National College of Ireland

Improving Security and Transparency in Data Sharing with Web3 Integration and Blockchain Smart Contracts for Amazon S3 Access

MSc Research Project
Msc Cloud Computing

KrishnaMurthy Kowsik Gelli
Student ID: x23242817

School of Computing
National College of Ireland

Supervisor: Sai Emani

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	KrishnaMurthy Kowsik Gelli
Student ID:	x23242817
Programme:	Msc Cloud Computing
Year:	2024-2025
Module:	MSc Research Project
Supervisor:	Sai Emani
Submission Due Date:	12/12/2024
Project Title:	Improving Security and Transparency in Data Sharing with Web3 Integration and Blockchain Smart Contracts for Amazon S3 Access
Word Count:	7800
Page Count:	26

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	KrishnaMurthy Kowsik Gelli
Date:	11th December 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Improving Security and Transparency in Data Sharing with Web3 Integration and Blockchain Smart Contracts for Amazon S3 Access

KrishnaMurthy Kowsik Gelli
x23242817

Abstract

This study introduces a novel approach for enhancing data sharing security and transparency in Amazon S3 using blockchain smart contracts and Web3 technologies. It does this by overcoming the limitations of traditional Identity Management Systems (IAM) through a hybrid model based on decentralized systems combined with cloud storage scalability. To make it cost-effective, Polygon blockchain is used to deploy cheap smart contracts while its user authentication interface employs decentralized application (dApp) for request signing. AWS Lambda functions are included in the architecture to enable seamless integration between S3 data retrieval and access control powered by blockchain technology. The process entails creating and deploying Smart Contracts, developing Web3 dApps, as well as implementing serverless functions at AWS where it is hosted. For better comparison against other conventional IAM systems in terms of security, transparency, usability metrics, extensive evaluation will be performed. This framework has immense potential in real-world applications, particularly healthcare systems where safe sharing of sensitive patient information among different medical professionals and organizations is allowed. While demonstrating the cloud system wide data access control revamp the research demonstrate that the web3 and the blockchain can deliver the higher security and accountability to the user across various domains while ensuring strict access control and complete audit trails remain as intact

1 Introduction

1.1 Research Background

There is an increase in the popularity of Amazon Web services(AWS) Simple Storage Service(s3) and other cloud based data storage solutions as the cloud computing has became the mainstream. However there is lack of transparency and security in traditional Identity and Access management(IAM) system despite being the efficient at processing the s3 dataBadirova et al. (2023). Generally this kind of solutions rely on the single point of control that can fail or attract various threats to system such as the unauthorized access or the information leakage. To mitigate this issue an approach cab be proposed that combine web3 with blockchain technology that allows for distributed access control across many devices and also improve accountability through smart contracts and immutable ledgersChinnasamy et al. (2023).

The Blockchain technology have provided a secure foundation for sharing the information because its a decentralized and unchangeable. The smart contract plays an prominent role in decentralized system as they automate complex processes, ensure enforcement of access control and prevent transactional opacity and tampering among others. On the platform like the ethereum, dapps(decentralized apps) cab also be used together with smart contracts to automate enforcement of access control polices without an involvement of intermediaries thereby enhancing the security while ensuring transparency regarding the data governance and access. Oliva et al. (2020). Additionally, if MetaMask – an Ethereum wallet plus browser extension designed for dApps is used, users could authenticate themselves using private keys before signing requests Renu and Banik (2021).

1.2 Problem Statement

Cloud computing is advancing with cloud based public storages likes Amazon S3, but IAMs are not entirely secure when it comes to user access policy management. Most importantly, there is a lack of fine control over access permissions, making for vulnerabilities to attacks. They also have difficulty in sharing data between two or more domains Nair et al. (2022). Furthermore, they are not governed by immutability audit trail data laws for incorruptible evidence Ahmad et al. (2021). While cloud storage does not have the scalability or integration complexity problems that are inherent in blockchain technology, it provides a decentralized alternative access control Habib et al. (2022). In addition, blockchain technologies are not deeply engaged by AWS cloud services so as to provide a user-friendly interface for on demand control access modification Li et al. (2021). This research assumes a hybrid version of the framework built around Web3 technologies and layer 2 blockchain solutions, which would enhance security and clarity of AWS S3 data sharing while maintaining the performance levels of enterprise.

1.3 Research Question

How can a Web3-Blockchain model using smart contracts be implemented to increase security, transparency, and economic viability of AWS S3 data sharing and access management for medical records, as well as overcome potential limitations and challenges of decentralized verification and authorization?

1.4 Research Objective

The research is an attempt toward developing a bottom up framework based on blockchain and smart contract, which can improve the security and transparency in data sharing in Amazon S3 specifically for medical systems. It aims to combine Web3 technologies with cloud storage scalability to fix up the problems of the traditional Identity Access Management (IAM) Systems by offering better security, transparency and user goverance.

1.5 Research Contributions

- Instigate a decentralized framework that enacts authentication and authorization using blockchain and smart contracts for safe sharing of medical records in healthcare systems powered by Amazon S3.

- Create a model that incorporates advantages from decentralized systems along with the scalability of cloud storage while solving issues with traditional IAM in healthcare data sharing.
- Construct a decentralized application framework that employs Web3 technologies for user verification and request signature to increase the safety and user governance in healthcare data sharing.
- To assess the efficiency of using Ethereum-based Polygon smart contracts for governance of access controls in healthcare systems.
- We will test the effectiveness of AWS Lambda functions to create a smooth connection between S3 data access and blockchain access control.
- To analyze how the proposed DeAuth framework stands against conventional centralized data sharing practices in healthcare contexts.

2 Related Work

This review of past research works done in the mechanisms of sharing data securely across clouds is segregated into four sub-sections based on their subjective matter as follows: Blockchains and the concept of smart contracts for access control are presented in section 2.1, the use of web3 and dApps for application deployment is discussed in section 2.2, secure data sharing in cloud environments with practical use-case scenarios is dealt in section 2.3, a comparative analysis of centralized and decentralized data sharing in healthcare systems, and finally, section 2.5 concludes the literature analysis identifying the research gaps and the proposal to overcome those issues.

2.1 Blockchain and Smart Contracts based Access Control

Song et al. (2020) propose a blockchain and smart contract based novel approach to attribute based access control (ABAC) for IoT environments. Pursuing on the ABAC model, blockchain-based access control decisions are being proposed by combining the above model with blockchain technology and using smart contracts to make access control decisions. The smart contract system consists of a management contract, a permission decision contract and several policy contracts. To implement the scheme, the authors used the Ethereum private blockchain Network and smart contracts. The main contribution of this study is to show that the proposed scheme solves well the problem of implementing dynamic, distributed, and trusted control of access in open IoT environments. In their work, (Kamboj et al.; 2021) introduced a Role-Based Access Control (RBAC) model that is created by smart contracts for managing user role permissions in an organization. This research focused on authentication and authorization of existing Role based access control by utilizing decentralized and transparent features of Blockchain. The paper proposed two smart contracts to manage user–role assignments as managed by the role issuer and resource owner, and tested that on the Ropsten Ethereum Test Network.

The access control framework for secure and scalable resource management in distributed systems presented in (Hao et al.; 2021) is developed on smart contracts. The

authors proposed framework addressed limitations of centralized access control and addressed them by using smart contracts. Specifically, the framework relies on attribute-based access control (ABAC), where the owner of the resource assigns attribute sets to clients through signatures. The access policies for resources are managed by an access control smart contract deployed on the blockchain, and requests for access are verified on-chain. The framework also makes off-chain signature unforgeable for its reliability and make on chain operation credible. The proposed framework in the research shows scalability, efficiency, and economic viability improvement over the existing blockchain based access control solutions along with preserving reliability and auditability. In their work, (Rana et al.; 2023) the decentralized model that will protect the digital evidence using smart contract in the Layer 2 Polygon blockchain was proposed that uses the immutability, transparency and decentralization of the blockchain to facilitate a trustless, automated framework around the handling of digital evidence. The access control, evidence storage and transaction tracking are done using the smart contracts on the Polygon blockchain. For decentralized storage of all the evidence file, they use InterPlanetary File System (IPFS) and save these content identifiers in blockchain. The paper concludes that decentralized model is much more valuable than traditional centralized key value stores when considering digital evidence.

Al Neyadi et al. (2023) present the case of the implementation of role-based access control (RBAC) in a private blockchain for IoT integrated smart contracts. This methodology involves a semi-decentralized mechanism of authentication, decentralized mechanisms of user group interactions and smart contract for enforcing RBAC. It also contains a machine learning layer to improve security by learning whether the user authenticates by employing user specific information. By seamlessly combining the private blockchain technology for authentication and RBAC model, this system considerably enhances the authentication efficiency and reduces authentication time over the existing state of art methods.In conjunction with smart contracts and sharding techniques, (Bakhtiary et al.; 2024) propose ComboChain, a blockchain based access control system for Internet of Things (IoT) environments that combine a hierarchical attribute based access control (HABAC) model. With the hierarchical structure of subject and object attributes brought forth by Combo-Chain the policy management and evaluation could be further simplified. The system architecture includes the request chain of how access requests are handled, the oracle nodes to talk to various tiers, and two shards to keep and manage attributes and policies. Several system functions are implemented with smart contracts, including policy splitting, attribute management and access request evaluation. Furthermore, (Yang et al.; 2024) put forward an access control model on blockchain master-sidechain collaboration (AC-BMS) for medical information systems aiming at the improvement of scalability, efficiency, and privacy of medical information storage. The model features a password-based authentication scheme, preventing access requests sent to the Hospital Information System (HIS) by other than the authenticated doctors. The model is contextualized by reducing the reliance on third party restrictions. Simulation experiments in Hyperledger Fabric find that the AC BMS model yields access efficiency and throughput improvement of 2-3 seconds faster access time with constant latency around ¡ 0.4 minutes and throughput of 288 - 296 transactions per second.

