International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Cloud Computing Access Control Using Blockchain

Manal Ayyadh Alshammari[1], Hedi Hamdi[1,2], Mahmood A. Mahmood[1,3], A. A. Abd El-Aziz[1,3]

Submitted: 12/10/2023 **Revised**: 11/12/2023 **Accepted**: 22/12/2023

Abstract: Access control is a pivotal aspect of modern computing systems, ensuring that only authorized entities can interact with sensitive data and resources. Traditional access control methods, while effective, face challenges in terms of security, transparency, and scalability. Blockchain technology has emerged as a promising solution to enhance access control in cloud computing environments. Blockchain's inherent features, such as decentralization, immutability, and cryptographic security, offer a unique framework for addressing the limitations of traditional access control systems. In this paper, we explore the integration of blockchain technology with cloud computing to establish a more secure and transparent access control mechanism. By utilizing blockchain, access control policies and permissions can be stored in a tamper-proof and transparent ledger. Scalability is an issue since processing many access control transactions on the blockchain might cause network congestion and sluggish processing. Transactions take time to upload to the blockchain, which can delay real-time access choices. It takes skill to integrate and manage blockchain and cloud technologies together. Choosing the correct consensus mechanism affects system efficiency and security. Consider the costs of establishing and maintaining such a system and the difficulty of fixing faults owing to blockchain immutability. In conclusion, this paper underscores the significance of access control in cloud computing and the limitations of traditional approaches. By harnessing the power of blockchain technology, a more secure, transparent, and scalable access control framework can be established, revolutionizing the way we manage access to sensitive data and resources in complex digital ecosystems.

Keywords: *Blockchain, Cloud Computing, Processing System, Network Access*

1. Introduction

In today's digital landscape, cloud computing has emerged as a transformative paradigm that reshapes the way organizations manage and deliver their IT resources. Cloud computing offers unparalleled scalability, flexibility, and cost-efficiency by providing on-demand access to a wide array of computing services, ranging from storage and processing power to software applications and networking resources. This shift from traditional on-premises infrastructure to cloud-based solutions has enabled businesses to streamline operations, enhance collaboration, and rapidly respond to dynamic market demands. However, as the adoption of cloud computing continues to rise, so do the challenges associated with ensuring the security and controlled access to the vast repositories of data and services hosted in the cloud [1].

Access control stands as a cornerstone of information security in cloud computing environments. It encompasses the mechanisms and policies that regulate the entry, interaction, and usage of resources and data within a cloud infrastructure. Various access control methods have been employed to safeguard sensitive information, including role-based access control (RBAC), attribute-based access control (ABAC), and discretionary access control (DAC). While these methods have proven effective in many scenarios, they each carry inherent limitations [2].

Role-based access control (RBAC) assigns permissions based on predefined roles, simplifying management but potentially resulting in overly broad access for some users and complexities in defining roles for different contexts. Attribute-based access control (ABAC) grants access based on attributes and conditions, allowing for more fine-grained control, but its complexity can hinder ease of implementation and management. Discretionary access control (DAC) permits resource owners to decide who can access their resources, but this can lead to inconsistencies and difficulties in enforcing consistent security policies [3].

[1] *College of Computer and Information Sciences, Jouf University, KSA*
[2] *University of Manouba., Tunisia*
[3] *Faculty of Graduate Studies for Statistical Research, Cairo University, Egypt*
** Corresponding Author aaeldamarany@ju.edu.sa*

As cloud environments grow in complexity, traditional access control methods face challenges in maintaining the desired level of security, transparency, and scalability. This is where the convergence of blockchain technology with cloud computing emerges as a compelling solution to address these limitations and elevate access control mechanisms to a new level of robustness and efficiency. In the subsequent sections, we will delve into how blockchain can revolutionize access control in cloud computing, offering enhanced security, transparency, and adaptability in an increasingly interconnected digital landscape [4].

With the great development in information technology, storing data in huge quantities and sharing it through cloud computing has become easy. Cloud computing represents a low-cost storage method for many users that gives them access to data. Additionally, this technology can be deployed in four models: community, private, hybrid, and public clouds [5]. These deployment methods vary in characteristics impacting how businesses in society apply them. For instance, NASA deployed private and hybrid clouds in their first open-source software in 2008 [6]. This combination of deployment methods highlights the flexibility of this technology which propels its incorporation in various industries. Generally, this technology's diverse application methods allow its users to enjoy its various advantages and have propelled its incorporation in various sectors of society.

With the estimated continuation of this development, the concept of sequential blocks for storing data emerged in the Blockchain approach. As an advancement in digital informatics, the blockchain field has provided ample solutions to address the flexibility issues in sharing digital data. The hashes, which are among the primary components of this technology, enable it to adapt to different use cases effectively and to present any size of data in the form of a block [7], highlighting its primary use advantages. Additionally, one of the notable characteristics of this technology is its code, which is an open source for everyone with access, eradicating the chances of creating a back door within the system [8]. Therefore, this technology has the capability to raise the degree of safety in information sharing in business with various application areas such as commercial, financial, sales export, and banking transactions. Generally, cloud computing and blockchain are interfaced, posing advantages that can be leveraged in the economy if utilized together.

