BFS-ZAT: Blockchain-Enabled Federated Security with Zero-Knowledge Adaptive Trust in Multi-Cloud Environments

Prabakaran K.* and Sivakumar B.

Department of Computing Technologies, School of Computing, College of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur, India
Email: pk8923@srmist.edu.in (P.K.); sivakumb2@srmist.edu.in (S.B.)
*Corresponding author

***Abstract*—The proposed system introduces a blockchain-federated security system and a zero-knowledge adaptive trust system that would allow secure and auditable access control on multi-cloud platforms. There is a need to overcome the issues of heterogeneity of infrastructure, dynamic threats, and uninteroperable and privacy-sensitive trust management in securing access in a multi-cloud environment. The suggested model incorporates a federated identity management system, which combines biometric authentication and cryptographic identity that runs on blockchain to provide non-repudiable identity management. The system involves a blockchain trust authority that handles cryptographic keys and a dynamic on-chain trust score, which facilitates privacy-preserving authentication employing zero-knowledge proofs. Hybrid cryptography provides secure data access and sharing by combining fine-grained access control with homomorphic encryption, based on attribute-based encryption and homomorphic encryption, respectively. The combination of a blockchain and differential privacy allows executing queries in an auditable and privacy-guaranteed manner with clear data owner policies. To achieve cross-cloud interoperability, a trust-aware blockchain-aided proxy re-encryption scheme permits protected information sharing with adjustable trust levels, and trust re-assessment and blockchain-aided key refreshing schemes are resistant to dynamic threats. Experiments conducted on a federated multi-cloud testbed have shown that the proposed framework obtains the following security guarantee and efficiency, as compared to existing schemes authentication delay of 180 ms, a transaction throughput of 220 tps, and an accuracy of trust evaluation of 94%. Results of the experiment indicate that the proposed framework is scalable, privacy-protecting, and reliable in terms of access control to federated multi-cloud systems.**

***Keywords*—multi-cloud environments, blockchain trust authority, adaptive trust scoring and attribute-based encryption**

I. Introduction

In recent years, the quick evolution of cloud computing has changed the way infrastructures store, manage, and process digital data. Currently, businesses, on in-house servers instead of relying on, computational resources, scalable storage, and application services to provide on-demand access, utilize cloud platforms [1]. This transformation has improved operational efficiency and significantly decreased infrastructure expenses. As organizations remain complex to grow, many of them are accepting multi-cloud infrastructures, and they have workloads across various Cloud Service Providers (CSPs) distributed. This strategy enhanced scalability, more fault tolerance, increased redundancy, and allowed for different vendors to choose specialized services, offering various benefits. Furthermore, various providers using enterprises avoid vendor lock-in and attain better cost optimization and performance [2].

However, multi-cloud environment flexibility and resilience to provide, and also trust management, privacy, and interoperability-related new issues have been introduced. Every CSP has its own security policies, access control mechanisms, and authentication systems that are maintained; thus, clouds in between seamless and secure interactions are difficult [3]. Many systems are still used in centralized trust models, and multi-cloud infrastructures are struggling to cope with decentralization. This leads to issues, such as data breaches, insider threats, and unauthorized access, especially with data and applications, when spread across multiple independent environments [4].

These challenges are to be overcome. Blockchain technology has its decentralized, transparent, and tamper-proof characteristics as a promising solution due to its emergence. A blockchain, as a distributed ledger are functions to securely and immutably record transactions, and a central authority is eliminated [5]. Multi-cloud systems are integrated with blockchain, and various providers secure data exchange, verifiable access control, and transparent trust management is implemented. Through its consensus mechanisms and cryptographic integrity, every transaction or access request registered is audited on the blockchain, thus accountability is

Manuscript received November 24, 2025; revised January 13, 2026; accepted March 2, 2026; published May 25, 2026.

enhanced and manipulation or data tampering risks are reduced [6].

Despite these benefits, existing blockchain-based cloud security frameworks still have various limitations to address. Among them, most rely on static trust models; the users and Cloud Providers (CPs) fail to adapt to changing behaviors [7]. Additionally, many existing solutions for authentication or data sharing exist when there are privacy-preserving mechanisms that protect sensitive information. For instance, blockchain transparency is ensured, and also carefully designed so that user identities or access patterns are not inadvertently revealed. Moreover, compromising confidentiality, multiple providers' data access must be secured, and fine-grained cross-cloud data sharing is limited support [8].

Privacy protection is another important concern. In multi-cloud environments, sensitive data, including healthcare records, financial transactions, or business analytics like critical data mostly across different clouds that need to be processed or transferred. Without cryptographic techniques, such data communication or computation, when it becomes leaked or intercepted, is more vulnerable [9]. Existing encryption protection offers, existing schemes have encrypted data direct computation, which avoids, so they make real-time or federated operations impossible. Similarly, various entities in between secure data sharing with the intention of implementing proxy re-encryption techniques, trust between intermediaries are typically not accounted for, thus data misuse risks are increased. So, among multiple CSPs, secure, dynamic, and interoperable collaboration is confirmable. A comprehensive, trust-aware, and privacy-preserving method has strong requirements. Such a system, maintaining transparency and accountability while protecting user privacy, integrates adaptive trust evaluation, federated identity management, and advanced cryptographic techniques [10]. This study presents a novel Blockchain-enabled Federated Security with a Zero-knowledge Adaptive Trust (BFS-ZAT) framework for multi-cloud environments using secure, transparent and privacy-preserving access control.

A. Novelty and Contribution

In this research, key findings are presented and outlined as follows:

- Unified Multi-Cloud Federated Identity (MCFI): The unified MCFI introduces the biometric features and cryptographic keys that allow cross-cloud authentication to be integrated. The identity of the blockchain is unchanged, to stored, duplication, and unauthorized modifications are prevented.
- Adaptive Trust Scoring (ATS): An ATS is implemented for both users and cloud providers according to historical behavior, Service-Level Agreement (SLA) compliance and anomaly detection. This real-time, content-aware trust assessment for traditional fixed access controls beyond security is enhanced.
- Privacy-preserving Zero-Knowledge Proofs (ZKP): ZKP allows users and cloud providers to authenticate to combine, without revealing important evidence, privacy-preserving in federated cloud environments, and trust-aware authentication is enabled.
- Hybrid encryption for secure data sharing: A novel hybrid encryption framework is introduced, Homomorphic Encryption (HE) for secure computations on encrypted data with Attribute-Based Encryption (ABE) for fine-grained is integrated, granular, policy-driven access control that guarantees only authorized users decrypt outcomes, allowing secure, privacy-preserving cross-cloud data sharing.
- Trust-Aware Blockchain-Assisted Proxy Re-Encryption (T-AB-APRE): A novel T-AB-APRE scheme is presented for ciphertext re-encryption that guarantees only trusted proxies manage sensitive data. Blockchain auditing logs all operations which support transparency, accountability and dynamic key management across clouds.

