The Scientific Temper
SEPTEMBER 2026 Vol. 17 (9): 6928-6941
ISSN: 0976-8653, E-ISSN: 2231-6396
DOI: 10.58414/SCIENTIFICTEMPER.2026.17.9.2551
A WEB OF SCIENCE JOURNAL

RESEARCH ARTICLE

SCDTACBC: A Smart Contract-Driven Dynamic Trust-Based Access Control for Secure Healthcare Data Sharing using Permissioned Blockchain

M.SUKANYA[1*], DR. V. JUDE NIRMAL[2]

[1]*Research Scholar, Department of Computer Science, St.Joseph's College (Autonomous), Affiliated to Bharathidasan University, Tiruchirappalli, 620 002, Tamil Nadu, India*

[2]*Assistant Professor & Research Supervisor, Department of Computer Science, St.Joseph's College (Autonomous), Affiliated to Bharathidasan University, Tiruchirappalli, 620 002, Tamil Nadu, India*

Corresponding author: sukan2suresh@gmail.com

ABSTRACT

In a healthcare domain, illegal access and data breaches are major privacy concerns during data exchange. Access control is a prevalent method for safeguarding sensitive information. The traditional access control system used centralized architecture and static model. It is inefficient due to vulnerable or a single point of failure. Nowadays, Blockchain based smart contract improve the privacy, and integrity of healthcare data while simultaneously enhance security via automated authentication and access control. This paper proposes smart contract-driven dynamic trust-based access control for secure health data sharing (SCDTACBC). The system employs authentication contract for participation validation, policy contract for managing polices, trust evaluation contract for computing trust score based on behavior and access control contract to regulate the workflow of the entire access control process. The behavior and trust-based mechanism is designed to improve the access control. The trust score can be computed using user behavior which includes correctness, availability and malicious. To guarantee blockchain scalability, data is stored in an off-chain database called Interplanetary File System (IPFS). By combining blockchain with smart contract, the proposed SCDTACBC can handle conflicting scenarios with scalability and tamper-resistant. The performance of the SCDTACBC is evaluated using throughput and latency. The proposed approach attains higher throughput (65.45 TPS) and also lower latency (0.018 Sec.) The SCDTACBC approach increases 27.58% of throughput and decreases 40% latency compared to other existing methods. The experimental results demonstrate the proposed work scalability and practicality with respect to throughput and latency.

Keywords: *Blockchain, Smart contract, Healthcare, Access control, IPFS.*

INTRODUCTION

The rapid integration of e-health systems and digital health applications has significantly changed healthcare delivery, by enabling remote patient monitoring, telemedicine and efficient electronic health records Preetha, A.D. et. al., (2023). Unauthorized access, data breaches and identity theft are just a few of the major security and privacy issues that have arisen as a consequence of growing dependence on digital platforms Yakubu, B.M. et. al., (2024). Access control is an extremely important factor in this context, for ensuring the safety and confidentiality of the information. It checks that the system resources are being used sensibly and appropriately by prohibiting both malicious users and authorized users from gaining unauthorized access or using them improperly Xiang et al., (2022).

Received: 5th September, 2026 **Accepted: 7th September, 2026** **Published: 28th September, 2026**

Mandatory access control, role-based access control Mpamugo, E. et. al., (2024) and attribute access control Karimi, L., et al., (2021) are some types of conventional access control which require a central entity to make decisions. These conventional access control methods encounter issues including single point of failure and untrustworthiness.

One possible solution to these problems is blockchain technology, which possesses a decentralized, immutable and cryptographically secure framework Abdi, A.I., et al., (2020). Blockchain enables safe authentication, access control, and data integrity protection in healthcare systems by using distributed consensus techniques and a tamper-resistant ledger.

Numerous researchers investigated the viability of developing blockchain-based access control system. The author Zhang, L., et al., (2023) utilized blockchain as an immutable database to provide consistency and reliability in decentralized rules and decision-making information. The author Kamboj, P., et. al., (2021) utilized smart contracts on blockchain to facilitate automated and verifiable authorizations. With blockchain technology serving as an immutable distributed ledger, Sigwart, M., et al., (2024) have suggested blockchain-based management approach to increase the reliability of the data provenance.

A smart contract comprises a series of digitally articulated conditions. Blockchain technology obviates the necessity for intermediaries via smart contracts Sharma, P., et al., (2023). All regulations governing transactions between two parties are delineated in smart contracts. The transaction cannot proceed until all stipulations of the smart contract are satisfied. In blockchain-based networks, a smart contract is automatically activated whenever a transaction is initiated. To overcome the challenges of using conventional access control methods, smart contracts are created and deployed on the blockchain to automate the authentication and authorization process allowing for non-centralized, transparent, fast, flexible and dynamic access control Lee, J.S., et al., (2022).

This paper proposes a SCDTACBC, smart contract-based access control using blockchain for securing healthcare data sharing. Different smart contracts are created for enhancing the security and privacy of healthcare data. The smart contract includes authentication contract (auth_sctr), policy contract (pol_sctr), trust evaluation contract (teval_sctr) and access control contract (actr_sctr) which provides user validation, policy management, compute trust based on behavior and fine-grained access control. In addition to this, this research utilizes interplanetary file system (IPFS) for secure healthcare storage. It offers decentralized alternative for maintaining massive medical files using cryptographic hashes to guarantee data integrity and rapid retrieval Azbeg, K., et al., (2022). The main contributions of this research are:

- Dynamic trust-based access control is suggested for efficient access control mechanism using smart contract.
- The trust can be computed based on the user behavior including correctness, availability and malicious.
- A series of experiments is conducted to establish its practicality and efficacy. The experimental result indicates that the suggested SCDTACBC is both viable and scalable regarding throughput and latency.

The subsequent sections of the paper are organized as follows: Section 2 provides the related work of blockchain-based access control. Section 3 explains the proposed smart contract-based access control mechanism. Section 4 examines the experimental results with security and privacy analysis, while Section 5 presents the conclusion and provides possibilities for the future research.

RELATED WORKS

Protecting patient information, ensuring authentication, integrity and managing access control mechanisms are important security concerns due to the increasing reliance on digital health systems. Because of its immutability, decentralization and privacy protection features, blockchain technology has become a practical solution for healthcare domain applications. This section explains previous research on blockchain access control including methodology, benefits and limitations within the realm of healthcare security.