2.2 Web3 and Decentralized Applications (dApps)

A decentralized system for protecting data sharing was created by (Lin et al.; 2024) that employed blockchain technology along with decentralized identifiers (DIDs). The ethereum blockchain was employed to preserve DIDs and hash values and IPFS kept the substantive data to increase persistence and access. The components for creating DIDs were part of the system framework while data storage was done on IPFS and sharing was facilitated by smart contracts. In comparison to centralized systems, the performance assessments revealed enhanced scalability and integration capabilities. Preliminary assessments revealed improved data integrity along with a minimized risk of unauthorized access. (Austria et al.; 2024) presented a DeAuth that is a decentralized authentication and authorization system for safe and private data sharing. By involving the blockchain and smart contract along with the decentralized identity and P2P storage the discussed framework allows user to manage the data effectively. The system architecture included a digital wallet for managing identity and access control smart contracts along with IPFS for data storage. The authors designed a test version that ran on Ethereum blockchain and evaluated its operations relative to existing cloud services.

The authentication mechanism of (Petcu et al.; 2023) is a secure and decentralized one based on Web 3.0 and Ethereum blockchain technology. It's a blockchain on Ethereum, powered by a contemporary web stack to improve the user interaction. The authors discuss next generation of Web 2.0 authentication methods to which the proposed mechanism is compared, showing its benefits over traditional Web 2.0 authentication methods in terms of security, privacy, and user control. The technical stack consists of Ethereum based wallets, front and back end where we run a node, a database to store login attempts and nonces. The steps of a wallet connection, nonce generation, signing and validation make up the authentication flow. Security attack models, and how quantum computing could affect Ethereum's cryptographic algorithms are discussed in the study. In accordance with this, (Madine et al.; 2021) came up with the appXchain, a cross-chain interoperability solution across blockchain networks by utilizing applications. The system is decentralized, tamper proof, and flexible for diverse use cases at either limited implementation and deployment costs. The first uses a fusion interface layer inside the Cross-Chain Hub DApp for translating cross chain interactions and delegating requests between blockchain networks. For the healthcare industry, the authors designed algorithms and smart contracts for the sharing of patient centered EMR documents among different blockchain networks. However, the deployment costs of appXchain increase, and it is not fully upgradable, yet it would need to be regarded when interfacing between different or heterogeneous domains.

Alencar et al. (2024) suggest a framework to help developers decide the most suitable decentralized platform to develop decentralized applications (DApps) on blockchain networks. The framework offers guidance for choosing decentralized platforms based on a set of requirements of DApp like fault tolerance, transparency, privacy, language. The limitations of the study are acknowledged by the authors including the critical nature of a systematic literature review and the paper of time frame of the reviewed articles (2020–2023). They also assert that since this framework is still in its initial phase and there are several aspects that still need refinement and expansion to include facets of user interface management strategy and data management strategy. Using this multi layer sharding blockchain structure, (Liu et al.; 2024) presents SS-DID: a secure and practical decentralized identity (DID) system for Web3 applications. The goal of the system is to

alleviate scaling and cross field management issues of existing DID solutions of large scale Web3 identity applications. SS-DID employs a four layer architecture composed of user, field user management, in field shard management and field management layers. The paper presents complete system processes of initialization, DID lifecycle management, credential usage as well as IMI node management, and introduces query optimizations over the cross field credential verification efficiency by means of Merkle proofs.

2.3 Secure Data Sharing in Cloud Environments

Prasad and Rekha (2023) propose a blockchain based Identity, Access Control and Secure Authentication (BC-IAS) protocol to increase security and privacy in cloud computing environments. This protocol compromises itself to counter security and privacy issues involving cloud computing in the presence of malicious attacks on wireless and mobile communication networks. The proposed BC-IAS protocol is compared with existing models and it is shown to outperform in terms of data access rate, message delivery ratio, end-to-end delay and energy consumption. However, the paper does not discuss the possible issues of scalability and compute overhead of BC-IAS protocol. Baviskar (2022) proposes that data leakage from Amazon S3 buckets can be prevented using AWS Lambda by way of an automated encryption approach. The problem that they tried to solve is the security vulnerabilities exposed by misconfigurations and human errors of S3 buckets, even if they are marked as private. Also, the author uses serverless architecture by implementing AWS Lambda to automatically encrypt S3 buckets when it's created without manual intervention. The whole methodology consists in the creation of the stack template in AWS CloudFormation creating the Lambda function, IAM roles, etc., in the necessary AWS services. While the study has its main limitation: it does not cover other cloud platforms as it is based on AWS services.

Khande et al. (2023) introduce a way of encrypting data stored in AWS S3 cloud storage without the need for continuous monitoring. The proposed method uses existing AWS services, including AWS CloudFormation, to automate the provisioning of resources, and encrypting data. The authors recommend the way to generate the key and used key management service (AWS KMS), and aes-256 block cipher for encryption uses. The data is encrypted back to the S3 bucket and when the data is requested from the client, the decryption process starts. The results show that in combination with the YAML script, and cloud services, data security can be automated with reasonable efficiency using encryption and server side computation using the proposed model. To secure the communication in cloud environments, (Sucharitha et al.; 2023) suggest a novel blockchain aided ciphertext policy decentralized attribute based encryption (BA-CP-DABE) scheme. Thus, CP-DABE is exploited to achieve fine grained search as well as secure key generation in the data access management while leveraging the immutability of the blockchain to ensure ciphertext confidentiality. In this proposed scheme, the encrypted data is stored on the blockchain and secure searching of encrypted data is achieved with the use of searchable encryption. Attribute based encryption is used to encode the keywords and store them on a remote sever, along with the ciphertext in the blockchain. They compare the scheme to existing attribute based encryption schemes and highlight its advantages for functional characteristics, such as support for searchability, privacy protection, and integration with blockchain technology.

2.4 Decentralized Secure Data Sharing in Healthcare Applications

In the e-commerce platform setting, (He et al.; 2023) study how e-commerce providers cooperate with the manufacturers to produce custom products through the Customer-to-Manufacturer (C2M) business model. A Hotelling linear model with customers uniformly distributed across a market segment was developed by the authors and a two stage analytical framework was constructed for evaluating both the centralized and decentralized channels. The results of the study show that whether custom products are offered to consumers depends upon consumers' fit sensitivity and that low fit sensitivity results in no customization, medium fit sensitivity suggests partial customization, and high fit sensitivity invokes full market customization.(Joshi and Kumar; 2020) propose a decentralized framework based on blockchain technology to tackle Customer to Customer (C2C) e-commerce platform challenges. The platform entwines Ethereum blockchain and smart contracts in order to create a decentralized platform that replaces traditional centralized e-commerce. The system consists of three core modules: Seller, Cost Evaluation and Reviewer. They propose a proof of stake approach for sellers and reviewers that is both accountable and discourages malicious behavior. The key achievements include the decentralized platform, the absence of transaction fees, the fraud prevention, the data privacy and trust in the trading environment higher than on any typical C2C e-commerce platform.

To improve the reliability and transparency of reputation systems, (Zhou et al.; 2021) propose a blockchain-based decentralized reputation (BC-DRS) system for online shopping. The system relies on blockchain, Interplanetary File System (IPFS), and smart contract technologies to institute a decentralized reputation management solution. It can alter the product information save in IPFS and reputation score in blockchain. This research studies online shopping environment and proposes a monetary incentive mechanism to stimulate real users to comment. The utility of it was demonstrated on the Ethereum Blockchain using Solidity that runs the methodology and is simulated on the blockchain. However, implementation complexity and the wide spread adoption of blockchain are potential shortcomings. (Xiao et al.; 2022) describe a novel distributed e-commerce transaction system, the decentralized e-commerce system based on the blockchain technology to help solve the problems of centralized e-commerce system. The system, designed using Ethereum and IPFS aims to improve transaction security and performance by incorporating four key stages: The information upload, purchase confirmation, logistics tracking and evaluation, etc. In addition, the system stores commodity details in the IPFS and stores addresses in the blockchain to increase data management efficiency. Results from experiments demonstrate low communication costs and the accurate calculation of reputation value and overall good reliability.

2.5 Critical Analysis

The observations from the literature review identify several areas where existing data sharing and access management research fall short, with Healthcare systems and e-commerce applications being among the most neglected. Traditional Identity and Access Management (IAM) systems suffer from critical limitations which include centralized control security breach, lack of transparency and inflexible access permissions. However, existing blockchain-based solutions tackling these problems have so far remained isolated

to specific domains such as IoT or healthcare.