The contribution of the proposed framework, BFS-ZAT architecture, is securing access control in federated multi-clouds in a secure, transparent, and privacy-preserving way. The suggested framework integrates identity management, adaptive trust, authentication, encrypted data sharing, and cross-cloud interoperability into one blockchain-based framework. MCFI system, biometric-secured cryptography credentials are anchored by the blockchain, and a blockchain trust authority initialises cryptographic variables and maintains a record of auditable trust scores of cloud service providers. ZKP-based authentication and blockchain-enforced rules of data owner policies ensure access control to enable credential-less and trust-aware authorization. The use of hybrid encryption based on homomorphic encryption and attribute-based encryption of privacy-preserving query processing and data sharing to ensure secure confidentiality and fine-grained access control is offered. In order to achieve interoperability, a T-AB-APRE scheme is used to support secure ciphertext transformation, trust-sensitive proxy selection, and active revocation using multiple clouds. The BFS-ZAT framework is a framework that offers an end-to-end adaptive trust platform that incorporates federated identity, privacy-preserving authentication, encrypted computation, and auditable multi-cloud data exchange.

B. Threat Model and Attack Scenarios

The threat model assumes a poly-time adversary, who can eavesdrop, intercept, replay and modify messages over the public and inter-cloud networks. The opponent can attack the individual cloud service providers or the user devices, but is not able to attack most of the blockchain consensus nodes. Critical assets are biometric templates, cryptographic keys, trust scores, access policies as well as audit trails. The blockchain is believed to provide immutability, integrity and availability on an honest-majority consensus model.

The attacker could perform identity impersonation attacks by creating fake credentials or replaying authentication messages on various cloud service providers. Trust manipulation attacks could focus on adaptive trust values to maliciously increase access privileges. Data inference attacks could leverage query

responses to obtain confidential data from encrypted data sources. Cross-cloud data leakage attacks could try to steal data without authorization during interoperability and re-encryption operations. Insider attacks could try to manipulate access logs or revoke policies, which can be prevented by using blockchain-based auditing.

The rest of the paper is organized as follows: Section II reviews the literature survey in multi-cloud environments. Section III discusses the system model. Section IV details the proposed BFS-ZAT protocol. Section V highlights the experimental results. Section VI concludes the paper and presents future work.

II. Literature Survey

This section reviews the existing routing methods in multi-cloud environments, highlighting their strengths and limitations in addressing the key challenges related to their trust re-evaluation and blockchain-based key renewal, dynamically adjusting trust scores, and enforcing adaptive access policies.

Karnik *et al*. [11] presented an Efficient Multi-Cloud Storage (EffMCS) framework, which improves security and performance. This method ensures data integrity 256-bit hash value using a Secure Hash Algorithm-256 (SHA). Moreover Rivest-Shamir-Adleman (RSA), Advanced Encryption Standard (AES), and Elliptic Curve Cryptography (ECC), using the combination of encrypted data, provided effective key management and maintained data confidentiality. This affects the data security, storage vulnerabilities are reduced, and secure Multi-Cloud Storage (MCS) systems for access performance are improved. Bansal *et al*. [12], encrypting user data using Two-fish 256-integrative Symmetric Key Cryptography (Twofish256-SKC) for a multi-cloud structure. AES and Twofish, the strengths were combined, the algorithm created a robust hybrid cryptographic approach, which is more secure. These methods increased security, enhanced multi-cloud information security, and were very useful in confirmation.

Chauke *et al.* [13] developed a Software-Defined Network (SDN) and a Machine Learning (ML) algorithm influenced by an adaptive threat identification method. This combination was multi-cloud environments, real-time anomaly detection, dynamic network reconfiguration and proactive threat mitigation was enabled. This method's critical impacts were accurately handled, and overall cloud data scalability was improved. Yang *et al*. [14] presented a multi-cloud environment used for an efficient ABE scheme with Data Security Classification (ABE-DSC). Data owners' encryption phase in different CSPs was stored, and security levels were basically divided into two parts, improving the security of outsourcing data. Furthermore, Ciphertext-Policy Attributed-Based Encryption (CP-ABE) was used to Data User (DU) to provide granular access control. The results showed that the computational overhead of the method was low.

Blockchain-based for MCS, an auditable deduplication scheme was developed by Jin *et al.* [15]. This system, blockchain technology, and bilinear pairing ecosystems were used to create a data duplication mechanism, saving storage space, and data integrity was checked. Safety analysis shows that schemes were achieving the expected security goals. The result showed that the scheme was feasible and provided high audit efficiency.

Bharot *et al*. [16] presented a cloudlock framework, authenticated users and secure data storage in a multi-cloud environment, and employed a secure data sharing for a hybrid ChaCha20-Poly1305 encryption mechanism. Additionally, the secret key used for encryption and decryption was shared between the data owner and the actual users using the Elliptic Curve Diffie-Hellman (ECDH) mechanism. This method demonstrated secure data sharing, and various security threats showed strong resilience.

Witanto *et al.* [17] developed a multi-cloud environment for blockchain-based distributed data integrity verification. This was more than a single verifier using multiple validators; enabling data validation increases the validation sampling rate without increasing computation and communication costs. This method protocol achieved minimum time consumption, and multi-verifiers, each verifier computation and communication costs were reduced. Noh *et al*. [18] implemented multi-cloud environments, a resilient and fast block transmission system for Hyperledger Fabric. This method's objective nodes among batch synchronization time, Hyperledger Fabric transaction, by reducing the time of transaction throughput, scalability, and resilience were enhanced. To accomplish this, time-varying multi-cloud environments were blocked for all participating nodes to deliver modules quickly and reliably.