However, incorporating blockchain into cloud computing is only done through two methods. Blockchain can be incorporated into this traditional technology to facilitate business networks such as replication, storage, and transactional base access [7]. Cloud computing is a technical means for storing data to be controlled remotely that will benefit from the network facilitation of blockchain. Additionally, blockchain can be incorporated with other cloud security concepts between user, task, and data management [7]. This showcases its identity and access management capabilities in this sector. There are transparency issues with cloud computing, such as users' lack of management of data use and movement within this system, which can be addressed by leveraging blockchain data security capabilities [9]. Hence, blockchain can be leveraged to address the limitations of cloud computing and enhance its performance. Furthermore, these two technologies have been applied in diverse areas and environments [9], making it crucial to leverage their advantages for the effectiveness of these areas. Therefore, leveraging these technologies create a more secure approach to controlling access to data and identity in cloud computing.

A. Blockchain backing for distributed computing

Blockchain joining with distributed computing carries us into the following time of information security and administration accessibility. Blockchain defeats the majority of the exploration issues of the cloud with its qualities.

B. Interoperability

In broad daylight mists, inside correspondence isn't permitted, and it makes numerous ventures ease off from utilizing the cloud. At the point when cloud incorporated with blockchain, think about the various mists as hubs. Between hub correspondence is conceivable in the blockchain. Every one of the hubs present in a similar organization divide the information between themselves with the goal that each hub contains a duplicate of exchanges. It carries us straightforwardness into the organization. They update each next exchange into the record, which distributes to any remaining hubs. Along these lines, organizations can add quite a few organizations and can save the openness of the information, which carries credibility into the organization.

C. Data encryption

D. We all realize the information is decoded prior to putting away it in the cloud, which questions the information respectability. In the blockchain network, all the block information is transformed into a hash code utilizing cryptographic calculations, and it produces a hash key for each block. Allow us to consider a situation wherein blockchain is utilized to protect task planning for the cloud. To guarantee practicality and super durable information respectability, the control framework that gathers information from the assignment planning produces hash code and keeps it in the blockchain network right away. Since the blockchain has the office for block disclosure agreement components, block information trustworthiness is kept up with. Every hub in the organization contains a duplicate of every exchange that gives us the accessibility and tirelessness that assists the organization with enduring potential shortcoming focuses and assaults. While cloud-gathered information is solid, the blockchain hubs boost information accessibility and information legitimacy by extending it by extending it as an on-request administration with no downtime [10].

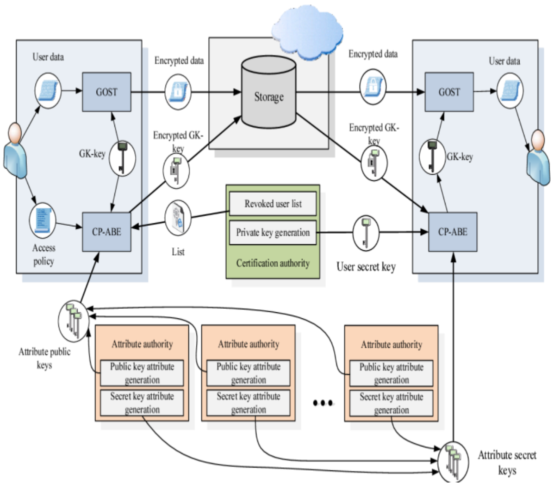


Fig. 1. A blockchain-based access control system for cloud storage.

E. Service level agreements

These arrangements in the cloud are good for the specialist co-op or client without equivalent equity. To address this issue, we can utilize blockchain savvy contracts. A brilliant agreement in blockchain assists with building trust between the gatherings who don't have the foggiest idea about one another. In a blockchain, Shrewd Agreements are characterized as a program written in programmable dialects that spat a compartment. The shrewd agreement permits self-execution when a particular condition is met on all hubs present in the blockchain network. It likewise assists the gatherings with foreseeing the results as the agreement execution relies upon the code accessible on a public organization, and they are certain as they are now marked.

F. Cloud data management

The information put away in the cloud is in an exceptionally unstructured way. The information put away in the blockchain is an extremely organized way. The information can be followed utilizing the hash key produced for each block. Each block contains the past block's hash key, and monitoring the network is vital. The information in the block is approved and can be gotten to by the hubs present in the organization. Cloud upholds versatility and can deal with the changes in computational burdens when required. Utilizing a Conveyed record, this can be effortlessly taken care of by dealing with countless occasions that cause different brilliant agreements, guaranteeing administration quality. Blockchain additionally guarantees the client's obscurity, and the client's record can be securely eliminated from the framework to forestall outsider admittance to the client's data. The combination of the cloud with blockchain will likewise guarantee that numerous organizations have certainty, and it would turn into an on-request service [11].

The client communicates with the server with the assistance of the application layer. Assume, when a client demands an exchange through the application layer, the exchanges' subtleties are put away by making a block for every exchange. To add the block into the blockchain network, the blockchain organization's information would be confirmed by blockchain network approving hubs. The approval will be done in light of agreement. When the block is viewed as authentic, any remaining organization hubs would be associated with the organization and information sent. The design of cloud coordinated with Blockchain is put away in blockchain assurance distributed storage. Blockchain fuse of the cloud gives information security, straightforwardness and furthermore improves services [12].

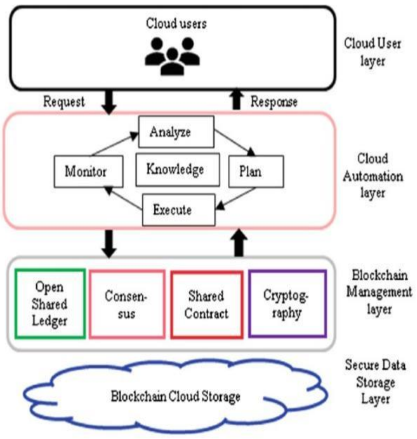


Fig. 2. Blockchain based Cloud framework.