These challenges are tackled by the proposed DeAuth framework which proposes a hybrid approach that is specifically designed to address Amazon S3 data sharing in the context of healthcare environments. The research then introduces a decentralized authentication model that leverages Web3 technologies and blockchain smart contracts to overcome the underlying limitations of traditional IAM systems.Using polygon blockchain strategy for the cost effective smart contract deployment, the data management becomes economically viable in medical applications. The dApp provides the user authentication decentralized application, the smart contracts provide granular access control and the Lambda functions seamlessly integrate the AWS cloud to make the whole product possible. At variance with past research anchored on theoretical frameworks, this approach presents a practical setting at the core of medical data sharing that enriched security, transparency, and user governance. It addresses critical healthcare systems challenges such as securing patient information, fine grained access controls, and immutable audit trails.

The proposed approach combines the scalability of cloud storage with the security provided by blockchain technology to create a comprehensive method that is superior to the previous research. This technology provides healthcare platforms with a powerful mechanism for sensitive data sharing along with preserving system security and enhancing the transparency also giving the users more control while meeting the performance quality demand of large enterprise applications.

3 Methodology

The secure and transparent access control mechanism for electronic health record implementation reserach methodology describe a approach to solve the challenges of the traditional centralized system by the use of blockchain technology and cloud computing advantages. The conceptual framework and the development of the approach along with the evaluation of methods used in the research are outline in this section.

3.1 Research Problem

The transparent access control and the secure management of the electronic health records (EHR) is the big problem for the healthcare institutes. The centralized access control system generally face security vulnerabilities, opaque access log and the sparse granularity over the data access permissions. This challenges become very critical in context to the health care organizations where the focus is on complying with the HIPAA data privacy regulations that requires the significant constraints on the use of the data and also trust in the transparency. In order to resolve these challenges, a multi-layered approach able to integrate blockchain technology with the use of cloud storage solutions, which is the conceptual framework applied to this research, is proposed. The framework proposes a decentralized system for access control, which uses smart contracts on the Polygon blockchain to control permissions and AWS S3 to securely store health records. They unite to generate a hybrid system that takes the best from blockchain and cloud storage: immutability and transparency as from blockchain and scalability and reliability as from the cloud storage.

3.2 Development Framework

The development framework is comprised of four levels of interaction of layers interconnected to each other for solving different parts of the system implementation. The first layer is all about environment setup and configuration, laying down the foundational infrastructure that all blockchain and cloud components needs to play together. It covers configuiring the Polygon network parameters, spinning up AWS services, and setting up the development environment for the smart contract deployment.

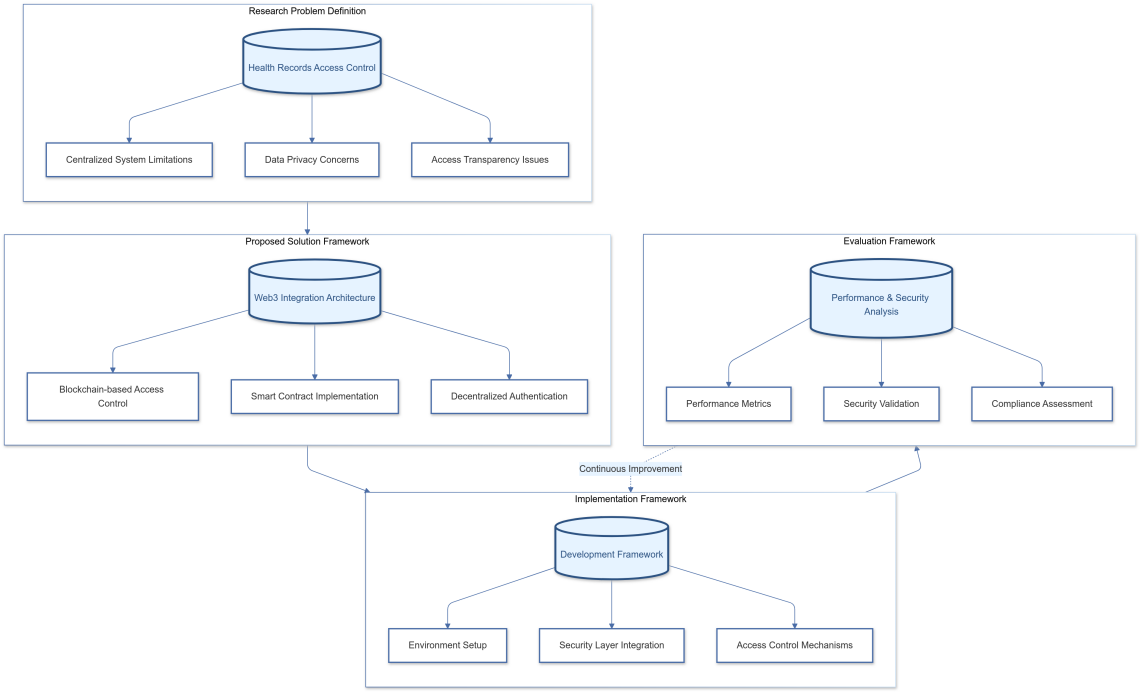

Figure 1: Proposed Research Framework

For the smart contract development layer, the core access control logic has been implemented using Solidity, the main development language for Ethereum compatible blockchain. The smart contracts are designed with three distinct access control mechanisms,one-time access, access limited to a defined time period, and access until revoked. As a consequence, the policies related to application access are flexible, but also secure, as it is possible to fine tune them according to requirements, and user's roles.

The blockchain components are bridged with a cloud storage service by the access control implementation layer. Web3 wallets, user authentication, and smart contracts access request processing are handled by this layer that also controls the exchange between smart contracts and AWS services. Signature verification mechanisms are implemented to insure only authorized user can request access to a specific health record.

3.3 Security Model

The protection mechanisms embodied at all system components are realized within the security framework layer. This includes the ability to encrypt health records at rest and in transit, having the ability to securely manage keys with AWS Secrets Manager, and applying the principals of least privilege access with IAM roles. CloudFront signed URLs

are used for secure content delivery through a framework, allowing not even authorized users to access documents except through time limited, cryptographically signed URLs.

Signature verification of all access requests is an important element of the security framework. Requests must be signed with the user's private key, and the signature verified against the corresponding public keys stored on the smart contract. This generates a sturdy authentication system that appears to ban any strikes of unauthorized get admission to even as maintaining a resilience audit path on the blockchain.

3.4 Evaluation Process

The three major dimensions of the evaluation framework are built on performance metrics, security analysis, and compliance validation. Response times to access requests, transaction costs on the Polygon network, as well as system throughput under different load conditions are measured in the performance testing. This entails looking at the latency in blockchain operations, as well as performance of the CloudFront content delivery network.

The security analysis involves the evaluation of the efficacy mechanism of signature verification and the security of cloud storage. Compliance validation helps to assure that the implemented solution are compliant with the healthcare industry regulation, especially required by HIPAA regulation. The analyzer at this point analyzes to what extent the system can provide audit capabilities, to how great an extent are access controls running, and whether the protected health information (PHI) in the system is securely maintained throughout its lifecycle.

This described research methodology is a comprehensive method for developing and evaluating a secure, transparent health records access control system. Blockchains with cloud services offer a solid platform for managing the electronic health records, and such facility must be in compliance with the relevant healthcare regulations.

4 Design Specification

The architectural components, interaction flows, and security mechanisms implemented as part of the proposed system are described through design specifications of an architecture that enables secure and transparent access control for health records. In this section, we discuss the system architecture, component interactions, and decision about the technical details of the system.

4.1 System Architecture Overview

A hybrid approach that incorporates blockchain technology and cloud service to realize a robust and secure health records management system forms the system architecture. The core of the architecture uses Polygon blockchain for access control management and the AWS services to secure store, and deliver health records. This design favours the immutability and transparency of blockchain technology as well as the scalability and maturity of the cloud infrastructure.

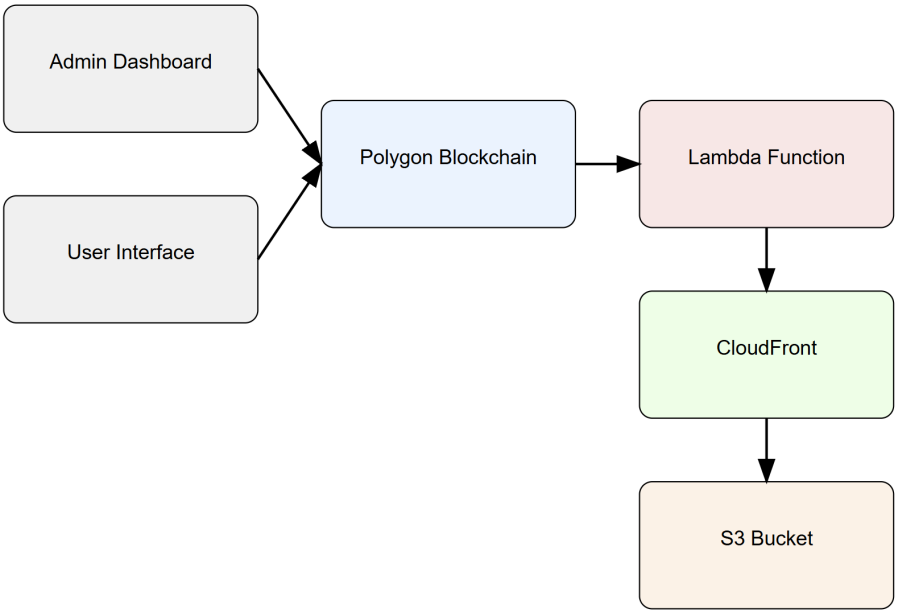

Figure 2: Proposed Research Methodology

4.2 Component Architecture

The access control system itself, being a set of interconnected components, covers certain functions of access control workflow. There is Admin Account (contract deployer) with highest authorities, and it is the account that deploys smart contracts; sets up access control rules. This component consists of smart contracts, which encode access control logic and maintain access permissions state, and communicates directly with the Polygon blockchain. The User Interface component presents the web interface for administrator and healthcare professionals. This interface works with Web3 wallets, this means it will allow for authentication and signing of transactions through MetaMask and more wallets in the future. Although it might seem it's not necessary, the wallet integration is crucial, since it gives us the cryptographic powers to send a message from the user securely and without a username / password authentication.