Jebakumari *et al*. [19] presented a multi-tier ELK herd-driven threshold cryptography integrated random forest (MEH-TC+RF). Secure cryptographic keys were used to produce the ELK Herd Optimization (EHO) technique was employed. Encrypt files MEH-TC system, the keys are used, after key shares among trusted resource providers were distributed. The encryption method provided excellent protection, and important performance was achieved. Huang *et al.* [20] presented an industrial internet environment with blockchain indexing for a multi-cloud collaborative data security sharing scheme. This method, keyword-based ciphertext retrieval, was supported, efficient, and secure, tokenized keyword search was allowed. This method has less time overhead, and a robust technique was constructed.

Zeydan *et al*. [21] developed a network management and orchestration to fully involve multiple entities in the management for a decentralized architecture based on Block-Chain Networks (BCNs) and BCN-based Self-Sovereign Identity (SSI). This method CSPs, Vertical Service Providers (SPs), and Mobile Network Operators (MNOs) multi-cloud environment, to manage the service trusted environment was provided. The operations where a shorter timeframe were accomplished in the indicated result. Bhatt *et al*. [22] presented a scalable and secure Multi-Cloud Data Ecosystem Architecture (MCDCA). This method involves data governance strategies,

availability, and security measures for analyzing architectural forms; these are essential for maintaining data integrity and confidentiality in distributed environments. This technique demonstrates maintaining robust, secure, and compliant environments in distributed environments.

The Federated Identity Systems Static and Centralized Trust is the majority of identity and access management systems based on blockchain, which include the use of static credentials and semi-centralized trusting entities. Such frozen trust models do not evolve in response to user and cloud behavioral patterns, which makes them vulnerable to user credentials attacks, insider attacks, and long-term misuse. Weaknesses in privacy-preserving and adaptive authentication is conventional authentication and authorization mechanisms are also likely to expose confidential identity information in the authentication process, and are incapable of incorporating adaptive trust evaluation. Therefore, they provide weak security against inference attacks and dynamic threats in federated multi-clouds. Inadequacy in flexibility to privacy-preserving data sharing is that single-encryption methods are used in most secure data sharing solutions, resulting in a trade-off between the confidentiality of data and dynamic access controls. They are unable to facilitate encrypted calculation and policy-based access to encrypted data on multi-clouds. Insufficient trust-aware interoperability and dynamic revocation is that the majority of cross-cloud data sharing and proxy re-encryption systems do not have any trust-based proxy selection and key revocation. This decreases the trustworthiness of breached or distrusted domains and compromises trusted interoperability between federated clouds. The lack of scalability and auditability of federated multi-cloud security is that the majority of available solutions provide centralized coordination and are not based on a single trust management system using blockchain technology, which makes them less transparent, scalable, and auditable in the case of multi-cloud environments.

In order to address these research gaps, this proposed system introduces the framework of BFS-ZAT that integrates adaptive trust assessment scoring, zero-knowledge authentication, hybrid (HE+ABE) encryption, and trust-aware blockchain-assisted proxy re-encryption within a decentralized federated multi-cloud system.

III. System Model

The proposed framework offers decentralized trust management and secure access control designed for a multi-cloud federated environment that combines blockchain. Users, cloud providers and BTA are the three system models that contain primary entities.

Fig. 1 illustrates the system architecture of the proposed BFS-ZAT framework, where users access multi-cloud services through a blockchain trust authority integrated with ZKP verification and adaptive trust scoring. The framework supports privacy-preserving authentication, tamper-resistant trust management, and dynamic, risk-aware access control across heterogeneous cloud platforms. By separating identity verification from service access and continuously updating trust scores, the system enhances security, interoperability, and auditability in multi-cloud environments while minimizing data exposure.

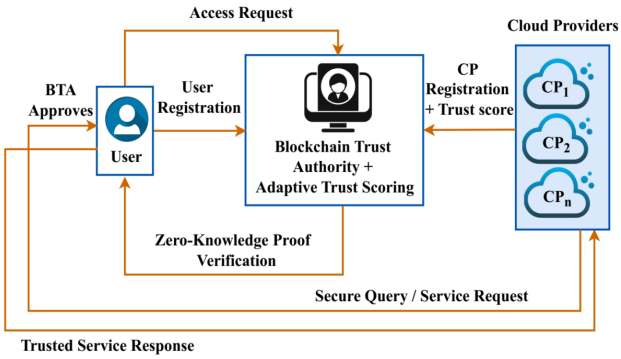

Fig. 1. System model (200 users, 3 cloud providers).

A. Users

Cryptographic keys and biometric attributes are derived from the Federated Identity (FID), which is assigned to every user. The identity is duplicated or prevents impersonation, and the blockchain is an immutable record. Users communicate with cloud providers and make requests for the resources and services.

B. Cloud Providers

Computational and storage services are delivered by several cloud providers that are participating in the federated network. Every CP is enrolled with the BTA and receives a designated first trust score. Every CP is dynamically updating its trust score based on the service quality, reliability, and historical interactions.

C. Blockchain Trust Authority

The BTA functions as the trust anchor of the system. Users and cloud providers must be accountable for registering, issuing the system parameters, and storing trust scores on-chain. BTA utilizes the access trust scoring to validate through access requests and enforces consensus across cloud providers. The advanced cryptographic methods are also supported, like ABE, HE, Proxy Re-Encryption, which ensure secure communication and privacy-preserving operations.

D. System Interactions

Registration phase: The BTA is registered by the users and cloud providers, and their trust anchors and federated identities are stored in the blockchain.

Access phase: When a user requests access to a CP. The ATS mechanism verifies if the user's trust score surpasses the defined threshold.

Authorization phase: The ZKP is utilized to verify the delicate credential's trustworthiness without revealing the request.

Service phase: Secure queries are implemented across cloud providers, trust-aware access, integrity and data confidentiality.