2. Related Work

In [13], authors proposed a blockchain-based access control framework with privacy protection by using the account address of the node in the blockchain as an identity and redefining the access control permission of data for the cloud, which is encrypted and stored in the blockchain. All-access control, authorization, and DE authorization processes are through the AuthPrivacyChain.

In [9], it was found that blockchain can be a suitable and powerful tool for providing security in a cloud computing environment after analyzing the overall structure of the blockchain and the characteristics of the security requirements of blockchain and cloud computing. Cloud storage used by blockchain is accessible and open and can view all sorts of services provided by the users can view the sameversion/copy. Blockchain, coupled with smart contract technologies, enables more trust and transparency.

In [7], the authors provided a review of the application of blockchain in the cloud computing system by analyzing relevant previous papers and studies and accessing the gap in each study separately to reveal ideas that researchers and interested parties can refer to provide more secure solutions in the future. By reviewing previous studies based on the application of the blockchain in cloud computing, the researchers found that the proposed models lack more features and suffer from data security problems, in addition to the fact that communication between multi-party accounts in light of a large number of users works to disrupt networks.

In [14], the authors provided a platform for those interested in developing their programs by providing a decentralized and transparent means based on public/crowd-based computing resources. So that it provides the ability to run application files in a decentralized environment similar to cloud computing resources in storage and sharing, but in some cases, it lacks the distributed infrastructure. It is interrupted during an unlimited no of nodes assessment.

In [15], the authors proposed a ProvChain architecture as an example of a blockchain-based data generation model in a cloud environment by gathering and validating the data source. In the presence of a good level of security and data transparency, the proposed environment needs to fill in some cases where the file size increases, which generates an increase in the load that requires an increase in the computational complexity in data protection.

[16] proposed a new approach to access data without the participation of the provider through the ciphertext-policy attribute-based encryption scheme with dynamic attributes by takingadvantage of the idea of the blockchain that depends on the ledger so that the model idea is based on building a record to generate the key, or set the access policy, or change or cancellation, or request access is not subject to change, and hence the degree of security in accessto data in the computerized cloud is achieved.

Table 1: Comparing the recent research.

3. Methodology

No. Author & Year	*Study Name*	*Method or Technique*	*Advantages*	*Limitations*
[13] Caixia Yang, Liang TAN, NA SHI, Bolei XU, Yang Cao, and Keping YU, 2022.	AuthPrivacyChain: A Blockchain-based Access Control Framework with Privacy Protection in Cloud	Using the account address of the node in the blockchain as an identity and redefining the access control permission of data for the cloud	Provides privacy protection for cloud data and a decentralized access control framework through blockchain	The implementation and scalabilityy of blockchain technology may be challenging in practical applications
[9] Ashok Gupta, Shams Tabrez Siddiqui, Shadab Alam, Mohammed Shuaib, 2019.	Cloud Computing Security using blockchain	Analysis of the overall structure of the blockchain and the characteristics of the security requirements of blockchain and cloud computing	Enables secure cloud computing through the integration of blockchain technology	Lack of standardization and regulatory frameworks for blockchain technology may hinder its adoption in cloud computing
[7] Simanta Shekhar Sarmah, 2019.	Application of Blockchain in Cloud Computing	Reviewing previous studies based on the application of the blockchain in cloud computing	Provides insights into the potential applications of blockchain technology in cloud computing	The practical implementation of blockchain in cloud computing still requires further research and development
[14] Zhongli Dong, Young Choon Lee, Albert Y. Zomaya, 2019.	Proofware: Proof of Useful Work Blockchain Consensus Protocol for Decentralized Applications	A run application files in a decentralized environment similar to cloud computing resources in storage and sharing	Provides a secure and efficient decentralized consensus protocol for cloud computing applications	The implementation of Proofware may require significant computational resources, which may limit its scalability
[15] Xueping Liang, SachinShetty, Deepak Tosh,Charles Kamhoua, Kevin Kwiat, Laurent Njilla, *2017*.	ProvChain: A Blockchain-Based Data Provenance Architecture in Cloud Environment with Enhanced Privacy and Availability	A decentralized cloud data architecture using blockchain technology for collecting and verifying the data provenance	Enables the secure and transparent management of cloud data provenance through the use of blockchain technology	The implementation of ProvChain may require significant computational resources, which may limit its scalability
[16] Xiaochen Zheng, Raghava Rao Mukkamala, Ravi Vatrapu, Joaqun Ordieres-Mere, 2018	Blockchain-based Personal Health Data Sharing System Using Cloud Storage	Enabled access control over the data stored in the cloud without the provider's participation through the ciphertext-policy attribute-based encryption scheme with dynamic attributes.	Provides a secure and privacy-preserving approach for personal health data sharing in the cloud through blockchain technology	The implementation of attribute-based encryption schemes may require significant computational resources, which may limit scalability