4.3 Smart Contract Design

The access control system is based on the smart contract implementation, which features three access types specifically for different healthcare applications. An instance of the access type called 'one time' is used for the kind of situation when limited access is necessary, for example, for external consultations. This solution provides limited time access to healthcare professionals for very specific moments such as during a patient's treatment period. Permanent staff members with continuous access to patients' records are assigned access-until-revoked. In order to facilitate role based access control, smart contract has a mapping of user addresses and their associated access rights. Every change which modified the access along has been recorded as a blockchain event, leaving an immutable trail of all access changes. We have functions to grant access, revoke access and validate access requests with the corresponding access controls and validation checks.

4.4 Cloud Infrastruture

AWS cloud infrastructure is designed to offer secure storage with secured access to health records. The S3 bucket is the main storage solution enabled with right encryption settings and policies for access. Access to the S3 bucket is direct and blocked, and all requests are directed through CloudFront for secure content delivery, and URL signing. As intermediaries between the blockchain and the cloud infrastructure, the Lambda functions perform signature verification and access validation. Each time a user requests access to a health record, the Lambda function checks the request signature, interacts with the blockchain to query the requestor if they have access rights, and return signed URLs, good for a limited time, for the authorized requests. This design also keeps potential security risks as low as possible by ensuring that even if a signed URL is compromised its validity period is limited.

4.5 Security Architecture

Multiple layers of security protection for the confidentiality and integrity of health records are implemented using the security architecture. All modifications to the access control at the blockchain level required cryptographic signatures from allowed addresses so that the access permissions cannot be changed unauthorized. Because of this, all state changing operations are checked before they are executed by the smart contract for transaction origins and access rights. AWS best practices for securing sensitive data are implemented by the cloud security layer. This is, like, server side encryption for S3 objects, IAM roles with least privilege, etc, CloudFront signed URLs for secure content delivery. Lambda functions run in VPCs with limited network access and everything communicates with each other over encrypted channels.

4.6 Access Control Flow

In the beginning, the flow of the access control starts when a user makes request access to a health record through the web interface. The user's Web3 wallet then creates a cryptographic signature of the request, and sends the request and the requested object identifier off to the Lambda function. The Lambda function checks the validity of the signature and requests access to the smart contract before this user can access to this resource. After validation is successful, the Lambda function returns a signed URL that leads to the requested document via CloudFront, and the URL stays active for a relatively short period (15 minutes). The design is such that access is secure, but also traceable, with each step of the process logged to the blockchain for audit purposes.

4.7 Performance Considerations

The system architecture has load variations and must have high performance and security. CloudFront is used for content delivery using edge locations, decreasing document access latency. Its high throughput and low transaction cost makes it a natural fit for handling changes to access control as they are frequent, without affecting performance. The memory and power of the Lambda functions have been set up to be able to handle each request concurrently efficiently. At different levels, the system implements caching strategies including using CloudFront to cache documents that are accessed frequently,

and smart contract state caching to mitigate the overhead incurred during the blockchain queries.

5 Implementation

In the implementation phase of this research project, the smart contracts are being implemented and deployed; cloud services are being configured, and other components are being integrated, to ultimately form a secure and efficient health records access control system. This part describes the implementation process of each of this major components and their interactions as seen in Figure 3.

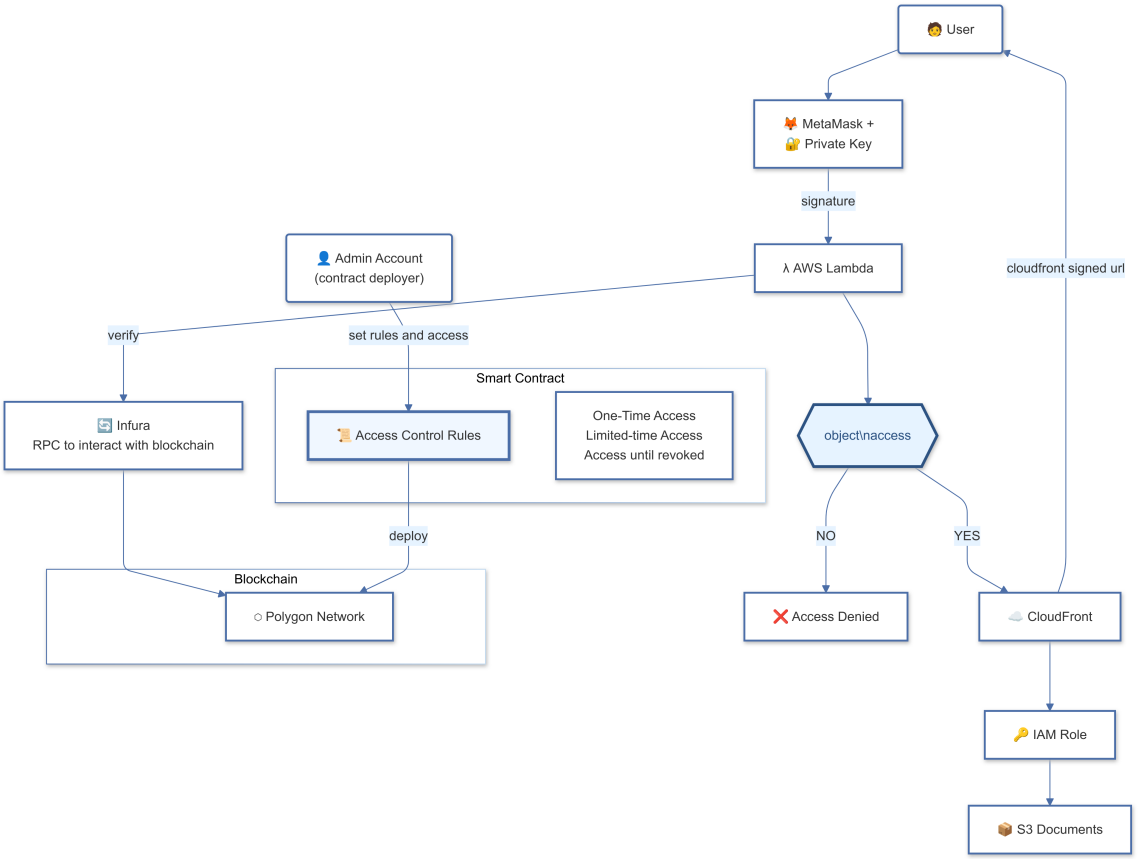

Figure 3: Proposed Implementation Workflow

5.1 Smart Contract Deployment

The start of the implementation was the deployment of smart contract on Polygon AmoyTestnet by using the Truffle development framework as in Figure 4. The Polygon network was deliberately chosen for lower transaction costs and faster confirmation times than those on the Ethereum mainnet, yet retaining compatibility to Ethereum development tools and standards. While implementing the smart contract, three different access control methods for various usage scenarios in the healthcare field are realized. An imagined state machine is used to perform one-time access functionality which automatically invalidates access after the first successful use. TimeStamp use validation means limited time access i.e., access permission of timestamped files will be expired after a

perspective duration. The persistent access rights include the ability to be revoked only by explicit administrative action. The contract's address and ABI (Application Binary Interface) are stored in AWS Secrets Manager in order to guarantee a secure deployment and integration. This approach securely stores the critical contract information in a single authorized place and allows controlled access to those contracts by backend services that are allowed access.