IV. Proposed Methodology

This research presents a novel BFS-ZAT framework designed to offer secure, transparent and privacy-preserving access control in the BFS-ZAT framework. The framework starts with user registration through the MCFI scheme, which combines biometric attributes and a cryptographic key pair that are immutably stored on the blockchain. For cloud providers, trust scores are assigned, and cryptographic parameters for all events for auditability registering on the chain BTA are initialized. When a user requests access, ATS trustworthiness is determined, and then sensitive credentials without exposing authentication are used to allow ZKP verification to be performed. Valid requests progress to privacy-preserving query execution and are secured by Differential Privacy (DP), while access to sensitive data needs explicit approval from data owners, enforced by blockchain-based policy verification and immutable logging. Upon approval secure data sharing is ensured by hybrid HE for secure computation and ABE for fine-grained access control. Moreover, T-AB-APRE facilitates secure and policy-compliant ciphertext transformation across cloud domains. To ensure long-term resilience the framework incorporates continuous trust re-evaluation and blockchain-based key renewal, dynamically adjusting trust scores rotating cryptographic keys and enforcing adaptive access policies. Fig. 2 depicts the overall architecture of the proposed BFS-ZAT technique.

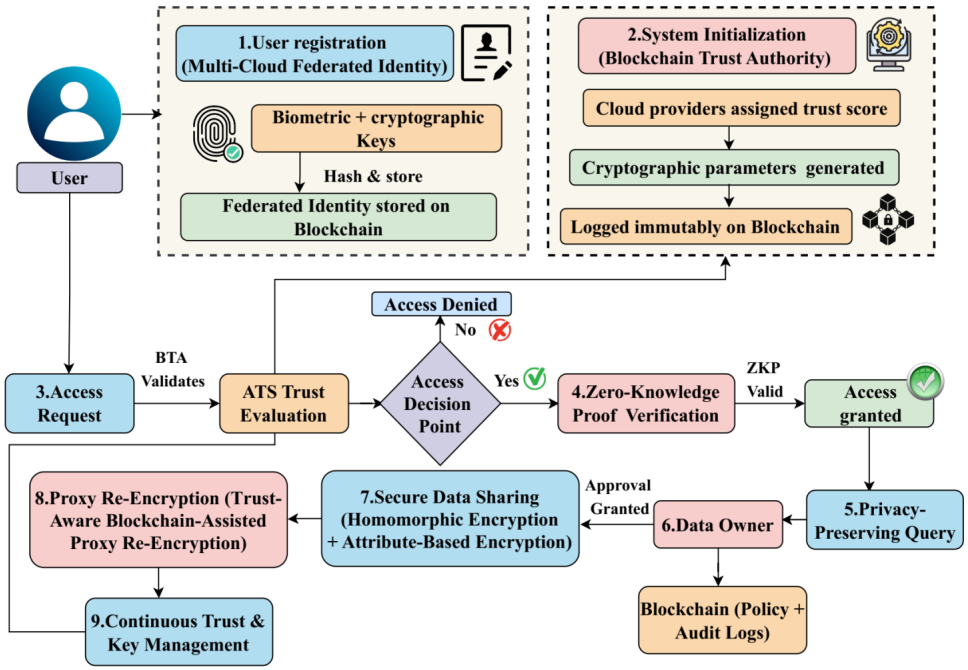

Fig. 2. Overall architecture of the proposed BFS-ZAT technique (200 users).

A. User Registration and System Initialization

This phase focuses on securely onboarding both users and CP, establishing a robust trust framework essential for federated operations. Its primary goal is to create a unified, federated identity that is consistently used across all participating CP to ensure that thrust anchors are stored immutably on the blockchain.

1) User registration with MCFI

The MCFI scheme allows users to register once, and then it helps to access multiple cloud platforms without having to register repeatedly. This procedure integrates biometric verification with cryptographic credentials, ensuring integrity, privacy, and tamper resistance.

During registration, the biometric vector is provided by the user A_u (iris scan and fingerprint), and a cryptographic key pair is generated SN_u, PN_u. To prevent forgery, these credentials are integrated with the blockchain and hashed before being stored.

$$K_u = Hash\left(A_u \| SN_u, PN_u\right) \tag{1}$$

where, K_u denotes the user's hash is stored in the blockchain, A_u represents the biometric vector, and SN_u, PN_u is the user's public key.

A unique federated identity FID_u is then generated to utilize the $MCFI_{Gn}$ function:

$$FID_u = MCFI_{Gn} \cdot \left(K_u, UID_u\right) \tag{2}$$

where, UID_u denotes the user's local identifier. This federated identity will be used to seamlessly authenticate the user across all CP, and it will be validated FID_u with the BTA trust anchor that is maintained.

The various benefits are provided by this method: (i) a single federated identity is enabled through the cross-cloud interoperability, (ii) tamper-proof identity storage on the blockchain, (iii) credentials are never exposed since they are privacy-preservation sensitive, (iv) the foundation for adaptive trust assessment in subsequent

access control. Fig. 3 depicts the Architecture of user registration and FID generation.

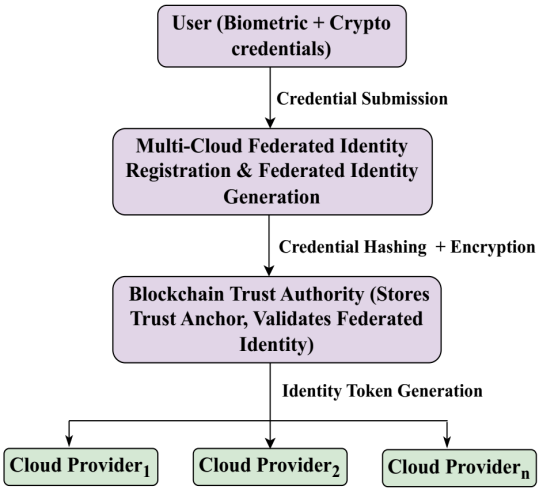

Fig. 3. Architecture of user registration and FID generation (200 users).

2) System initialization via BTA

When users are connected through MCFI, cloud providers are registered and initialized by BTS. Each CP_e is assigned an initial trust score by compliance and historical performance.

$$ST_e^0 = ST_{init} \tag{3}$$

where, ST_{init} denotes the baseline trust constant, ST_e^0 represents the initial trust score assigned to the CP_e registration.

The entire registration events are like a trust score, public key, and provider identity are an indelible entry by the blockchain ledger.

$$Ledg_{BTA} \leftarrow \left\{CP_e, SN_e ST_e^0, timestamp\right\} \tag{4}$$

where, $Ledg_{BTA}$ indicates from BTA that the blockchain ledger is maintained, CP_e is the e^{th} CP in the multi-cloud federated system, SN_e denotes the public key of the CP_e.