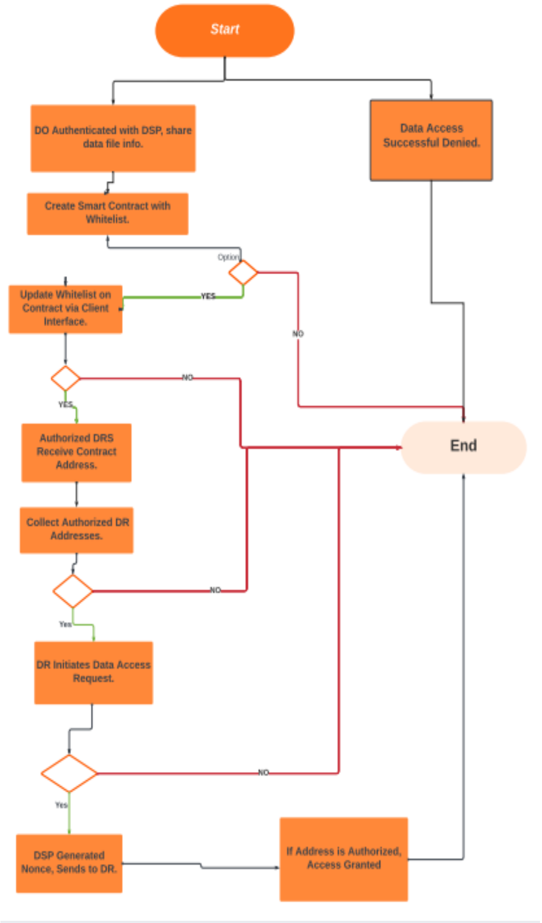

Start
DO Authenticated with DSP, share data file info.
Data Access Successful Denied.
Create Smart Contract with Whitelist.
Option
Update Whitelist on Contract via Client Interface.
YES
NO
NO
YES
Authorized DRS Receive Contract Address.
End
Collect Authorized DR Addresses.
NO
Yes
DR Initiates Data Access Request.
NO
Yes
DSP Generated Nonce, Sends to DR.
If Address is Authorized, Access Granted

Fig 3. Blockchain model for access control the cloud computing.

Four elements define our access control scheme:

Data storage provider (DSP)—it has enormous capacity to manage dispersed remote servers and host application services. Data owners may manage distant server data using these services. Since the DSP is not a blockchain client, it can only read the blockchain.

Data owners (DOs) store and distribute data with numerous entities using data storage provider services. He creates a DSP file whitelist of permitted entities with respected access privileges. The whitelist is recorded in a blockchain smart contract (C) per file.

Our system lets the DO set access privileges for each data resource.

First, because all blockchain-specific processes are safe and non-corruptible, no one can change the list of permitted organizations to access particular resources, assuring data access proofs.

Second, our method is data owner-centric since the data owner forms a contract for each outsourced data resource, including the data use access control list. BC features underpin the identifying system and authentication procedure.

Fourth, DR access is logged in BC, allowing access control system audits afterwards.

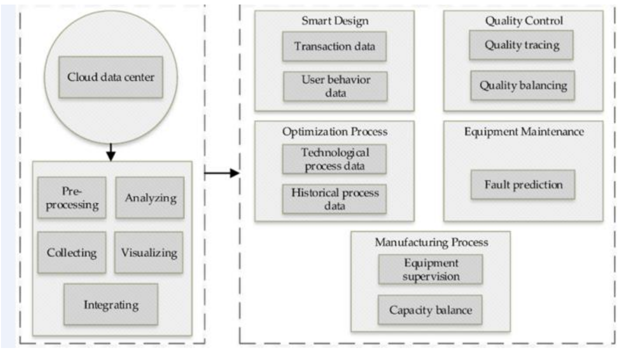


Fig. 4. Overview of edge computing architecture [17].

Interactions between entities and BC support the following rules:

• The DO, the owner of the outsourced data, may only amend the whitelist, hence the contract must contain his address for proper permissions.

• The DR may request resources through contract transaction. Thus, after authentication, the smart contract registers DR's address in the blockchain as an authorized entity. Note that our technique implies the hosting server compares. Therefore, the contract must be passive while receiving user transactions.

• Any entity, including DSP, may read the whitelist to compare addresses for DR access requests.

Indeed, variables (DR addresses in our example) are specified so the DO might modify them for later solicitations. The contract script cannot be updated, but the whitelist may.

2- Whitelist Creation Every smart contract formed by a data owner allows to list the addresses of permitted DRs to access a remote server-hosted data file. Thus, the challenge is how to define the smart contract (C)-outsourced data file relationship. We want to simplify the process of creating smart contracts by DOs using the DSP, enabling easy whitelist establishment for each DO. Fig. 1 shows the steps: 1. DO verifies with DSP and reveals the data file name he wants to safeguard.

3- DO builds smart contract. DO sends a transaction to null address. The client interface randomly assigns the contract address. The smart contract must record the whitelist in the BC to retain a history so the DSP may get the previous valid one.

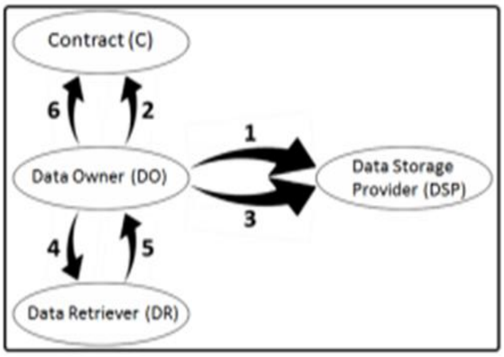


Fig. 5. Whitelist Creation Process.

4-Do provides the remote DSP the contract address and data file.

The approved DRs get the contract address from.

5-DO. We underline that each authorized DR must provide the contract address to access the outsourced data file.

6-DO retrieves approved DR addresses for the whitelist.