Román-Martínez, I., et al., (2023) integrates service-oriented architecture (SOA), blockchain, and business process management for healthcare auditing of access to health resources based on individual consent. Blockchain technology ensures non-repudiation and tamper-proof information, allowing supervisory authorities to examine protected resource access without healthcare organizations intervention. The implementation of a blockchain for a single configuration evaluated in Docker containers which restricts the generalizabillity of the findings.

With the use of side chains and the main blockchain, Yang, L., et al., (2024) provides a new mode for access control. The author design

password-based authentication and polygon side chain for improve blockchain storage scalability. The side chain stores the main Ethereum chain access node information, and Roll-up contracts are used to acquire resources on the side chain. However, the approach fails to address the difficulty of identifying the request type of Ethereum master chain nodes during instances of elevated concurrent access requests.

A blockchain based system for improving privacy in health diagnostics was described by Alsuqaih, H.N., et al., (2023). The framework provides a reliable method of controlling who can access what medical records by letting data owners choose their own preferences. Users are able to add or remove authorized doctors by using their transactions for key generation. Nevertheless, dependence on a local database may undermine decentralization.

Abutaleb, R.A. et al., (2023) develop a system for managing patient health data that guarantees patients can grant the requisite permission for access to their health information. The system will keep track of everything through blockchain technology and usage control. Nonetheless, blockchain may not be suitable for the storage of extensive health data.

Li, P., et al., (2024) suggests a policy hiding approach based on attribute-based cryptography and a consortium blockchain for privacy preserving access control. With this method, users of the data incur less computational burden as the blockchain handles ciphertext transformation and data retrieval, ensuring dependability and accuracy. Nevertheless, the solution fails to resolve the problem of revoked keys retaining access to blockchain data.

Kaur, J., et al., (2024) presents a blockchain-based secure record-keeping and reliable sharing system. The author employs the Cipher-text Policy Attribute-Based Encryption (CP-ABE) algorithm and has developed Smart Contracts for precise access control, permitting only authorized users to view particular EHR records according to their attributes. IPFS distributed storage is utilized to retain encrypted patient records and hashes on the blockchain. This approach has high latency and diminished throughput.

To guarantee the confidentiality, availability, security and integrity of medical data, Jakhar, A.K et al., (2024) introduces a blockchain-based system that both controls access and preserves privacy via consensus-driven decentralized data management. Due to its immutability and lack of reversibility, blockchain aids in safeguarding transactions from manipulation. The hyperledger fabric and composer were used to create the permissioned blockchain. One problematic aspect is the high expenses associated with response time.

Tawfik, A.M. et al., (2025) suggest a blockchain-based system that uses hyperledger fabric to facilitate transparent and decentralized access control. It incorporates IPFS for decentralized storage and guarantees anonymity using hyperledger fabric channels. The primary constraints of this approach is its relies on hyperledger fabric, which is only accessible within a permissioned network. This improves security, control, and privacy, but may hinder compatibility with open, decentralized blockchain-based healthcare system.

Shi, G., et al. (2025) present a blockchain-based access control system that integrates attribute-based access control (ABAC) and role-based access control (RBAC). It utilizes four smart contracts on the blockchain to facilitate decentralized and adaptable data access and sharing in IoT-based medical settings. It emphasizes that ABAC necessitates greater computational resources and effort for implementation compared to RBAC, owing to its intricate policy logic.

Guo, Z. (2025) proposes a decentralized methodology that utilizes blockchain technology and smart contracts to guarantee security and trust. It standardizes smart contract behaviour through Transition Systems and Computation Tree Logic, facilitating the verification of essential characteristics. The Solidity code is converted into the Verds model-checking tool, which verifies the accuracy and security of the contracts.

Wang, K., et al., (2026) propose a blockchain-based, privacy-preserving access control method that amalgamates an optimized hidden policy CP-ABE with multiplicative homomorphic encryption to provide secure and efficient data sharing in digital twin systems. It inhibits the disclosure of sensitive information while facilitating safe key creation for CP-ABE decryption. Table 1 shows the research gap analysis.

Table 1 Research gap analysis

Methodology used in the related papers	Author & Year	Access Control Polices	Smart Contract	Trust	Behaviour	IPFS
SOA–Blockchain–Business Process Management based	Román-Martínez, I., et al., (2023)	×	✓	×	×	×

healthcare auditing and consent management						
Blockchain-based privacy-preserving E-health access control with patient-controlled authorization	Alsuqaih, H.N. et al., (2023)	✓	×	×	×	×
Patient-centric blockchain-based health record access control with usage control	Abutaleb, R.A. et al., (2023)	×	✓	×	×	×
Blockchain master–sidechain collaboration with password authentication and Polygon sidechain	Yang, L., et al., (2024)	×	✓	×	✓	×
Blockchain-based flexible and secure EHR access control using attribute-based cryptography	Li, P., et al., (2024)	✓	×	×	✓	×
CP-ABE based blockchain and IPFS framework for secure EHR storage and sharing	Kaur, J., et al., (2024)	✓	✓	×	×	✓
Permissioned Hyperledger Fabric-based blockchain framework for privacy-preserving EHR access control	Jakhar, A. K., et al., (2024)	×	✓	×	×	×
Hyperledger Fabric and IPFS-based decentralized healthcare access control and privacy framework	Tawfik, A. M., et al., (2025)	×	✓	×	×	✓
ABAC and RBAC integrated blockchain-based dynamic access control for IoMT	Shi, G., et al., (2025)	✓	✓	×	×	✓
Blockchain and smart-contract based formal verification of IoT access-control mechanisms	Guo, Z. (2025)	×	✓	×	✓	×
Optimized hidden-policy CP-ABE with multiplicative homomorphic encryption for blockchain-based access control	Wang, K., et al., (2026)	✓	✓	×	×	×
Proposed SCDTACBC: Four smart contracts with dynamic trust-based access control and IPFS		✓	✓	✓	✓	✓

The majority of the prior work on blockchain focused on its potential for decentralized policy management, immutable and thus trustworthy data storage and the use of smart contracts to support automatic access control assessment. Numerous previous methods establish access control polices based on user identities. Static policies are ineffective in an open and dynamic world that necessitates adaptive access control policies and decentralized autonomous authorization management. Incorporating dynamic access control techniques could offer significant potential for adaptable authorization based on the trust and behaviors of the accessing entity and the historical evolution of the accessed object.