ABE, HE, and PRE are further generated in BTA for the system-wide cryptographic parameters, which are distributed safely to the CP.

$$SN_{sy} = KeyGen\left(ABE, HE, PRE\right) \tag{5}$$

where, SN_{sy} denotes the system-wide cryptographic parameter set.

The ATS is dynamically updated using the continuous trust monitoring support for the provider trust score.

$$ST_e^{t+1} = \alpha . ST_e^t + \beta . f\left(behavior, SLA\,compliance, anamalies\right) \tag{6}$$

where, α and β are weighting factors ($\alpha + \beta = 1$) and $f(.)$ is the trust evaluation function, ST_e^{t+1} is the updated trust score of provider CP_e at the next evaluation, ST_e^t is the trust score of provider CP_e at time t.

Every trust update, policy changes, and key renewal are immutably logged on-chain, ensuring non-repudiation, auditability, and regulatory compliance.

An immutable trust foundation, a safe deployment of cryptographic parameters across providers, and a dynamic mechanism for continuous trust reassessment are the components of the initialization phase of the providers.

B. Access Control and Authorization Mechanisms

In the multi-cloud environments, the evolving nature of inside threats, the dynamic workload, and the need for fine-grained policy enforcement make consistent access control mechanisms inadequate. To deal with this, this framework combines ATS with ZKP-based authorization, ensuring that access to cloud resources is secured and privacy-preserving.

1) Access negotiation with ATS

When a user initiates an access request to a CP, the request is not directly granted or denied. Instead, it is assessed by the ATS mechanism. The ATS continuously calculates a trust value for each user, considering behavioral consistency, historical activity, and anomaly detection metrics.

Formally, $T_u(t)$ is the trust score, u denoted by user, t is the time defined as Eq. (7):

$$T_u\left(t\right) = \alpha . K_u + \beta . A_u\left(t\right) + \gamma . B_u\left(t\right) \tag{7}$$

where, K_u denotes the normalized historical reliability score (previous interaction and successful authentications), $A_u(t)$ indicates the behavioral consistency factor at time t, $B_u(t)$ represents the anomaly risk score (anomaly detection models are derived), $\alpha + \beta + \lambda = 1$ are the weight coefficients dynamically used by the system.

A threshold-based policy is enforced as Eq. (8):

$$Acc\left(u\right) = \begin{cases} Granted, & if\ T_u\left(t\right) \geq \theta \\ Denied, & if\ T_u\left(t\right) < \theta \end{cases} \tag{8}$$

where:

- $Acc(u)$: Access decision for user u.
- $T_u(t)$: Adaptive Trust Score (ATS) of user u at time t.
- θ: Minimum trust threshold for access approval.

This ensures that the users will face restricted access or that access is denied, even though their credentials are valid.

2) User authorization via ZKP

Once the ATS authorizes an access request, the user must prove the authorization right without revealing sensitive credentials. To achieve this, ZKPs employs the authentication layer.

Let, x represent the user's private credential, and the verifier CP only knows a commitment $C = f\left(x\right)$, where f denotes a one-way function. Utilize ZKP, the user proves possession x such that:

$$Pr\,o: f\left(x\right) = C \quad without\,revealing\,x \tag{9}$$

This, during verification, privacy preservation means private keys or biometric tokens, similar to evidence, are never exposed and are confirmed. Cross-cloud interoperability is original identification without the need for direct access to federated clouds that authenticate users. Moreover, every ZKP resisting re-attacks, confirmed by random evidence that cannot be reused maliciously. Only if two conditions are met, final authorization is granted:

$$Auth(u)=\begin{cases} Granted, & if\ (T_u(t)\geq\theta)\wedge(ZKP(u)=True) \\ Denied, & otherwise \end{cases} \quad (10)$$

For trust evaluation, the two-step mechanism ATS and ZKP, for privacy-preserving credentials, this is for trusted and legitimate users, ensuring only sensitive multi-cloud resources are accessed.

C. Privacy-Preserving Query Processing and Data Sharing

A federated multi-cloud environment data sharing and query execution is confidentiality, integrity, and accountability it is handled in a way that ensures that. The traditional query processing often reveals sensitive metadata and leaves audit trails incomplete, through this, the enemy inference or internal abuse opportunities are created. To reduce these risks, the proposed framework combines the privacy-preserving query generation, blockchain-based approval and logging, and advanced cryptographic methods like HE and ABE. This guarantees queries remain secret. This query remains confidential, the authorized data owners only approve the access, also all operations are immutably auditable.

1) Privacy-preserving query and transaction generation

When submitting Q query by a user submits, the system changes it into a privacy-preserving representation before the implementation. First, Q is encrypted using a session-specific symmetric key k:

$$Q_{enc}=Enc_k(Q) \quad (11)$$

where, Enc_k denotes the symmetric encryption.

Sensitive metadata (query frequency, resource identifiers) to prevent leakage, and DP is used. Noise is added to the query metadata such that the transformed query Q' satisfies $\in$ \epsilon $\in$ $^{-}$DP:

$$Q'=Q_{enc}+M(0,\sigma^2) \quad (12)$$

where, $M(0,\sigma^2)$ indicates the Gaussian noise. This ensures that adversaries cannot infer important characteristics from query access patterns.

2) Data owner approval

A privacy-preserving query, once created, before execution, should be verified by the corresponding data owner. A blockchain is a consensus mechanism that ensures the access rights of decentralization are verified.

Normally, Q' represents a query request and DO denotes the data owner, then approval is granted only if:

$$Appr(Q')=\begin{cases} 1, & if\ policy(DO)\wedge Auth(User)=True \\ 0, & otherwise \end{cases} \quad (13)$$

where, $policy(DO)$ the access policy is defined by the data owner and $Auth(User)$ represents the authentication proof provided via federated identity. This verification is unauthorized, or a malicious query is executed on the sensitive cloud datasets.

3) Blockchain-based audit logging

All queries, approval decisions, and SLA compliance metrics are irretrievably recorded on the blockchain. These decentralized ledgers serve to provide two primary functions.

Traceability: All the query activities are traced back to their origin, which is accountably ensured.

Auditability: The transparent and tamper-proof logs are supported through regulatory compliance.