7-DO uses the client interface to add allowed DR addresses to the whitelist. Similar to adding a new permitted address, you may remove an address from the whitelist. The DSP address links the contract to the outsourced file. Our solution's key benefit is that the whitelist may be changed without DSP interaction.

8-An authorized DR initiates the resource access procedure with the distant hosting DSP (Fig. 5) to access an outsourced data file:

1. DR delivers DSP access request through off-blockchain channel.

2. The DSP delivers the DR a random nonce. The DSP then listens to the blockchain by scanning freshly added transactions connected with the contract and analyzing transactions with the supplied nonce.

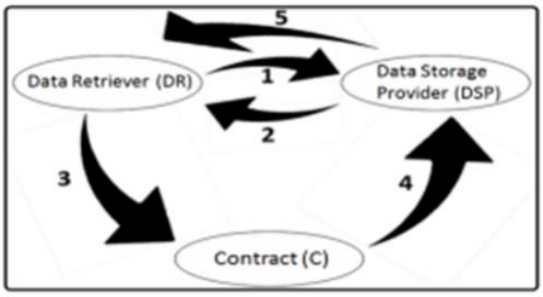


Fig. 6. Access to Resources Process.

3. The DR sends the contract a nonce-input transaction. Unlike the DO solicitation, the contract is passive at this stage. This transaction leaves a trail on the BC and establishes its authenticity without reacting to the contract.

4. After identifying the produced nonce in the input data field of a contract-related transaction, the DSP chooses the entity that issued the transaction. It then checks the transaction's originating address to the whitelist's allowed addresses.

5. If the transaction's originating address matches a whitelist-authorized address, the DSP identifies the DR.

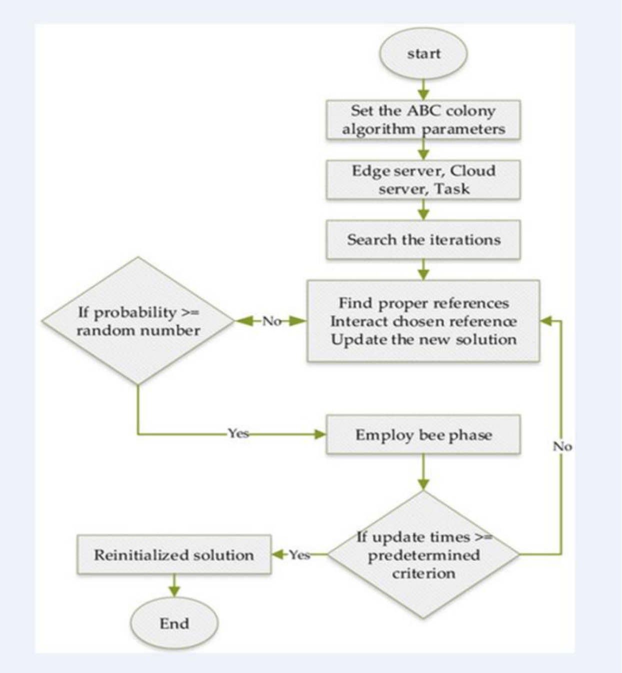

start
Set the ABC colony algorithm parameters
Edge server, Cloud server, Task
Search the iterations
Find proper references Interact chosen reference Update the new solution
If probability >= random number
No
Yes
Employ bee phase
No
If update times >= predetermined criterion
Yes
Reinitialized solution
End

Fig. 7. Artificial bee colony workflow [18].

Finally, each authentication session needs a randomly generated nonce. That is, the nonce verifies transaction freshness with the DSP. This latter links a received request to BC authentication thanks to the produced nonce. Our access control tool uses public blockchain technology for decentralized authentication, guaranteeing data auditability and transparency. Although the list of allowed addresses is publicly verifiable, the whitelist, which the DO may award DR privileges, cannot be changed [19].

a) **Implementation.**

Implementation Summary: Setting Up Blockchain Access Control System Locally.

Tools and Technologies Used:

- Programming Language: Solidity (for smart contract development)

- Node.js: JavaScript runtime for executing code outside a web browser

- Truffle: Development framework for Ethereum smart contracts

- Ganache: Local Ethereum blockchain instance for testing and development

- MetaMask: Ethereum wallet and browser extension for interacting with DApps

Steps Taken:

1. Downloaded and unzipped the access control system's code ZIP file.

2. Installed Node.js for JavaScript runtime.

3. Installed Truffle globally using `npm install -g truffle`.

4. Installed and ran Ganache for a local Ethereum blockchain instance.

5. Navigated to the project directory using the terminal.

6. Compiled the smart contract with `truffle compile`.

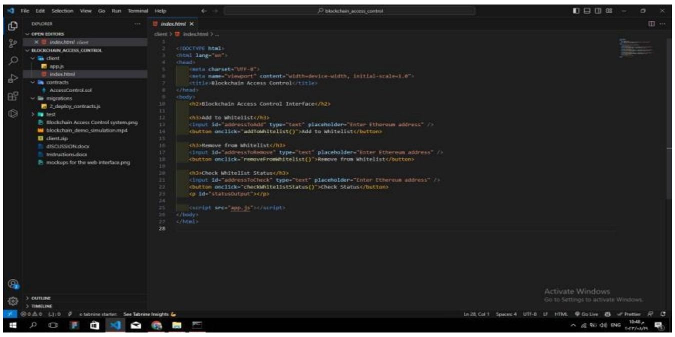
7. Migrated the compiled contract to the local blockchain using `truffle migrate`.