The previous solutions encounter issues including insufficient fine-grained access control, significant processing overhead, scalability limitations, latency and reliance on centralized components or external storage system. The suggested SCDTACBC strategy upholds a decentralized, permissioned framework that facilitates structured data management, resulting in high throughput and low latency. Additionally, SCACBCC reduces storage overhead through the integration of IPFS, hence improving data availability and privacy. This combination guarantees that sensitive data is secure and available solely to authorized parties, while facilitating effective access control.

PROPOSED METHODOLOGY

This section explains the proposed SCDTACBC approach which employs blockchain technology to create safe and decentralized access control management within the healthcare sector. Figure 1 shows the general framework of SCDTACBC.

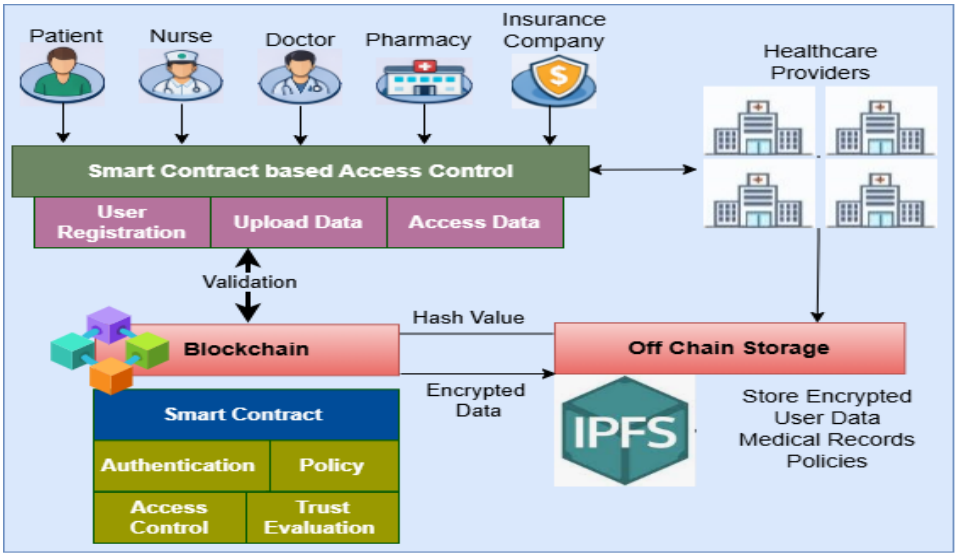


Figure 1 SCDTACBC General Framework

SYSTEM MODEL

The proposed architecture contains different types of entities: Participants, Blockchain and Offchain storage.

Participants: The system has different kinds of participants: Healthcare providers, Patient, Nurse, Doctor, Pharmacy, and Insurance company. Each participant plays diverse role in SCDTACBC. The healthcare provider collects and encrypts patient information and assign doctor. The other participants access patient information based their access polices.

Let U = {u_1, u_2, u_3, ..., u_n} be the set of participants. R = {Patient, Doctor, Nurse, Pharmacist, Insurance} be the set of participant roles. O = {o_1, o_2, o_3, ..., o_n} be the healthcare objects (patient information including personal, and disease details). Each participant is represented by, u_i=(Role$_i$, UserName$_i$, Password$_i$, Token$_i$). When a participant u_i wants to access the healthcare objects o_i, the access decision is denoted by,

$$Access\ (u_i, o_i) = \begin{cases} Grant, & Auth(u_i) = True\ \wedge Pol = True\ \wedge T \geq T_{th} \\ Deny, & Otherwise \end{cases} \quad (1)$$

Where Auth (u_i) is authentication, Pol indicates policy, T denotes Trust value and T_{th} represents trust threshold.

The Pol (policy) associates each user role with a designated set of permissible actions, such as read, or write.

The trust (T) can be computed based on the user behaviour metric which includes correctness, availability and malicious.

The detailed description of authentication, policy and trust value can be explained in the following subsection.

Blockchain: It includes decentralized ledger, smart contract, and consensus protocol. Every authentication and access request event is recorded on the distributed ledger indefinitely. With the goal of protecting user privacy, the blockchain only stores hashed values and minimal information. This research work uses four smart contracts. With their own distinct rules and functionalities, smart contracts encapsulate particular business logic that guarantees automation, security, and compliance with access control restrictions.

For distributed systems, such as blockchain networks, the consensus protocol is an essential tool for determining the authenticity and sequence of events or transactions among numerous nodes. The objective is to guarantee that all network participants achieve a uniform state and concur on state modifications. This method utilizes the Proof-of-Authority (PoA) consensus technique. It ensures that solely the admin node authenticates and verifies transactions on the blockchain. This consensus method reduces computation overhead and enhances scalability.

Offchain Storage: The suggested work use IPFS distributed storage (offchain) to store

medical data instead of blockchain (onchain). In order to comply with data privacy regulations and decrease blockchain storage demands, it safely keeps sensitive medical data. Storage on the blockchain includes hash references that point to data stored offchain. To ensure the patient privacy, the medical is encrypted using the symmetric key technique before being saved in IPFS.

SMART CONTRACT

Smart contract offers automated legal entitlements, operational logic, and obligations, establishing a foundation for system security and privacy safeguards while enhancing system efficiency. Smart contracts have the ability to implement various privacy protection measures according to the degree of private information that each user has submitted.

Without relying on a central authority, smart contracts can be used between anonymous and untrusted individuals. The blockchain facilitates smart contracts by creating links between contracting parties using a local programming language, wherein the execution of contract stipulations and conditions is stored on the blockchain as an immutable transaction. Consequently, a related action may be carried out automatically whenever a smart contract condition is satisfied. The smart contract mechanisms comprise a collection of operations that enable the smart contract to communicate with both other contracts and the blockchain.

Smart contracts can facilitate blockchain-based solutions for access control. By utilizing smart contracts, rules may be programmed to impose user-or group specific access control regulations. The rules can specify which users on the blockchain can see which data and execute which operations. Smart contracts allow blockchain-based system to restrict access to sensitive data and processes so that only authorized users can access them.