Mathematically, every log entry is also referred to as Eq. (14):

$$L=H(Q'||Approval(Q')||SLA||t) \quad (14)$$

where, $H(.)$ denotes the function of a cryptographic hash, t indicates the timestamp. L Immutability ensures that no entities alter the query or approval history.

4) Confidential data sharing using HE and ABE

A query is verified, and once approved, the data apportionment procedure is secured through a hybrid encryption mechanism that combines ABE and HE. This dual method guarantees both confidentiality and fine-grained access control in multi-cloud environments.

a) Homomorphic encryption

Without expressing the underlying plaintext, HE allows safe computing through encrypted data. Let n indicate the plaintext and $C=Enc(n)$ is a ciphertext. For a computational function f, a homomorphism ensures the following:

$$f(Enc(n))=Enc(f(n)) \quad (15)$$

where, n represents the plaintext data, $Enc(n)$ indicates the encryption of n, $f(.)$ denotes the computation function.

This trait of cloud providers ciphertext enables performing calculations directly (filtering, statistical analysis, aggregation), the raw data is never exposed during processing. As a result, the critical information is protected, even when computation is outsourced to multi-cloud infrastructures where it is not reliable.

b) Attribute-based encryption

During the computation, HE data is secured, at the same time, ABE, during data encryption, executes fine-grained access control. In ABE, decryption rights are tied to user attributes rather than personal identities. A ciphertext C, an attribute set A belonging to the user, and given the access policy P defined by the data owner, only if the attribute set meets the policy, decryption will be allowed.

$$Dec(C,A)=\begin{cases} n, & if\ satisfy(A,P)=True \\ \perp, & otherwise \end{cases} \quad (16)$$

where, A denotes the set of attributes held by a user, P indicates the data owner defined the access policy, C is the ciphertext protected by an access structure, $\perp$ represents a result is invalid decryption result. This will comply with the data owner's access policy, only users with the attribute retrieve plain text that ensures.

c) Integration of HE and ABE

The hybrid encryption suggested is based on HE and ABE to allow secure computation with fine-grained access control. Data are encrypted to HE first, and then they are subjected to privacy-preserving computation without decryption. The calculated value is again encrypted under ABE based on the data-owner policy such that only people with valid attributes can decrypt the value. Let the Dbe dataset be encrypted as C_{HE}; computation is carried out to obtain $C_{HE,ABE}$. Decryption will only be allowed when user attributes meet the access policy.

$$Dec(C_{HE,ABE},A)=f(m),\ if\ Satisfy(A,Q)=True \quad (17)$$

D. Cross-Cloud Data Interoperability and Trust Management

1) Trust-aware blockchain-assisted proxy re-encryption

Traditional PRE enables a proxy to convert ciphertext encrypted under one user's key into another ciphertext under a different user's key, without having to learn it. Although it facilitates secure data sharing across clouds, the reliability of proxies under FID does not inherently consider dynamic key revocation or multi-cloud interoperability. To overcome these challenges, this research introduces the proposed T-AB-APRE protocol.

a) Trust-aware proxy selection

Every proxy P_u is assigned a trust score $ST_u(t)$ is updated dynamically based on SLA like, historical behavior, compliance, and anomaly detection.

$$ST_u^{t+1}=\alpha\cdot ST_u^t+\beta\cdot f(behaviour, SLAcompliance, anomalies) \quad (18)$$

Only proxies with $ST_u^{(t)}\geq\theta$ are eligible for the re-encryption tasks ensure that only the trustworthy entities handle sensitive ciphertexts.

b) Blockchain-based auditing

Every re-encryption request, key updates, and trust evaluation on the blockchain are immutably recorded:

$$Ledg_{PRE}\leftarrow\{C_{orig},C_{reenc},P_u,ST_u,timestamp\} \quad (19)$$

where, C_{orig} denotes the original ciphertext before re-encryption, C_{reenc} indicates the re-encryption after the proxy transforms it to the recipient's key, P_u represents the identifier of the proxy performing the re-encryption, ST_u is the trust score of the proxy P_u at the time of the re-encryption operation.

The non-repudiation, auditability, and prevention of malicious proxies from tampering with the ciphertext transformation are ensured.

c) Dynamic key management

Keys are utilized for re-encryption (*rek*), which are updated periodically and correlated with proxy trust scores. If a proxy's trust score falls below the threshold, the rights of re-encryption are revoked immediately, and real-time security adoption is implemented.

d) Attribute and policy integration

T-AB-APRE combines the data owner with the ABE policies. The re-encryption ciphertext user attributes are bound to limited re-encryption rights; many clouds offer granular access control. Fig. 4 depicts the Architecture of T-AB-APRE.

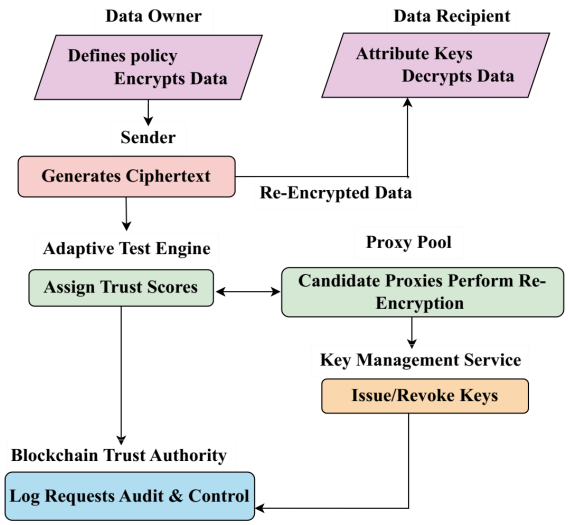

Fig. 4. Architecture of T-AB-APRE (200 users).

2) Continuous trust re-evaluation of blockchain-based key renewal

The User and proxy trust score utilizes the same ATS Eq. (7), which is constantly updated. The user and the proxy trust score are updated dynamically through time. The access policies and cryptographic keys to prevent compromise are updated periodically and recorded on the blockchain. This technique guarantees long-term confidentiality in a federated multi-cloud system, integrity, and policy compliance. The pseudocode of the BFS-ZAT framework is presented in Algorithm 1.