8. Extracted contract address and ABI, replaced placeholders in `app.js`.

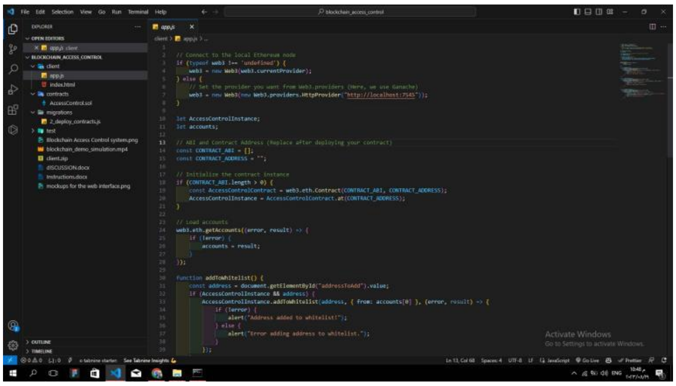
9. Installed MetaMask browser extension.

10. Connected MetaMask to the local Ethereum blockchain from Ganache.

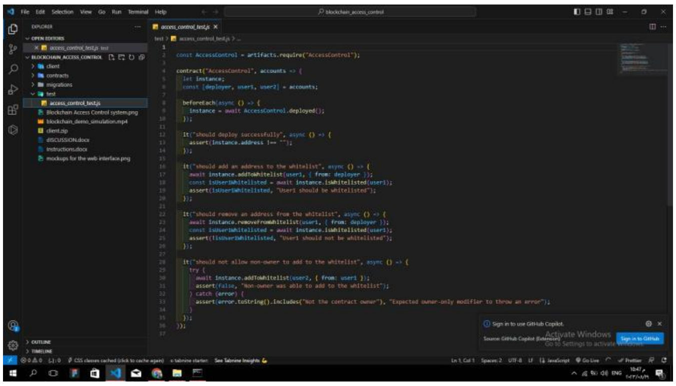
11. Interacted with the access control system's smart contract using MetaMask.



```html

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Blockchain Access Control</title>
</head>
<body>
    <h2>Blockchain Access Control Interface</h2>

    <h3>Add to Whitelist</h3>
    <input id="addressToAdd" type="text" placeholder="Enter Ethereum address" />
    <button onclick="addToWhitelist()">Add to Whitelist</button>

    <h3>Remove from Whitelist</h3>
    <input id="addressToRemove" type="text" placeholder="Enter Ethereum address" />
    <button onclick="removeFromWhitelist()">Remove from Whitelist</button>

    <h3>Check Whitelist Status</h3>
    <input id="addressToCheck" type="text" placeholder="Enter Ethereum address" />
    <button onclick="checkWhitelistStatus()">Check Status</button>
    <p id="statusOutput"></p>

    <script src="app.js"></script>
</body>
</html>

```



```javascript

// Connect to the local Ethereum node
if (typeof web3 !== 'undefined') {
    web3 = new Web3(web3.currentProvider);
} else {
    // Set the provider you want from Web3.providers (Here, we use Ganache)
    web3 = new Web3(new Web3.providers.HttpProvider("http://localhost:7545"));
}

let AccessControlInstance;
let accounts;

// ABI and Contract Address (Replace after deploying your contract)
const CONTRACT_ABI = [];
const CONTRACT_ADDRESS = "";

// Initialize the contract instance
if (CONTRACT_ABI.length > 0) {
    const AccessControlContract = web3.eth.Contract(CONTRACT_ABI, CONTRACT_ADDRESS);
    AccessControlInstance = AccessControlContract.at(CONTRACT_ADDRESS);
}

// Load accounts
web3.eth.getAccounts((error, result) => {
    if (!error) {
        accounts = result;
    }
});

function addToWhitelist() {
    const address = document.getElementById("addressToAdd").value;
    if (AccessControlInstance && address) {
        AccessControlInstance.addToWhitelist(address, { from: accounts[0] }, (error, result) => {
            if (!error) {
                alert("Address added to whitelist!");
            } else {
                alert("Error adding address to whitelist.");
            }
        });
```



```javascript

const AccessControl = artifacts.require("AccessControl");

contract("AccessControl", accounts => {
  let instance;
  const [deployer, user1, user2] = accounts;

  beforeEach(async () => {
    instance = await AccessControl.deployed();
  });

  it("should deploy successfully", async () => {
    assert(instance.address !== "");
  });

  it("should add an address to the whitelist", async () => {
    await instance.addToWhitelist(user1, { from: deployer });
    const isUser1Whitelisted = await instance.isWhitelisted(user1);
    assert(isUser1Whitelisted, "User1 should be whitelisted");
  });

  it("should remove an address from the whitelist", async () => {
    await instance.removeFromWhitelist(user1, { from: deployer });
    const isUser1Whitelisted = await instance.isWhitelisted(user1);
    assert(!isUser1Whitelisted, "User1 should not be whitelisted");
  });

  it("should not allow non-owner to add to the whitelist", async () => {
    try {
      await instance.addToWhitelist(user2, { from: user1 });
      assert(false, "Non-owner was able to add to the whitelist");
    } catch (error) {
      assert(error.toString().includes("Not the contract owner"), "Expected owner-only modifier to throw an error");
    }
  });
});

```

Sign in to use GitHub Copilot.

Outcome:

- Successfully set up a local environment for testing and understanding the blockchain access control system.