In this paper, four smart contracts are created for secure access control which includes authentication, policy, access control and trust evaluation. The authentication contract specifies procedures for validating participation, the policy contract handles the administration of polices, the trust evaluation contract calculates a trust score from observed behaviour, and the access control governs the procedure for controlling access.

Authentication Contract (auth_sctr):

The auth_sctr checks user identities to ensure that only authorized individuals have access to the system. During login, it verifies credentials including username, password and created 8-bit token. If the supplied information corresponds with the archived data, the user is authorized and permitted to advance; if not access is refused and the system logs the erroneous activity. This contract enhances the trust model by constantly monitoring user behaviour metrics. When logins are successful, trust is increased, when attempts are unsuccessful, trust is decreased. This ensures that identity verification is both secure and behaviour-aware.

The authentication contract contains two methods user registration and authentication. In user registration, the user provides username and password. The system generates 8-bit token and hash value for each user. In user authentication, the registered users are verified based on the password and token. The objective of this contract is to validate user details and update user behaviour. It ensures that only authorized users are allowed to continue, and their behaviour is continuously monitored for potential future trust assessments.

The user authentication process is presented in Algorithm-1.

Algorithm 1

auth_sctr.UserAuthentication()

Input: UserName U_{name}, Password U_{pass}, Token U_{token}
Output: auth_status
1: Fetch user_records
2: If user_records exists then
3: If U_{pass} == user_records.P_i and U_{token} == user_records.T_i then
4: auth_status = TRUE
5: teval_sctr.Update(correct=TRUE)
6: Else
7: auth_status = FALSE
8: eval_sctr.Update(correct=FALSE)
9: EndIf
10: Else
11: auth_status = FALSE
12: EndIf

13: Return auth_status

The user authentication is success only if all credentials (password and token) match.

$$Auth(u_i) = \begin{cases} True, & U_{pass} = P_i \wedge U_{token} = T_i \\ False, & Otherwise \end{cases} \quad (2)$$

Where P_i and T_i indicate stored password and token (8-bit length).

Policy Contract (pol_sctr):

The policy smart contract delineates and regulates role-based permissions within the system. It associates each user role with a designated set of permissible actions, such as read, or write. Based on the role that has been assigned to a user, this

contract functions as a rule engine, determining what actions are allowed. Its scalability and flexibility are guaranteed by decoupling policy logic from other components. Administrators can simply adjust access rules without having to modify the entire system.

The pol_sctr contains two methods: add/Update policy and validate policy. In add policy method, the healthcare provider, add action read or write for the particular user role. The validate policy method check whether the particular user has permission to access the data. Algorithm-2 explains pol_sctr.

Algorithm-2 Pol_sctr

Add / Update Policy

Input: User Role U_{role}, User Action U_{action}, Permission List permission_lt

Output: Policy Status pol_status

1: If permission_lt is empty then

2: Return Invalid Input

3: EndIf

4: If U_{role} exists in Policy then

5: Update permission_lt for the U_{role} and U_{action}

6: pol_status = POLICY UPDATED

7: Else

8: Insert U_{role} and U_{action} with permission_lt into Policy

9: pol_status = POLICY ADDED

10: EndIf

11: Return pol_status

Validate Policy

Input: User Role U_{role}, User Action U_{action}

Output: Permission Status per_status

1: Get permission_lt for U_{role}

2: If U_{action} ε permission_lt then

3: per_status = TRUE

4: Else

5: per_status = FALSE

6: EndIf

7: Return per_status

In order to proceed with further processing, the validate policy gives a positive result (permission status = TRUE) if the requested action is deemed to be within the role's permissions; otherwise, the request will not be granted.

Trust Evaluation Contract (teval_sctr):

A dynamic trust score is calculated by the trust management smart, which influences access decisions based on continuous monitoring of user behaviour. There are three main criteria that are assessed: the correction of transactions (successful activities), the consistency of participation (availability), and the presence of malicious behaviour (unauthorized or harmful actions). It employs a weighted method to update the trust score. The system becomes more resilient thanks to adaptive security, which is made possible by this contract. Users who behave well are given more trust and more access, while those who behave riskily are given less trust and fewer privileges. The teval_sctr contains two methods trust score computation and behaviour updation.

All users trust score is computed and updated by teval_sctr depending on their actions over time. It extracts behavioural parameters including correct interaction, total interaction, activity session, total session, and malicious activities. Based on these parameters, it computes correctness ratio, availability ratio and malicious ratio. To compute total trust score, combine these ratios via weighted method. This score is used by access control contract for making decision. This evaluation is designed to be dynamic (which continuously updates trust), so it can adjust to changes in user behaviour.

The trust score can be computed as follows:

The correctness (C) indicates the successful interactions. It can be computed as,

$$C = \frac{correct}{total} \tag{3}$$

Where correct indicates the number of successful interactions and total represents total number of interactions.

The availability (A) represents how the user is during the observation period. It can compute using,

$$A = \frac{online_{session}}{total_{session}} \tag{4}$$

Where $online_{session}$ and $total_{session}$ represents the active session duration and total observation time.

The malicious behaviour(B) indicates unauthorized and harmful action of the user. It can be calculated as,

$$M = \frac{malicious}{total} \tag{5}$$

Where malicious represents number of unauthorized activities and total indicates total number of activities.

The overall trust can be calculated using weighted model.

$$T = (\omega_1 * C) + (\omega_2 * A) + (\omega_3 * (1 - M)) \quad (6)$$

Where $\omega_1 + \omega_2 + \omega_3 = 1$.

The weights are assigned as

$$\omega_1 = 0.4, \omega_2 = 0.3, \omega_3 = 0.3$$

In this trust computation, the malicious behaviour (M) is converted into non-malicious behaviour (1-M). The lower malicious behaviour increases trust and high malicious behaviour decrease trust. The example for trust calculation is as follows:

C	correct = 4	total=5	$\frac{4}{5} = 0.8$
A	online =32	total=290	$\frac{32}{290} = 0.11$
M	malicious=2	Total = 5	$\frac{2}{5} = 0.4$
T	(0.4*0.8) + (0.3*0.11) + (0.3*(1-0.4))		**Trust=0.533**

The trust (T) value can be lies between 0 to 1. The higher value indicates higher reliability.