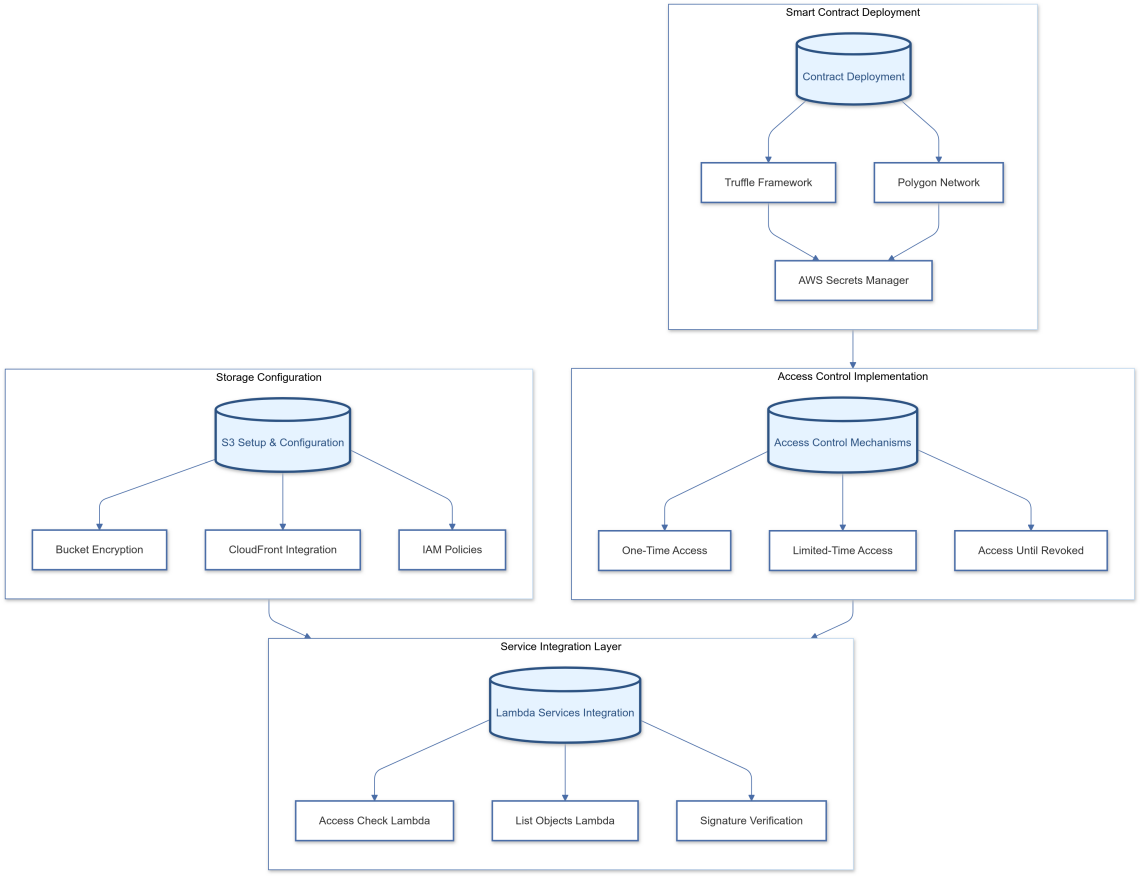

Figure 4: Core Components Implementation

5.2 Storage Infrastructure

An AWS S3 bucket is configured for the storage infrastructure use specifically for health records storage as seen in Figure 4. It includes rich security measures that cover the storage layer. Secure content delivery is implemented such that CloudFront is the only option to make requests to the S3 bucket. Thus, an access to health records is restricted to authentication and monitoring channels. Implementation of bucket permissions falls into bucket policies and IAM roles. Since CloudFront needs access to what's in your S3 bucket, you allow CloudFront access via an Origin Access Identity (OAI) while the bucket policy explicitly denies direct S3 access. With signed URLs, a security measure in place ensures that shouldn't a signed URL get compromised, the attacker can't go around the CloudFront distribution and access the S3 bucket.

5.3 Backend Services

The Serverless Framework was used in the backend implementation where Python was used to set up and deploy Lambda functions as in Figure 4. There are two primary Lambda functions that make up the backend's services: the Access Check Lambda and List Objects Lambda. The Access Check Lambda has a complete verification workflow. The first thing it does is prove the cryptographic signature of all incoming requests so that they are known to originate from a trusted source. It then asks the smart contract on Polygon to verify access rights. It generates time limited CloudFront signed URLs for authorized requests, giving secure, temporary access to the requested health record. The second Lambda function, List Objects, plays a supportive role by implementing a limited set of IAM permissions such that the only objects it has access to are those that it's allowed to see (which, in our case, are objects on the S3 bucket). Using this function allows us to show available documents to the frontend without compromising security according to the principle of least privilege.

5.4 Front-end Integration

The frontend application built using React and integrated with Web3.js allows for separate interfaces for users as well as for administrators. The implementation of the comprehensive access management features behind the admin dashboard it provides allows the administrators to provide and cancel users permission to access. It provides a complete access control management solution by enabling the selection of the object IDs, the user addresses and the access types in the interface.

A streamlined document access workflow is implemented on the user dashboard. The users merely connect their Web3 wallet (mostly MetaMask) and may see the existing documents. The application handles cryptographic signing of the access request transparently so that the security of the connection remains, but the client remains free of this complexity. This deploy uses S3 static hosting with a CloudFront distribution for secure and fast content delivery. Using this approach you get scalability, security, the lowest latency due to CloudFronts global edge network.

5.5 Security Implementation

The security implementation consists of a variety of layers of protection, from the Blockchain layer up to the Cloud infrastructure as described in Figure 5. The implementation is focused at the smart contract level on secure access control mechanisms and state management. All significant state changes are comprehensively event logged to an immutable audit trail of access modifications for the smart contracts, which also undergo rigorous security audits.

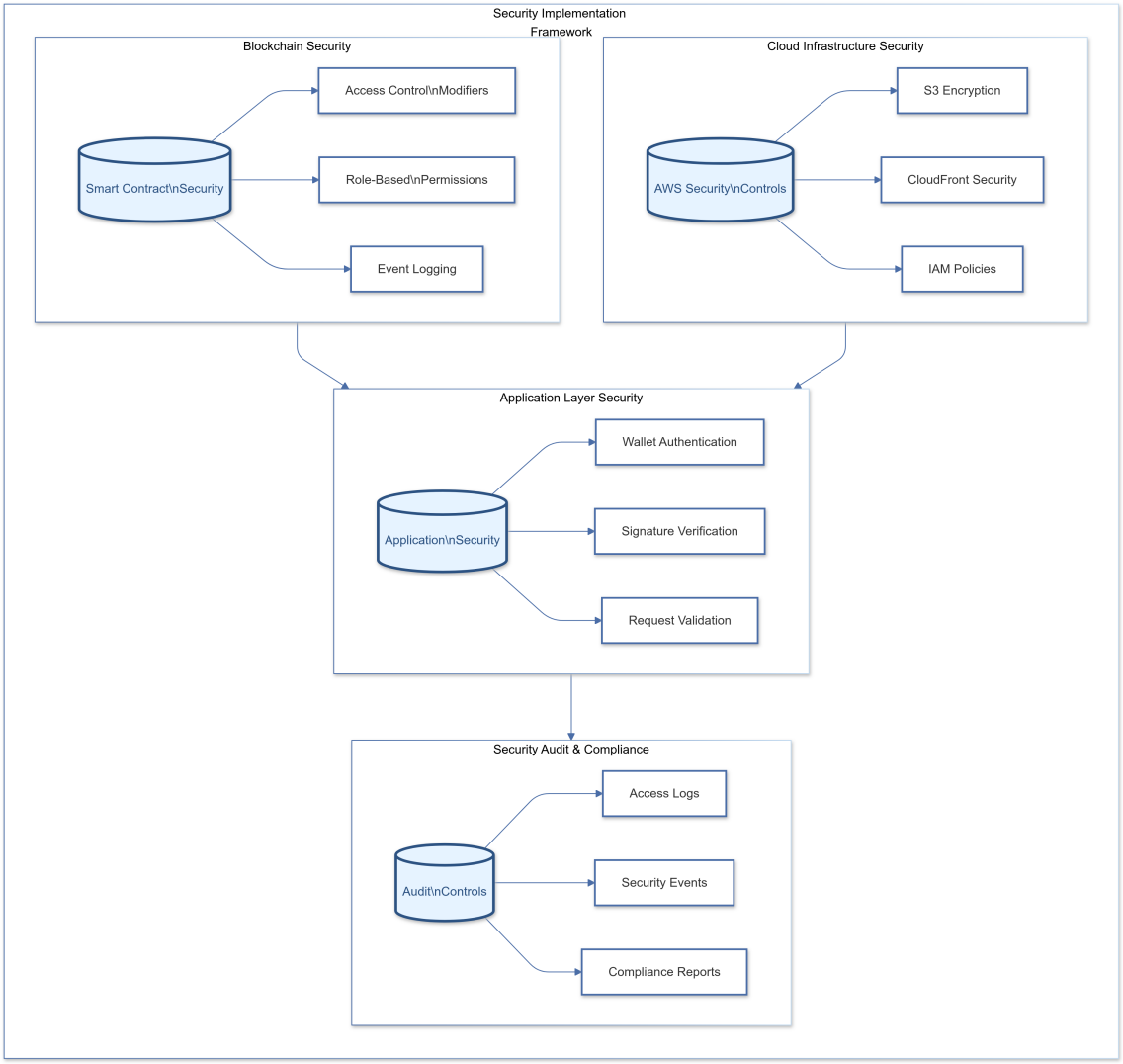

Figure 5: Security Implementation

This process of signature verification is what we can consider as the critical security component of this system since we need to make sure that all the access requests are coming from the only wallet address (or multiple wallet addresses in case of a multi sig wallet) that is authorized to make access requests. It then uses the standard Ethereum cryptographic functions to verify signatures, and checks that the recovered address from the signature matches the address stored inside the smart contract. It thwarts request forgery, and provides all such access requests with a nonrepudiation.