- Compiled and deployed the smart contract on the local Ethereum blockchain.

- Used MetaMask to interact with the smart contract, executing transactions and managing Ethereum accounts.

4. Results and Discussion

In this segment, we first present our statement model. Then, we give a security conversation of our proosed blockchain-based admittance control plot, regarding the security and protection necessities itemized.

Threat Model

We assume that an attacker can peruse, send, and drop an exchange addressed to the blockchain when designing a secure blockchain-based access control scheme. As follows, the attacker targets data owners, data retrievers, data storage providers, and the blockchain:

• An attacker attempts to impersonate a data owner based on previous data access request sessions as well as provided blockchain data in order to grant an honest data storage provider some rights to be logged into the blockchain without the legal data owner's granted privileges. This attack is evaluated in light of the requirements for authenticated access control and auditability.

• An attacker attempts an attack against the privacy property with respect to both data owners and data retrievers when attempting to directly link a smart contract to a specific owner.

• An aggressor attempts to keep a genuine exchange from being distributed in the blockchain. For instance, an assailant might endeavor a DoS assault against an entrance list change movement or a blockchain flooding assault with invalid information. This assault is assessed considering the auditability and accessibility prerequisites.

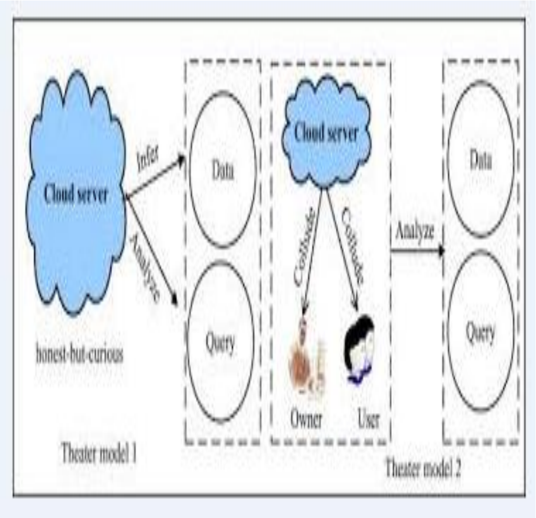
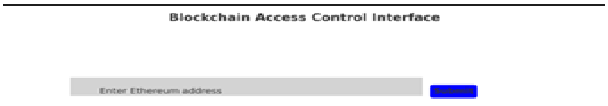
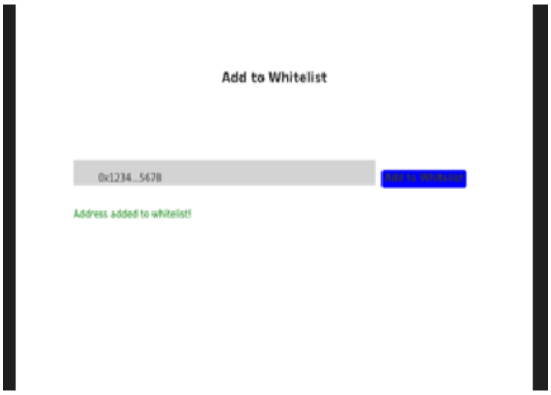


Fig. 8. Threat Model [17].

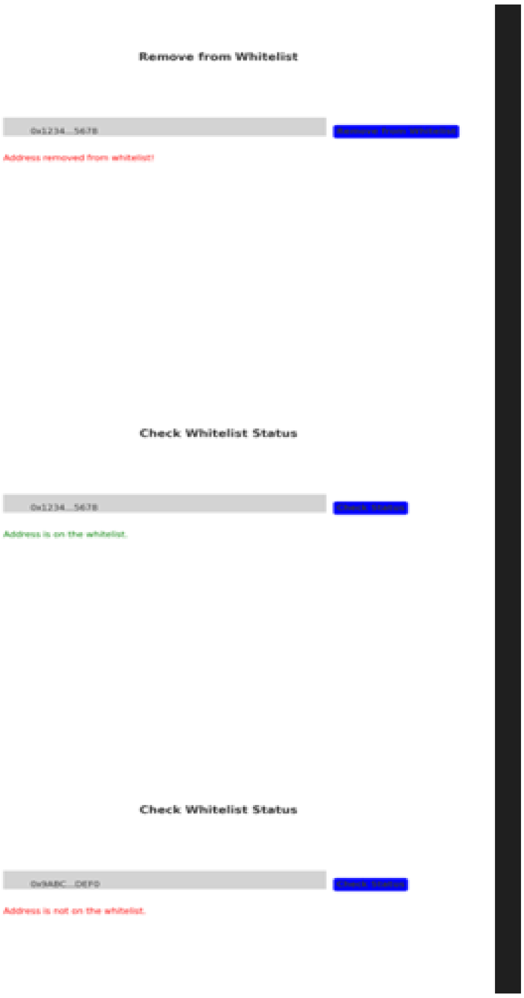

Blockchain Access Control Interface

Enter Ethereum address


Add to Whitelist

0x1234...5678

Address added to whitelist!


Remove from Whitelist

0x1234...5678

Address removed from whitelist!

Check Whitelist Status

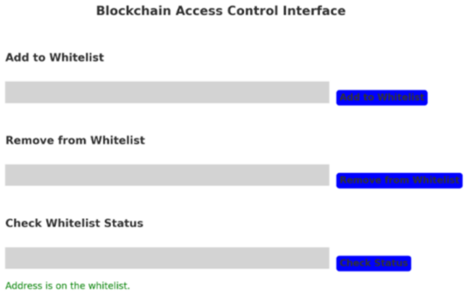
0x1234...5678

Address is on the whitelist.