The behaviour update method assists trust management by accurately recording user activities. When a user logs in or access data, it updates the correct and malicious count depending on the action validation. This ongoing feedback system guarantees that trust values are up-to-date and accurately represent actual user behaviour. Algorithm-3 explains teval_sctr.

Algorithm-3 teval_sctr

Compute Trust Score

Input: User Name U_{name}

Output: Trust Score $trust_{scr}$

1: Get behaviour metrics: correct, total, malicious, active_session, total_session.

2: Compute correctness using Eq. (3)

3: Compute availability using Eq. (4)

4: Compute malicious using Eq. (5)

5: Compute Trust Score using Eq. (6)

6: update $trust_{scr}$ for U_{name}

7: Return $trust_{scr}$

Update

Input: User Name U_{name}, Correct status corr_status, Malicious status mal_status

1: Get current behaviour metrics

2: If corr_status == TRUE then

3: correct = correct +1

4: EndIf

5: If mal_status == TRUE then

6: malicious = malicious +1

7: EndIf

8: total = total +1

9: Store update values

10: teval_sctr.computeTrustScore(U_{name})

Access Control Contract (actrl_sctr):

Secure data access is enforced by the access control smart contract through the combination of role-based polices and trust evaluation. This contract initially examines the policy contract to see if a user can access patient data within their position. Following this, it assesses the user's present trust score, which is calculated in real-time using measures including availability, correctness and harmful activity. The system is designed to ensure that even authorised users can be restricted if their behaviour becomes suspicious by requiring both legitimate authorization and adequate trust before granting access. Algorithm-4 explains the actrl_sctr.

Algorithm-4 actrl_sctr

Input: User Name U_{name}, User Role, U_{role}, User Action U_{action}, Threshold thr_value

Output: Access Status acc_status

1: per_status = pol_sctr.validatepolicy(U_{name}, U_{role})

2: If per_status == FALSE Then

3: teval.sctr.Update(U_{name}, correct=FALSE, malicious = TRUE)

4: acc_status = DENIED

5: Return acc_status

6: EndIf

7: $trust_{scr}$ = teval.computeTrustScore($_{Uname}$)

8: If $trust_{scr} \geq$ thr_value

9: teval.sctr.Update(U_{name}, correct=TRUE, malicious = FALSE)

10: acc_status =GRANTED

11: Else

12: teval.sctr.Update(U_{name}, correct=FALSE, malicious = FALSE)

13: acc_status =DENIED

14: EndIf

15: Return acc_status

This contract act as a decision making that decides whether a user is allowed access to a particular data. Initially, it executes pol_sctr to validate and get the permission status of user. If the validation fails, the system promptly refuses access and categories the action as malicious and updates the behaviour. Following a successful validation, it computes the trust score from teval_sctr. When the trust score reaches or surpasses the threshold is access allowed; otherwise, access is prohibited. Even authorised users cannot access data if their behaviour is suspect.

The four smart contracts work together as an integrated access control mechanism to provide secure and dynamic healthcare data sharing. The authentication contract then checks the identity of the participant using the registered credentials and change the user behaviour according to the result of the authentication. Once authenticated, the policy contract is responsible for determining whether the action the user wishes to carry out is allowed based on the role the user has. The trust evaluation contract continually assesses the user's behaviour based on correctness, availability and malicious behaviour to calculate a trust score that changes over time. Last, the access control contract can take the policy validation and trust score along with the defined trust score to determine the final access decision. The four smart contracts, therefore, enable a comprehensive workflow starting with the authentication of the user, validation of the policy, evaluation of trustworthiness and granting access rights safely.

Moreover, this method is designed to allow for flexible access decisions over time, depending on user actions. It also increases system resilience through operation, by identifying and mitigating malicious or abnormal activities. Additionally, running several smart contracts increases the modularity, thus making maintenance and updates of the system easier without impacting the overall structure. This overall workflow shows the sequential interaction between the four smart contracts, starting with login and then moving on to policy validation and trust evaluation. Depending on the policy permission and the trust level, the system denies or allows the encrypted healthcare data stored in IPFS to be viewed. If access is approved, the encrypted data is retrieved from IPFS, decrypted, and then shown to the approved user. The workflow also captures misbehaviours, incorrect actions and bad trust, to enable ongoing trust assessment and adaptive access decisions. The overall flow of the smart contract algorithm is illustrated in the figure below (Figure 2).