Health records can have an additional layer of security when implemented with CloudFront signed URLs, giving time limited access. These URLs are generated by the implementation using carefully-controlled expiration times (15 minutes is common) to provide a userable yet secure result. Specifically, each signed URL is cryptographically bound to both the specific resource, and the time window, making URL manipulation or reuse impossible.

5.6 Core Implementation Workflow

The steps of its implementation are followed according to a precise operation sequence. If access is granted to administrators, a blockchain transaction is triggered which then updates the smart contract local state. This is included with the parameters of object identifier, user address, and access type for this transaction. The use of smart contract in this architecture allows us to run validation checks before actually committing the change in the state and guarantee the integrity of the access permissions. The requests for user access undergo a multi-step validation. The first thing the Access Check Lambda does is to verify the authenticity of the signature obtained. So it then queries the smart contract, which validate the user has access rights to the requested object. The resulting signed URLs are generated only after both request authenticity and access permissions validity verification occur in this two-phase verification.

Due to the fact that the smart contract implements immediate effect, the access revocation process is instant too. If an administrator revokes the user, the smart contract state will be updated immmediate and any subsequent the user access request which the the administrator revokes, will fail the permission check. With this immediate revocation capability, users are assured the highest level of control over resources and security, something that is necessary in the healthcare settings, when it is possible that access should be terminated quickly.

5.7 Development Environment

The figure 6 shows the development environment which integrates the health records access control system development components and their interactions. The architecture is divided into three main development environment such as Smart Contract Development, Backend Development and Frontend Development each having set of specialized tools and frameworks at the core. The environment for development of Smart Contract uses Truffle Suite and Solidity to develop contract and Backend Development utilizes Serverless Framework and lambda function. React applications with integration of Web3 are implemented in the Frontend development environment, which ensures user friendly interactions over wallet connection.

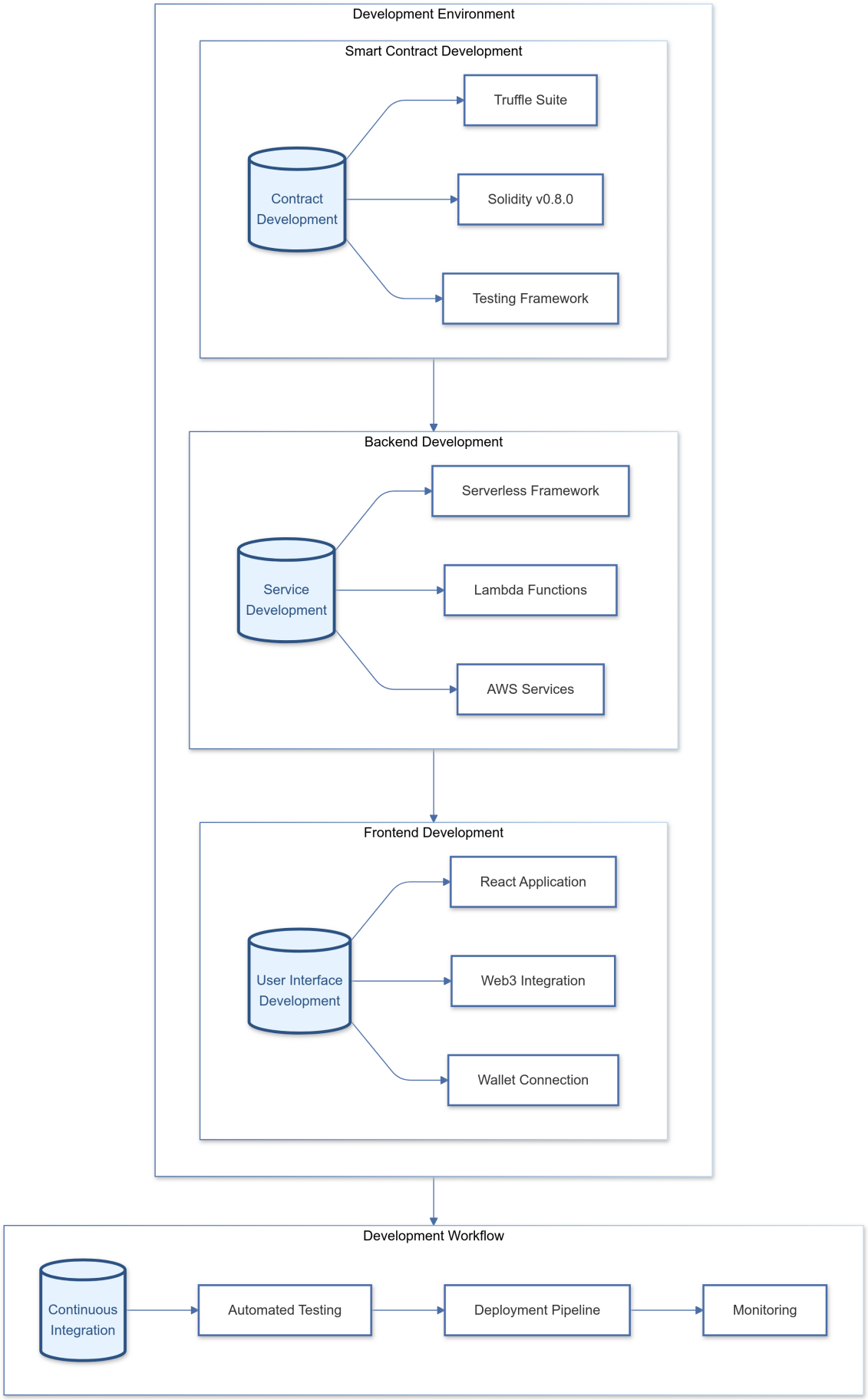

Figure 6: Development Ennvironment

The architecture also shows the workflow portion that continues to deliver through continuous integration and deployment pipeline to maintain the system's reliability and security. The start of this pipeline consists of automated testing procedures used to test individual components and the overall functionality of such components when combined together. Finally, after testing the smart contracts pass, change in the smart contracts, backend services, and front end applications get hit. The workflow ends with a monitoring phase that supplies ongoing inspection of the system's performance and security metric to feed back to future development iterations. This approach to the system is cyclical and maintains the system to be robust and adaptive with the strict security requirements that healthcare data management requires.

5.8 Performance Monitoring

The implementation has included a few performance optimizations. Since smart contract functions are coded into gas efficiency, transaction costs on the Polygon network is reduced. Data structures are efficiently used and state changes are minimized to enable trading of costs for gas while keeping implementation functionality intact. The caching in CloudFront is set up to balance security vs performance. Signed URLs prevent cache sharing between users, but the implementation uses edge cache for static frontend assets and frequently accessed public resources. These Lambda functions are configured with right memory allocaton and timeout settings for concurrent requests handling. It includes complete capabilities for monitoring and maintenance. Lambda function portability is tracked by metrics regarding the CloudWatch function performance, API Gateway requests, and CloudFront distributions. Blockchain event listeners are used to monitor smart contract events for notifications on access control operations in real time.

5.9 Implementation Benefits

Through the integrated approach, the implementation achieves a few key benefits. CloudFront acts as an intermediary for any S3 access, it gives both security and performance benefits. These signed URLs still maintain security through CloudFront's global edge network, and reduce latency for document access. Unlike many other solutions this architecture blocks direct S3 access, which prevents attack vectors such as URL manipulation or bucket enumeration attempts. As a result, access control management is integrated to the blockchain, which makes its transparency and immutability easy. All access grants, revocations and access attempts are being logged unto the blockchain, an audit trail that cannot be tampered or deleted. Especially in the healthcare domain, where logging access requirements are very stringent and audit capability is necessary for compliance, this ability to be transparent turns out to be very valuable.

6 Evaluation

6.1 Performance Metrics Analysis

The system is evaluated for its performance on concurrent requests and it effectively works with both scenarios without signature and blockchain verification and with signature and blockchain verification.

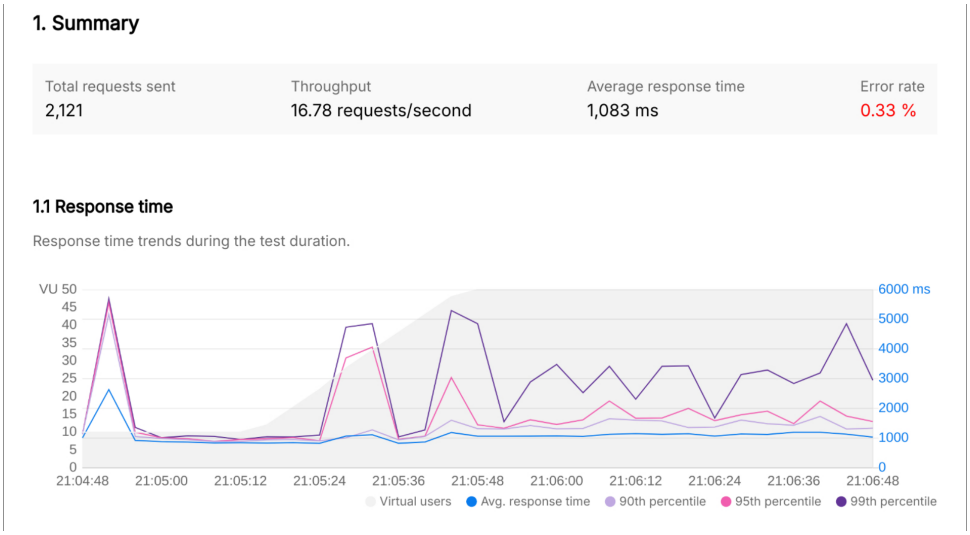

Figure 7: concurrent requests (Ramp Up) Load testing without blockchain verification