Check Whitelist Status

0x9ABC...DEF0

Address is not on the whitelist.



Address is on the whitelist.

Fig. 9. Blockchain Access Control interface.

5. Conclusion

Blockchain-based access control systems offer a paradigm shift in how we manage data and enforce security measures. By eliminating central points of control and introducing transparency, decentralization, and immutability, these systems bring unprecedented levels of security and privacy to data management.

The step-by-step instructions provided in this discussion allow you to explore and test the potential of blockchain in access control on your local machine. However, the true impact of this technology lies in its broader implementation across industries and sectors. As blockchain continues to evolve, its role in access control will become more significant, providing a foundation for a more secure and decentralized digital future.

Acknowledgment

The authors would like to thank the Deanship of Graduate Studies at Jouf University for funding and supporting this research through the initiative of DGS, Graduate Students Research Support (GSR) at Jouf University, Saudi Arabia.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] OECD, “Recommendation of the Council on Health Data Governance,” 2019. OECD/LEGAL/0433, Paris. Retrieved fromhttps://www.oecd.org/health/health-systems/Recommendation-of-OECD-Council-on-Health-DataGovernance-Booklet.pdf

[2] B. P. SERIES, “Opportunities and Challenges of Blockchain Technologies in Health Care,” 2020.

[3] Nedhal, A. Al-Saiyd, N. Sail, “Data Integrity in Cloud Computing Security,” *Journal of Theoretical and Applied Information Technology*, vol. 58, no. 3, pp. 570-581, 2013.

[4] M. Alshinwana, A. Y. Shdefatb, N. Mostafab, A. A. M. AlSokkarc, T. Alsarhana, and D. Almajalic, “Integrated cloud computing and blockchain systems: A review,” *International Journal of Data and Network Science*, vol. 7, no. 2023, pp. 941-956, 2023.

[5] S. T. Faraj, S. A. Jassim, and K. Kifayat, “Management of Identity and Access in the Cloud,” *Journal of the University of Anbar for Pure Science*, vol. 6, no. 2, pp. 11-23, 2012.

[6] W. Ahmad, A. Rasool, A. R. Javed, T. Baker, and Z. Jalil, “Cyber security in IoT-based cloud computing: A comprehensive survey,” *Electronics*, vol. 11, no. 1, 2022.

[7] S. S. Sarmah, “Application of block chain in cloud computing,” *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 8, no. 12, pp. 4698-4704, 2019.

[8] O. Ali, A. Jaradat, A. Kulakli, and A. Abuhalimeh, “Comparative study: Blockchain technology utilization benefits, challenges, and functionalities,” *Ieee Access*, vol. 9, pp. 12730-12749, 2021.

[9] A. Gupta, S. T. Siddiqui, S. Alam, and M. Shuaib, “Cloud computing security using blockchain,” *Journal of Emerging Technologies and Innovative Research (JETIR)*, vol. 6, no. 6, pp. 791-794, 2019.

[10] W. Venters and E. A. Whitley, “A critical review of cloud computing: Researching desires and realities,” *J. Inf. Technol.*, vol. 27, no. 3, pp. 179–197, 2012.

[11] J. Zhang, X. Nian, and H. Xin, “A Secure System For Pervasive Social Network-based Healthcare,” *IEEE Access*, vol. 4, pp. 9239-9250, 2016.

[12] Ch. V. N. U. B. Murthy, L. M. Shri, S. Kadry, and S. Lim, “Blockchain Based Cloud Computing: Architecture and Research Challenges,” *IEEE Access*, vol. 8, 2020.

[13]C. Yang, L. Tan, N. Shi, B. Xu, Y. Cao, and K. Yu, “AuthPrivacyChain: A blockchain-based access control framework with privacy protection in cloud,” *IEEE Access*, vol. 8, pp. 70604-70615, 2020.

[14] Z. Dong, Y. C. Lee, and A. Y. Zomaya, "Proofware: Proof of useful work blockchain consensus protocol for decentralized applications," 2019, arXiv preprint arXiv:1903.09276.

[15] X. Liang, S. Shetty, D. Tosh, C. Kamhoua, K. Kwiat, and L. Njilla, "Provchain: A blockchain-based data provenance architecture in a cloud environment with enhanced privacy and availability," *IEEE*, 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID), pp. 468-477, 2017.

[16] X. Zheng, R. R. Mukkamala, R. Vatrapu, J. Ordieres-Mere, "Blockchain-based Personal Health Data Sharing System Using Cloud Storage," *IEEE*, J. Kubicek (Ed.), 20th International Conference on E-Health Networking, Applications and Services (Healthcom) Los Alamos, CA: IEEE, 2018.

[17] N. Mansourov, and D. Campara, "Knowledge of risk as an element of cybersecurity argument," System Assurance, 2011.

[18] Z. Shahbazi and Y. C. Byun, "Improving transactional data system based on an edge computing–blockchain–machine learning integrated framework," *Processes,* vol. 9, no. 1, pp. 92, 2021.

[19] M. Laurent, N. Kaaniche, C. Le, and M. Vander, "Plaetse, A blockchainbased access control scheme," *ICETE,* 2018 - Proceedings of the 15th International Joint Conference on e-Business and Telecommunications, vol. 2, no. 7, pp. 168–176, 2018.