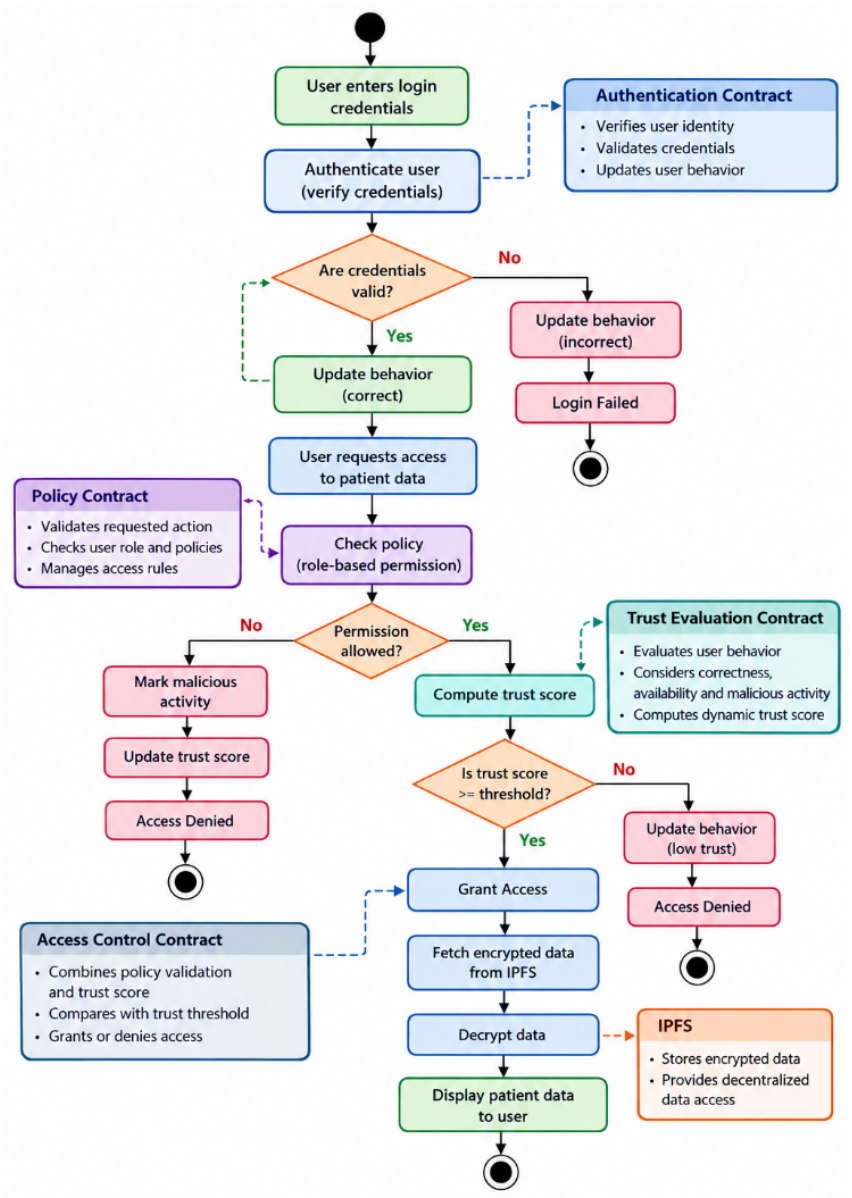


Figure 2 Overall workflow of SCDTACBC

RESULTS AND DISCUSSION

This section provides the simulation settings and an evaluation of the proposed SCDTACBC. To evaluate the suggested access control system a comprehensive simulation environment is created. Table 2 shows the simulation setting.

Table 2 Simulation Setup

Hardware	***Configuration***
OS	Windows10
CPU	Intel Pentium 2.30GHz
Memory	4 GB RAM
Software	
Visual Studio Code	1.113.0
Node js	V25.8.2
Python	3.10.7
IPFS	0.46.1

The visual studio code, node js and python is used to create a web-based access control system. The IPFS is used to store the encrypted medical information.

The metrics throughput and latency are used to analyze the performance of the proposed approach. The throughput can be computed using,

$$Throughput = \frac{|Tx|}{tt} \quad (7)$$

Where Tx is the total transaction count and tt indicates the total time to complete the process.

Latency is the time required to handle the process and it can be computed as,

$$Latency = \frac{e_t - s_t}{|Tx|} \quad (8)$$

Where e_t and s_t denotes the end time and start time of the process.

Table 3 shows the shows the comparison of throughput and latency of each smart contract in SCDTACBC for number of transactions is 100. The throughput ranges from 5.28 to 50 TPS (Transaction per Second) and latency from 0.02 to 0.189 seconds.

Table 3 Throughput and Latency Comparison of each smart contract in SCDTACBC

Smart Contract	Throughput (TPS)	Latency (Sec.)
auth_sctr	5.57	0.179
pol_sctr	50	0.02
teval_sctr	45.35	0.022
actrl_sctr	5.28	0.189

Figure 3 and Figure 4 shows the throughput and latency. The throughput for pol_sctr (50) and teval_sctr (45.35) has higher value compared to auth_sctr (5.57) and actrl_sctr (5.28). Because, the auth_sctr and actrl_sctr execute teval_sctr contract which take more time to update the trust value. Similarly, the auth_sctr (0.179) and actrl_sctr (0.189) has higher latency compared to pol_sctr (0.02) and teval_sctr (0.022).

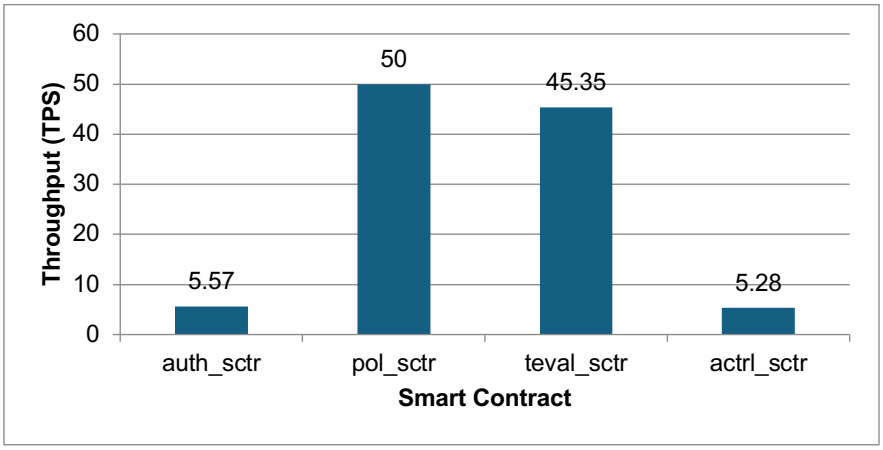


Figure 3 Throughput for SCDTACBC

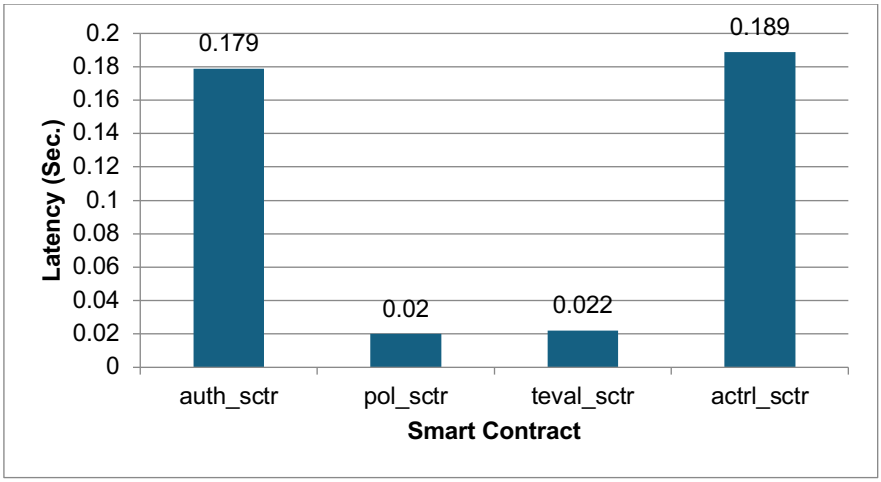


Figure 4 Latency for SCDTACBC

The performance of the proposed approach is compared with Ucbas, Y., (2023), Li, P. (2024) and Tawfik, A. M., (2025) in terms of latency and throughput.