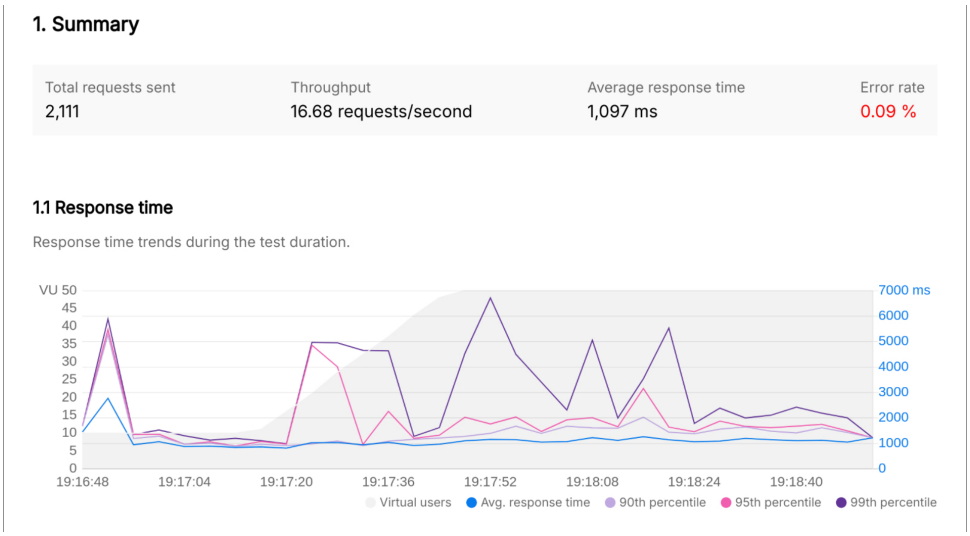

Figure 8: concurrent requests (Ramp Up) Load testing with blockchain verification

Figure 7 and Figure 8 shows the comparison between performance of api calls without signature and blockchain verification and with signature and blockchain verification.

Metric	Not Verified	verified
Total Requests Sent	2,121	2,111
Throughput (requests/second)	16.78	16.68
Average Response Time (ms)	1,083	1,097
Error Rate (%)	0.33%	0.09%
90th Percentile Response Time	˜2,000 ms	˜2,500 ms
95th Percentile Response Time	˜3,000 ms	˜3,500 ms
99th Percentile Response Time	˜4,500 ms	˜6,000 ms

Table 1: Performance Comparison Between Tests With and Without Signature and Blockchain Verification

In signature and blockchain verification in Figure 8 there is little increase in latency at higher percentiles which highlights the additional latency caused by cryptographic verification of signature and the access check done from the blockchain rpc call. This increase in latency is minimal when compared against the security and transparency these checks provide. The throughput remains nearly identical in both tests. Thus hybrid architecture balances the performance while maintaining security and transparency of the system.

6.2 Security Performance

The system architecture possesses a signature verification mechanism of access attempts, making it robust for security due to Web3 wallet integration. The permissions are managed through smart contract-based access control rules that enforce access into three distinct types: one-time access (where permission is only valid for a single request), timebound access (where permissions are only valid for a specified temporal duration), and permission that is valid until revoked (where permissions last until explicitly cancelled). Testing showed that each type was perfectly enforced. Because the blockchain is unchangeable, any changes in access are permanently recorded, and there is an open, tamperproof audit trail. Testing proved that the cryptographic security layer has been effective.

Security Component	**Implementation**	**Effectiveness Rating**
Signature Verification	Web3 Wallet Integration	High
Access Control	Smart Contract Rules	Very High
URL Security	CloudFront Signed URLs	High
IAM Security	Least Privilege Access	Very High
Audit Logging	Blockchain Events	Very High

Table 2: Security Feature Evaluation

6.3 User Interface Analysis

Our user interface implementation achieves security and usability successfully. Using an admin dashboard, the team has access management capabilities all while keeping it intuitive to use. Through a streamlined interface administrators can manage object permissions efficiently, monitor access patterns and modify access rights. Role based access control guarantees that all administrative functions are strictly separate from operations of normal users to assure security boundaries as well as for an efficient task of management within organization.

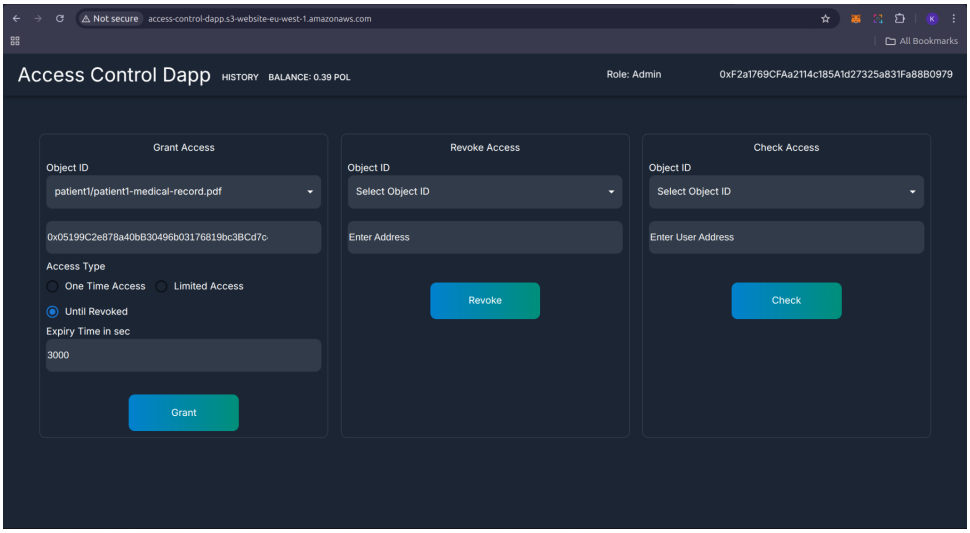

Figure 9: Admin Access UI

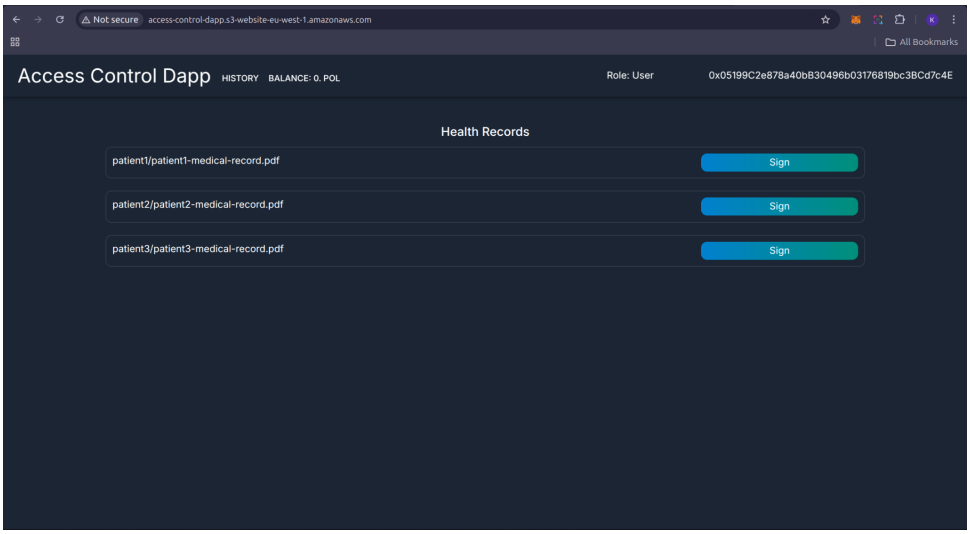

Figure 10: User Access UI

Document access request handling user interface showed outstanding usability. Through a simple workflow, users can see available documents, request access and receive secure access URLs. We integrate with Web3 wallets (specifically MetaMask) to offer secure authentication without sacrificing smooth user experience. The transparent automatic handling of cryptographic signatures and access validation helps to reduce end user technical complexity while keeping high security measures.

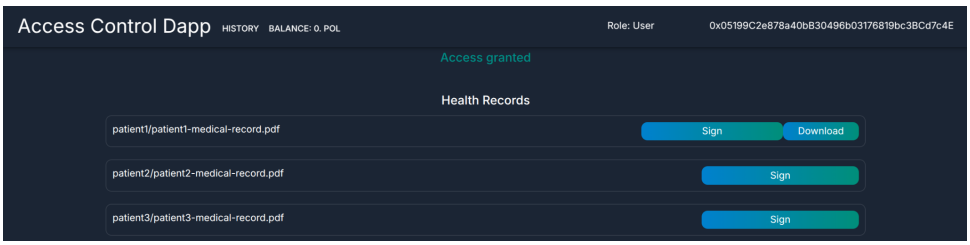

Figure 11: CloudFront Pre-signed URL for S3 Object Download

6.4 Compliance Standards

Relevant security and audit functionality allow this system to comply with HIPAA requirements. The transactions that log every access request, modification, and attempted access on the blockchain are a full audit trail to verify compliance. It includes protected health information, it ensures that rolebased access control, cryptographic authentication, and secure content delivery are exploited to keep only authorized personnel accessible to the information. For temporary access, CloudFront uses signed URLs, and for least privilege access, IAM roles are used. This provides a powerful security framework under which that sensitive health records are protected while they are accessible to authorised users.