Algorithm 1. BFS-ZAT Framework

Input: User biometric vector (A_u), User key pair (SN_u, PN_u), CPs
Output: Secure cross-cloud access, privacy-preserving query execution, trust-aware data sharing

// Phase 1: User Registration via MCFI
Compute blockchain anchor is evaluated using equation (1)
Generate FID is evaluated using equation (2)
Store FID_u , K_u on blockchain
// Phase 2: System Initialization via BTA
for each CP_e in CPs do
Assign initial trust score is evaluated using equation (3)
Generate SN_e, PN_e // Cloud cryptographic keys
Log registration is evaluated using equation (4)
$BTA_{log}\left(LedgEvent\right)$
end for
$SN_{ABE}, PN_{ABE} \leftarrow$ ABESetup() // ABE setup
$SN_{HE}, PN_{HE} \leftarrow$ HESetup() // HE setup
// Phase 3: Access Control and Authorization
When user *u* requests access to CP_e :
ATS is evaluated using equation (6)
if $T_u\left(t\right) < \theta_u$ then
Access (*u*) ← Denied
return Access (*u*)
end if
$AuthZ\left(u\right) \leftarrow ZKVerify\left(\pi u, FID_u\right)$ //ZKP verification
if $AuthZ\left(u\right)$ = FALSE then
Access (*u*)← Denied
return Access (*u*)
end if
Access (*u*)← Granted
// Phase 4: Privacy-Preserving Query Processing
Symmetric encryption is evaluated using equation (9)
DP evaluated using equation (10)
if ($DataOwner_{Approve}\left(Q', Auth\left(u\right)\right)$ = FALSE
then
Reject query
else
$LedgEvent \rightarrow \left\{Q', Approve = 1, SLA, timestamp\right\}$
$BTA_{log}\left(LedgEvent\right)$
end if
Phase 5: Confidential Data Sharing (HE + ABE)
$C_{He} \leftarrow HE_{Enc}\left(SN_{He}, Data\right)$
$C_{He} \leftarrow HE_{Compute}\left(C_{He}, f\right)$ //
Computation over ciphertext
$C_{ABE} \leftarrow ABE_{Enc}\left(SN_{ABE}, Policy, C_{HE}\right)$
// ABE
$Data_u \leftarrow ABE_{Dec}\left(PN_{ABE}\left(u, Attrs\right), C_{ABE}\right)$
// Authorized user decryption
// Phase 6: Cross-Cloud Data Interoperability via T-AB-APRE
For each proxy P_u do
Update using equation (7)
if $ST_u\left(t+1\right) \geq \theta$ then
$rek \leftarrow PRE_{ReKeyGen}\left(SN_{from}, PN_{to}\right)$
$C_{re} \leftarrow PRE_{ReEnc}\left(rek, C_{ABE}\right)$
$Ledg_{PRE} \leftarrow \left\{C_{orig}, C_{reenc}, P_u, ST_u, timestamp\right\}$
$BTA_{log}\left(LedgEvent\right)$
end if
end for
Phase 7: Continuous Trust and Key Management
Update user and provider trust scores using equation (18)
Periodically renew cryptographic keys and log updates on blockchain
return Success

V. Experimental Results

The experimental results discuss the experimental setup, parameter setting, evaluation measures, comparative analysis and performance analysis of the BFS-ZAT technique. The performance and the comparative analysis of various existing approaches are assessed to evaluate the effectiveness of the techniques.

A. Experimental Setup

The proposed blockchain-enabled multi-cloud federated framework, this evaluates introduced a performance of five Cloud Providers into a heterogeneous environment prototype (AWS, Azure, GCP, Openstack, and a private cloud), and the behavioral patterns and varying trust levels are simulated with 200 users. Using Hyperledger Fabric v2.5 with four peers under the Raft consensus protocol, the Blockchain layer is implemented. In this, the fault-tolerance and the immutable trust management are ensured. Experiments were conducted on servers equipped with Intel Xeon 3.4 GHz processors, Ubuntu 22.04, and 32 GB RAM, while cryptographic operations were supported by Charm-Crypto for ABE, PALISADE for HE, and a PRE library. The weights α, β, γ determine the relative influence of these components and satisfy $\alpha+\beta+\gamma = 1$. The experimental workload (200 users, 5 cloud providers, 10% attack rate) was selected by sensitivity analysis where trust score weights (α = 0.5, β = 0.3, γ = 0.2) were used because past trustworthiness was of the greatest importance, but behavioral consistency and anomaly danger were also taken into account to select the best weights to obtain the highest accuracy in the assessment of the trustworthiness.

B. Parameter Setting

Table I outlines the parameter settings for the BFS-ZAT protocol, where every parameter is carefully selected based on multi-cloud environments simulation practices to guarantee accurate, fair, and reproducible outcomes.

Table II provides the baseline, which are the closest state-of-the-art design paradigms related to BFS-ZAT.

TABLE I. PARAMETER SETTINGS OF THE BFS-ZAT TECHNIQUE (200 USERS, 5 CLOUD PROVIDERS, 1000 MIXED QUERIES)

Parameter	Value
Number of Users	200
Number of Cloud Providers	5
Cryptographic security level	128-bit
Trust Score Weights	$\alpha = 0.5$ (historical), $\beta = 0.3$ (behavior), $\gamma = 0.2$ (anomaly)
Adaptive Trust Threshold (θ)	0.7
Query Workload	1000 mixed queries
Attack Injection Rate	10% of total requests
ZKP proof size	~288 bytes
HE bit length	4096

TABLE II. BASELINE METHODS FOR BFS-ZAT TECHNIQUE

Baseline	Represent	Why Relevant to BFS-ZAT
MCFI	Multi-cloud federated identity	Unauthorized modifications are prevented.
ZKP	zero-knowledge proofs	Access evaluation
BTA	Blockchain trust authority	Parameter manage
ATS	Adaptive trust scoring	Evaluate the user trustworthiness.
HE / ABE	Homomorphic encryption / Attribute-based encryption	Data Leakage prevention / Fine graind access control
T-AB-APRE	Trust-aware blockchain-assisted proxy re-encryption	Cipher text transformation across cloud domains.