Table 4 and Figure 5 shows the comparison of throughput. The proposed approach has 65.45, 63.74, 64.03, and 64.5 throughput for 100, 500, 1000, and 5000 transactions.

Table 4 Comparison of Throughput

No of Trans actions	Ucbas, Y., (2023)	Li, P., (2024)	Tawfik, A. M., (2025)	Proposed
100	34.75	47.4	51.3	**65.45**
500	47	48.6	50.5	**63.74**
1000	49.9	50.2	50.8	**64.03**
5000	49.9	50.4	50.9	**64.5**

The proposed approach increases 88.34%, 38.08% and 27.58% of throughput compared to Ucbas, Y., (2023), Li, P., (2024) and Tawfik, A. M., (2025) approaches.

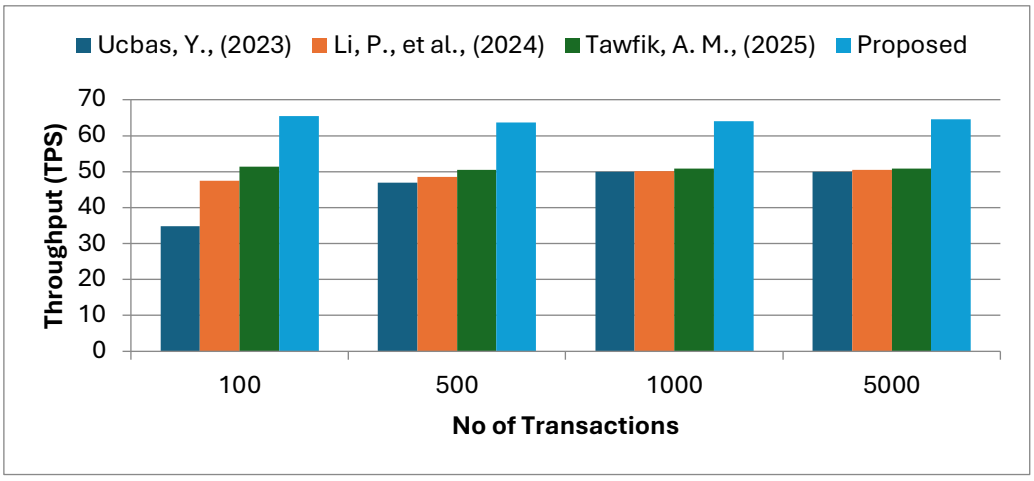


Figure 5 Throughput Comparison

Table 5 shows the comparison of latency. The proposed approach has 0.018, 0.017, 0.016 and 0.012 sec. latency for 100, 500, 1000, and 5000 transactions.

Table 5 Comparison of Latency

No of Trans	Ucbas, Y., (2023)	Li, P., (2024)	Tawfik, A. M., (2025)	**Proposed**

100	1.9	1.6	0.03	**0.018**
500	0.6	0.5	0.02	**0.017**
1000	0.4	0.31	0.02	**0.016**
5000	0.73	0.65	0.01	**0.012**

Figure 6 shows the comparison of latency. The proposed approach decreases 90.05%, 89.87%, and 40% of latency compared to Ucbas, Y., (2023), Li, P., (2024) and Tawfik, A. M., (2025) approaches.

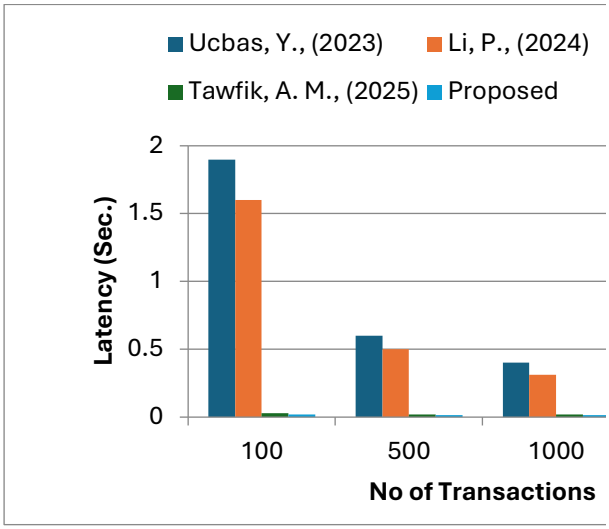


Figure 6 Latency Comparison

From these results it was observed that the proposed SCDTACBC achieves high throughput and low latency compared to other existing methods.

The suggested method guarantees that data integrity is maintained throughout transmission and storage. It protects sensitive healthcare information from unauthorized access and guarantees that only authorized entities can access or alter it. At the same time the proposed SCDTACBC has some limitation including computational complexity due to repeatedly computing and updating the trust value, the improper selection of weight can lead to over or under user trustworthiness and the model is only suitable healthcare domain.

CONCLUSION AND FUTURE WORK

This research presents SCDTACBC, a smart based access control using blockchain. By using blockchain technology's scalability and immutable transactions, the proposed work is better to handle the demands of the healthcare applications. The suggested approach creates four smart contracts (auth_sctr, pol_sctr, teval_sctr and actrl_sctr); these contracts improve security and privacy without depending the need for centralized authorities. According to the results, the proposed SCDTACBC outperforms the existing approaches in terms of latency and throughput. The proposed approach attains higher throughput (65.45 TPS) and also lower latency (0.018 Sec.) for 100 transactions. The SCDTACBC approach increases 27.58% of throughput and decreases 40% latency compared to other existing methods. The scalability analysis verifies its effectiveness in the growing of workloads, which IPFS integration improves storage management and security. Its robustness is further bolstered by security evaluations, which emphasizes resilience against typical attacks. The system adaptability is enhanced when one user can grant access to other users to utilize the service. Consequently, numerous difficulties about security and trust will emerge.

