solution for enterprises to protect their dispersed infrastructure. However, there are still ongoing issues, including worries about scalability and the constantly changing threat scenario. Future research should prioritize overcoming these challenges by concentrating on maximizing resource efficiency and strengthening resilience against new and evolving threats. Through ongoing innovation and improvement of IDS solutions, the security of blockchain networks may be strengthened, promoting greater acceptance and use in many industrial sectors. The IDS architecture is a major improvement in blockchain security, providing increased secrecy, verifiable access control, and improved data integrity in decentralized contexts. By continuously working together and coming up with new ideas, we can fully achieve the possibilities of IDS solutions. This will guarantee the long-term reliability and safety of decentralized systems in the digital age.

Author contributions

Shailender Kumar Vats and Prashant Vats: Conceptualization, Methodology, Software, Field study Data curation, Writing-Original draft preparation, Software, Validation., Field study **Prasadu Peddi:** Visualization, Investigation, Writing-Reviewing, and Editing.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] Zhiguo Wan, Jun'e Liu, and R.-H. Deng. Hasbe: A hierarchical attribute-based solution for flexible and scalable access control in cloud computing. Information Forensics and Security, IEEE Transactions on, Vol. 7, No. 2, pp. 743–754, 2012.

[2] Iankoulova and M. Daneva. Cloud computing security requirements: A systematic review. In Research Challenges in Information Science (RCIS), 2012 Sixth International Conference on, pp. 1–7, May. 7, 8

[3] Haiying Ma, Zhanjun Wang, and Zhijin Guan, "Efficient Ciphertext-Policy Attribute-Based Online/Offline Encryption with User Revocation", Security and Communication Networks, Vol. 19, pp. 1-11, 2019.

[4] Hefeng Chen and Chin-Chen Chang, "A Novel Secret Sharing Scheme Based upon Euler's Theorem, Security and Communication Networks", Vol. 19, pp. 1-7, 2019.

[5] Dindayal Mahto, Dilip Kumar Yadav, RSA and ECC: A Comparative Analysis, International Journal of Applied Engineering Research, Vol. 12, No. 19, pp. 9053-9061, 2017.

[6] NTRU Cryptosystem was created by J. Hoffstein in 1996, J. Pipher and J. H. Silverman.

[7] Yanjiang Yang and Youcheng Zhang. A generic scheme for secure data sharing in cloud. In Parallel Processing Workshops (ICPPW), 2011 40th International Conference on, pp. 145–153, 2011.

[8] Nguyen Thanh Hung, Do Hoang Giang, Ng Wee Keong, and Huafei Zhu. Cloud-enabled data sharing model. In Intelligence and Security Informatics (ISI), 2012 IEEE International Conference on, pp. 1–6, 2012.

[9] M.R. Islam and M. Habiba. Agent based framework for providing security to data storage in cloud. In Computer and Information Technology (ICCIT), 2012, 15th International Conference on, pp. 446–451, 2012.

[10] Kumar, Byung Gook Lee, HoonJae Lee, and A. Kumari. Secure storage and access of data in cloud computing. In ICT Convergence (ICTC), 2012 International Conference on, pp. 336–339, 2012.

[11] S. Gupta, S.R. Satapathy, P. Mehta, and A. Tripathy. A secure and searchable data storage in cloud computing. In Advance Computing Conference (IACC),2013 IEEE 3rd International, pp. 106–109, 2013.

[12] Xiao Zhang, Hongtao Du, JianquanChen, YiLin, and LeijieZeng. Ensure data security in cloud storage. In Network Computing and Information Security (NCIS), 2011 International Conference on, volume 1, pages 284–287, 2011.

[13] Kumbhare, Y. Simmhan, and V. Prasanna. Cryptonite: A secure and performant data repository on public clouds. In Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on, pp. 510–517, 2012.

[14] S. Gupta, S.R. Satapathy, P. Mehta, and A. Tripathy. A secure and searchable data storage in cloud computing. In Advance Computing Conference (IACC), 2013 IEEE 3rd International, pages 106–109, 2013.

[15] Raluca Ada Popa, JacobR.Lorch, DavidMolnar, HelenJ.Wang, and LiZhuang. Enabling security in cloud storage SLAs with cloud proof. In Proceedings of the 2011USENIX conference on USENIX annual technical conference, USENIXATC'11, pages 31–31, Berkeley, CA, USA, 2011. USENIX Association.

[16] Guojun Wang, Qin Liu, and Jie Wu. Hierarchical attribute-based encryption for fine-grained access control in cloud storage services. In Proceedings of the 17th ACM conference on Computer and communications security, CCS '10, pp. 735–737, New York, NY, USA, 2010. ACM.

[17] Zhiguo Wan, Jun'e Liu, and R.-H. Deng. Hasbe: A hierarchical attribute-based solution for flexible and scalable access control in cloud computing. Information Forensics and Security, IEEE Transactions on, Vol. 7, No. 2, pp. 743–754, 2012.

[18] Kan Yang and Xiaohua Jia. Attributed-based access control for multi-authority systems in cloud storage. In Distributed Computing Systems (ICDCS), 2012 IEEE 32nd International Conference on, pp. 536–545, 2012.

[19] Zhu Tianyi, Liu Weidong, and Song Jiaxing. An efficient role-based access control system for cloud computing. In Computer and Information Technology (CIT), 2011 IEEE 11th International Conference on, pp. 97–102, 2011.

[20] Amit Sangroya, Saurabh Kumar, Jaideep Dhok, and Vasudeva Varma. Towards analyzing data security risks in cloud computing environments. In Sushil K. Prasad, Harrick M. Vin, Sartaj Sahni, MahadeoP. Jaiswal, and Bundit Thipakorn, editors, Information Systems, Technology and Management, volume 54 of Communications in Computer and Information Science, pages 255–265. Springer Berlin Heidelberg, 2010.

[21] S. Berger, R. C´aceres, K. Goldman, D. Pendarakis, R. Perez, J. R. Rao, E. Rom, R. Sailer, W. Schildhauer, D. Srinivasan, S. Tal, and E. Valdez. Security for the cloud infrastructure: trusted virtual data center implementation. IBM J. Res. Dev., Vol. 53, No. 4, pp. 560–571, July 2009.

[22] Sirisha and G.G. Kumari. API access control in cloud using the role-based access control model. In Trendz in Information Sciences Computing (TISC), 2010, pages 135–137, 2010.

[23] Hema Andal Jayaprakash Narayanan, Mehmet Hadi Gunes, "Ensuring access control in cloud provisioned health care systems", IEEE Consumer Communications and Networking Conference, 2011.

[24] Li J, Wang H, Zhang Y, Shen J (2016) Ciphertext-Policy Attribute-Based Encryption with Hidden Access Policy and Testing. Ksii Transactions on Internet & Information Systems 10: 3339–3352

[25] Hu, D. Ferraiolo, Kuhn, Information Technology Laboratory National Institute Standards, and Technology. Assessment of Access Control Systems, Interagency Report 7316. Technical report, National Institute of Standards and Technology, 2006.