Requirement	**Implementation**	**Compliance Status**
Access Control	Role-based + Blockchain	Compliant
Audit Trails	Blockchain Events	Compliant
Data Encryption	S3 + CloudFront	Compliant
Authentication	Web3 + IAM	Compliant

Table 3: HIPAA Compliance Verification

6.5 Discussion

When compared to the base paper of Chinnasamy et al. (2023), we found the following major enhancements and differences. Though both works focus on a secure access to health records using blockchain technology, our implementation makes advancements. The base paper mainly relies on the Ethereum blockchain for its data storage while using IPFS and implements the basic access control through smart contract. However, our solution utilizes the Polygon network, which is more scalable and has lower transaction fees, while still being an Ethereum compatible chain. To further strengthen the security architecture, we have added an extra layer on top of the blockchain based access control, by using CloudFront signed URLs with very short expiry times. In addition to the more complex access model, our system proposes a more sophisticated granular access control mechanism with three distinct access types (one time, limited time and until revoked). Moreover, we have written our backend handling URL generation and access verification in Lambda functions which increases scalability and maintenance. We evaluate our performance and demonstrate lower latency (763ms for 20KB files) than their implementation, and we provide thorough security such as IAM roles, S3 encryption, and signature verification. In addition, our solution has a better audit trail system, with immutable blockchain logging and real time monitoring, that complies better with regulations such as HIPAA as applicable in healthcare.

7 Conclusion and Future Work

In this research, a novel approach for securing health records access is proposed, by integrating blockchain technology with cloud services which brings with large improvement in security and transparency of healthcare data management. The supported implementation merges Polygon blockchain's immutable access control with AWS's capable cloud

infrastructure successfully creating a HIPAA compliant system with granular access management. The research is significant to the field in that it introduces security by the use of Web3 wallets, addresses the traditional centralized system limitation, and provides a transparent audit trail for every access activity. We prove that decentralized authentication can coexist with existing cloud security practices by leveraging smart contracts for access control and CloudFront for secure content delivery while preserving usability and performance.

From this work, there are several promising avenues for future research. Improvements could be made to achieve better tradeoffs between the accuracy of the information verified in the access control, and the privacy during such verification, possibly by using zero knowledge proofs in the access verification cases. The system could also be extended with integration with standards and interoperability frameworks used in emerging healthcare systems. It may also be of interest for future work to extend the realm of artificial intelligence to include the implementation of techniques to detect anomaly in access patterns to yet another layer of security, the behavioral layer. Second, research can expand the solutions and optimizations to lower transaction costs and improve performance for larger scale healthcare networks. More importantly, we believe these enhancements will propel the field of secure, transparent health record management and preserve the integrity of the core principles of both privacy and accessibility simultaneously.

References

Ahmad, A., Saad, M., Al Ghamdi, M., Nyang, D. and Mohaisen, D. (2021). Blocktrail: A service for secure and transparent blockchain-driven audit trails, *IEEE Systems Journal* **16**(1): 1367–1378.

Al Neyadi, D., Puthal, D., Dutta, J. and Damiani, E. (2023). Role-based access control in private blockchain for iot integrated smart contract, *IFIP International Internet of Things Conference*, Springer, pp. 227–245.

Alencar, F. C., Ferreira, C. H. and Filho, D. L. (2024). Developing decentralized applications: A framework approach on blockchain networks, *Proceedings of the 20th Brazilian Symposium on Information Systems*, pp. 1–10.

Austria, P., Kim, Y. and Jo, J.-Y. (2024). Deauth: A decentralized authentication and authorization scheme for secure private data sharing, *Computer Networks and Communications* pp. 1–46.

Badirova, A., Dabbaghi, S., Moghaddam, F. F., Wieder, P. and Yahyapour, R. (2023). A survey on identity and access management for cross-domain dynamic users: issues, solutions, and challenges, *IEEE Access* .

Bakhtiary, V., Mirabi, M., Salajegheh, A. and Erfani, S. H. (2024). Combo-chain: Towards a hierarchical attribute-based access control system for iot with smart contract and sharding technique, *Internet of Things* **25**: 101080.

Baviskar, C. R. (2022). *Cloud based automated encryption approach to prevent S3 bucket leakage using AWS Lambda*, PhD thesis, Dublin, National College of Ireland.

Chinnasamy, P., Albakri, A., Khan, M., Raja, A. A., Kiran, A. and Babu, J. C. (2023). Smart contract-enabled secure sharing of health data for a mobile cloud-based e-health system, *Applied Sciences* **13**(6): 3970.

Habib, G., Sharma, S., Ibrahim, S., Ahmad, I., Qureshi, S. and Ishfaq, M. (2022). Blockchain technology: benefits, challenges, applications, and integration of blockchain technology with cloud computing, *Future Internet* **14**(11): 341.

Hao, J., Huang, C., Tang, W., Zhang, Y. and Yuan, S. (2021). Smart contract-based access control through off-chain signature and on-chain evaluation, *IEEE Transactions on Circuits and Systems II: Express Briefs* **69**(4): 2221–2225.

He, B., Mirchandani, P. and Yang, G. (2023). Offering custom products using a c2m model: Collaborating with an e-commerce platform, *International Journal of Production Economics* **262**: 108918.

Joshi, P. and Kumar, A. (2020). A novel framework for decentralized c2c e-commerce using smart contract, *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, IEEE, pp. 1–5.

Kamboj, P., Khare, S. and 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.

Khande, R., Rajapurkar, S., Barde, P., Balsara, H. and Datkhile, A. (2023). Data security in aws s3 cloud storage, *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, IEEE, pp. 1–6.

Li, X., Zheng, Z. and Dai, H.-N. (2021). When services computing meets blockchain: Challenges and opportunities, *Journal of Parallel and Distributed Computing* **150**: 1–14.

Lin, I.-C., Yeh, I., Chang, C.-C., Liu, J.-C., Chang, C.-C. et al. (2024). Designing a secure and scalable data sharing mechanism using decentralized identifiers (did)., *CMES-Computer Modeling in Engineering & Sciences* **141**(1).

Liu, Y., Zhao, B., Zhao, Z., Liu, J., Lin, X., Wu, Q. and Susilo, W. (2024). Ss-did: A secure and scalable web3 decentralized identity utilizing multi-layer sharding blockchain, *IEEE Internet of Things Journal* .

Madine, M., Salah, K., Jayaraman, R., Al-Hammadi, Y., Arshad, J. and Yaqoob, I. (2021). appxchain: Application-level interoperability for blockchain networks, *IEEE Access* **9**: 87777–87791.

Nair, R., Zafrullah, S. N., Vinayasree, P., Singh, P., Zahra, M. M. A., Sharma, T. and Ahmadi, F. (2022). Blockchain-based decentralized cloud solutions for data transfer, *Computational Intelligence and Neuroscience* **2022**(1): 8209854.

Oliva, G. A., Hassan, A. E. and Jiang, Z. M. (2020). An exploratory study of smart contracts in the ethereum blockchain platform, *Empirical Software Engineering* **25**: 1864–1904.

Petcu, A., Pahontu, B., Frunzete, M. and Stoichescu, D. A. (2023). A secure and decentralized authentication mechanism based on web 3.0 and ethereum blockchain technology, *Applied Sciences* **13**(4): 2231.

Prasad, S. N. and Rekha, C. (2023). Block chain based ias protocol to enhance security and privacy in cloud computing, *Measurement: Sensors* **28**: 100813.

Rana, S. K., Rana, A. K., Rana, S. K., Sharma, V., Lilhore, U. K., Khalaf, O. I. and Galletta, A. (2023). Decentralized model to protect digital evidence via smart contracts using layer 2 polygon blockchain, *IEEE Access* .

Renu, S. A. and Banik, B. G. (2021). Implementation of a secure ridesharing dapp using smart contracts on ethereum blockchain, *International Journal of Safety and Security Engineering* **11**(2): 167–173.

Song, L., Li, M., Zhu, Z., Yuan, P. and He, Y. (2020). Attribute-based access control using smart contracts for the internet of things, *Procedia computer science* **174**: 231–242.

Sucharitha, G., Sitharamulu, V., Mohanty, S. N., Matta, A. and Jose, D. (2023). Enhancing secure communication in the cloud through blockchain assisted-cp-dabe, *IEEE Access* .

Xiao, Y., Zhou, C., Guo, X., Song, Y. and Chen, C. (2022). A novel decentralized e-commerce transaction system based on blockchain, *Applied Sciences* **12**(12): 5770.

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.

Zhou, Z., Wang, M., Yang, C.-N., Fu, Z., Sun, X. and Wu, Q. J. (2021). Blockchain-based decentralized reputation system in e-commerce environment, *Future Generation Computer Systems* **124**: 155–167.