Further studies will be directed towards improving the proposed SCDTACBC framework by proposing intelligent mechanisms for smart-contract enabled blockchain threat and attack detection. The future system can be put together to accommodate intelligent threat detection techniques, which recognize abnormal and/or malicious user activity as it occurs when accessing the system. It can further analyze behavioural patterns to detect potential attacks at an early stage. Detected threats may also be used to dynamically adjust the trust score and prevent suspicious users from having access to sensitive healthcare information. This method can help the framework be better equipped to meet changing security threats. Therefore, the implementation of intelligent threat and attack detection and dynamic trust-based access control will be an essential way to enhance the security and robustness of the proposed framework.

ACKNOWLEDGEMENT

The author thanks DST-FIST, Government of India for funding towards infrastructure facilities at St. Joseph College (Autonomous), Tiruchirappalli - 620002

Source of Support: None

Conflict of Interest: None

REFERENCES

Abdi, A. I., Eassa, F. E., Jambi, K., Almarhabi, K., & Al-Ghamdi, A. S. A. M. (2020). Blockchain platforms and access control classification for IoT systems. Symmetry, 12(10), 1663.

Abutaleb, R.A., Alqahtany, S. S., & Syed, T. A. (2023). Integrity and privacy-aware, patient-

centric health record access control framework using a blockchain. Applied Sciences, 13(2), 1028.

Alsuqaih, H. N., Hamdan, W., Elmessiry, H., & Abulkasim, H. (2023). An efficient privacy-preserving control mechanism based on blockchain for E-health applications. Alexandria Engineering Journal, 73, 159-172.

Azbeg, K., Ouchetto, O., & Andaloussi, S. J. (2022). BlockMedCare: A healthcare system based on IoT, Blockchain and IPFS for data management security. Egyptian informatics journal, 23(2), 329-343.

Guo, Z. (2025). Blockchain-enhanced smart contracts for formal verification of IoT access control mechanisms. Alexandria Engineering Journal, 118, 315-324.

Jakhar, A. K., Singh, M., Sharma, R., Viriyasitavat, W., Dhiman, G., & Goel, S. (2024). A blockchain-based privacy-preserving and access-control framework for electronic health records management. Multimedia Tools and Applications, 83(36), 84195-84229.

Kamboj, P., Khare, S., & Pal, S. (2021). User authentication using Blockchain based smart contract in role-based access control. Peer-to-Peer Networking and Applications, 14(5), 2961-2976.

Karimi, L., Aldairi, M., Joshi, J., & Abdelhakim, M. (2021). An automatic attribute-based access control policy extraction from access logs. IEEE Transactions on Dependable and Secure Computing, 19(4), 2304-2317.

Kaur, J., Rani, R., & Kalra, N. (2024). Attribute-based access control scheme for secure storage and sharing of EHRs using blockchain and IPFS. Cluster Computing, 27(1), 1047-1061.

Lee, J. S., Chew, C. J., Liu, J. Y., Chen, Y. C., & Tsai, K. Y. (2022). Medical blockchain: Data sharing and privacy preserving of EHR based on smart contract. Journal of Information Security and Applications, 65, 103117.

Li, P., Zhou, D., Ma, H., & Lai, J. (2024). Flexible and secure access control for EHR sharing based on blockchain. Journal of Systems Architecture, 146, 103033.

Mpamugo, E., & Ansa, G. (2024). Enhancing network security in mobile applications with role-based access control. Journal of Information Systems and Informatics, 6(3), 1872-1899.

Preetha, A. D., & Kumar, T. P. (2023). Securing IoT-based healthcare systems from counterfeit medicine penetration using Blockchain. Applied Nanoscience, 13(2), 1263-1275.

Román-Martínez, I., Calvillo-Arbizu, J., Mayor-Gallego, V. J., Madinabeitia-Luque, G., Estepa-Alonso, A. J., & Estepa-Alonso, R. M. (2023). Blockchain-based service-oriented architecture for consent management, access control, and auditing. IEEE Access, 11, 12727-12741.

Sharma, P., Jindal, R., & Borah, M. D. (2023). A review of smart contract-based platforms, applications, and challenges. Cluster Computing, 26(1), 395-421.

Shi, G., Qi, M., Zhong, Q., Li, N., Gao, W., Zhang, L., & Gao, L. (2025). MEDACCESSX: A Blockchain-enabled dynamic access control framework for IoMT networks. Sensors, 25(6), 1857.

Sigwart, M., Borkowski, M., Peise, M., Schulte, S., & Tai, S. (2024). A secure and extensible blockchain-based data provenance framework for the Internet of Things. Personal and Ubiquitous Computing, 28(1), 309-323.

Tawfik, A. M., Al-Ahwal, A., Eldien, A. S. T., & Zayed, H. H. (2025). ACHealthChain blockchain framework for access control and privacy preservation in healthcare. Scientific Reports, 15(1), 16696.

Ucbas, Y., Eleyan, A., Hammoudeh, M., & Alohaly, M. (2023). Performance and scalability analysis of ethereum and hyperledger fabric. IEEE Access, 11, 67156-67167.

Wang, K., Zhang, N., Wu, A., & Han, R. (2026). Blockchain-Assisted Privacy-Preserving and Secure Access Control for IoT-Enabled Digital Twin. Blockchain: Research and Applications, 100450.

Xiang, X., Cao, J., & Fan, W. (2022). Decentralized authentication and access control protocol for blockchain-based e-health systems. Journal of network and computer applications, 207, 103512.

Yakubu, B. M., Ali, S. M., Khan, M. I., & Bhattarakosol, P. (2024). PatCen: A blockchain-based patient-centric mechanism for the granular access control of infectious disease-related test records. Plos one, 19(9), e0310407.

Yang, L., Jiang, R., Pu, X., Wang, C., Yang, Y., Wang, M., Zhang, L. and Tian, F., (2024). An access control model based on blockchain master-sidechain collaboration. Cluster Computing, 27(1), 477-497.

Zhang, L., Li, B., Fang, H., Zhang, G., & Liu, C. (2023). An internet of things access control scheme based on permissioned blockchain and edge computing. Applied Sciences, 13(7), 4167.