C. Evaluation Measures

The proposed BFS-ZAT protocol is widely adopted for evaluation by using six metrics in multi-cloud environments, such as Authentication Latency (*AL*), Transaction Throughput (*TT*), Trust Evaluation Accuracy (*TEA*), Attack Resilience Rate (*ARR*), Resource Utilization (*RU*), and Service Availability (*SA*). Each metric is associated with its mathematical formulation, and the variable definitions are provided below:

Authentication latency: Authentication latency indicates the average time needed to finish the authentication process between the system and the entity (user, device, or service). This measures the response for the authentication protocol.

$$AL = \frac{\sum_{e=1}^{N}\left(t_{verify,e} - t_{request,e}\right)}{N} \tag{20}$$

where, $t_{verify,e}$ denotes the request initiation time, $t_{request,e}$ represents the verification completion time, and N indicates the total number of authentication requests.

Transaction throughput: Transaction throughput defines the total number of transactions successfully processed by the system. This reflects the capacity and scalability of the system.

$$TT = \frac{T_{succ}}{T_{total}} \tag{21}$$

where, T_{succ} represents the total transaction that is successfully executed, and T_{total} indicates the total execution time.

Trust evaluation accuracy: The trust evaluation accuracy represents how the trusted or untrusted accurately classifies the system and the entities that are compared with the basic truth. This trust mechanism measures correctness.

$$TEA = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \tag{22}$$

where, *TP* represents true positives, *TN* indicates the true negatives, *FP* represents the false positives, and *FN* denotes false negatives.

Attack resilience rate: Attack resilience rate is the different types of attacks in the ability of the system resist and mitigate (sybil, replay, insider). A large value denotes robust resilience.

$$ARR = \frac{N_{blocked}}{N_{attack}} \times 100\% \tag{23}$$

where, $N_{blocked}$ denotes the number of attacks successfully prevented N_{attack} indicates the total number of attempted attacks.

Resource utilization: During execution, the rate of system resources consumed is defined as resource utilization (memory, CPU, and bandwidth). This is a workload handling framework that reflects efficiency.

$$RU = \frac{R_{used}}{R_{total}} \times 100\% \tag{24}$$

where, R_{used} denotes the resources consumed the amount, R_{total} is the total available resource.

Service availability: Service availability represents the system's ability to remain operational and the percentage of time that is accessible to authorized users. Large availability in multi-cloud environments ensures reliability.

$$SA = \frac{T_{uptime}}{T_{uptime} + T_{downtime}} \times 100\% \tag{25}$$

where, T_{uptime} indicates the total operational time, $T_{downtime}$ denotes the total system downtime.

D. Performance Analysis

The section analyzes the performance of the BFS-ZAT method, highlighting its effectiveness. Table III presents the system performance with varying numbers of users, considering CPU usage, memory consumption and bandwidth utilization. These parameters determine the system's ability to handle concurrent users while ensuring secure encryption and decryption operations. The analysis shows that as the number of users increases, resource consumption grows moderately, reflecting the scalability and efficiency of BFS-ZAT. Efficient utilization of resources optimizes operational costs and also enhances performance, ensures flexibility, and supports large-scale deployment across federated multi-cloud environments.

The experimental evaluation results by the proposed BFS-ZAT framework are authentication latency (180 ms), transaction throughput (220 tps), and trust evaluation accuracy (94%).

TABLE III. SYSTEM RESOURCE UTILIZATION OF BFS-ZAT WITH VARYING USERS (200 USERS)

Number of Users	CPU (%)	Memory (MB)	Bandwidth (Mbps)
50	45	1400	110
100	50	1600	120
150	55	1800	130
200	60	2000	150

E. Comparative Analysis

To comprehensively assess the performance of the BFS-ZAT method, it is compared with five existing routing protocols: MCDCA [22], ABE-DSC [14], Twofish256-SKC [12], EffMCS [11], and MET-TC+RF [20].

Fig. 5 presents the authentication latency performance of the BFS-ZAT method against other approaches with a number of users ranging from 10 to 200. This technique delivers the lowest authentication latency with 180 ms at 200 users. In comparison, the existing method shows the highest authentication latency in 200 users of MCDCA (205 ms), ABE-DSC (192 ms), Twofish256-SKC (201 ms), EffMCS (185 ms), and MET-TC+RF (196 ms). The result indicates the low latency and a quick authentication process, which signifies the performance that leads to user frustration and potential security vulnerabilities.

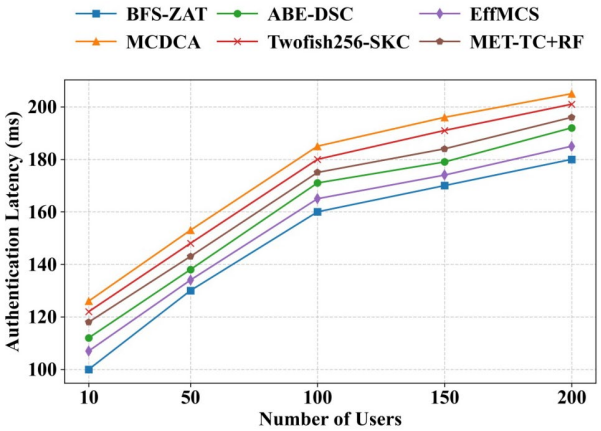

Fig. 5. Authentication latency comparison between the proposed and the existing techniques.

Fig. 6 depicts the transaction throughput of the BFS-ZAT technique, which shows a different number of transactions ranging from 200 to 1,000 compared to existing methods. The BFS-ZAT techniques deliver the highest performance, with users achieving 220 tps at 1,000 transactions. In contrast, the other methods exhibit lower values of MCDCA (190 tps), ABE-DSC (208 tps), Twofish256-SKC (195 tps), EffMCS (214 tps) and MET-TC+RF (201 tps) at 1000 transactions. This method demonstrates a system's capacity, efficiency, and ability to handle user demand.

Fig. 7 depicts the trust evaluation accuracy performance of the BFS-ZAT method varying the attack injection rate ranging from 0 to 20, against other techniques. This approach achieves the highest trust evaluation accuracy, reaching 94% at 20 attack injection rates. In comparison, the other approaches show the lowest trust evaluation accuracy values: MCDCA (71%), ABE-DSC (85%), Twofish256-SKC (76%), EffMCS (90%) and MET-TC+RF (81%) at 20 attack injection rates. This model considers multiple trust metrics, including communication trust, data trust, and energy trust, and incorporates penalty factors and adjustment functions to calculate direct trust.

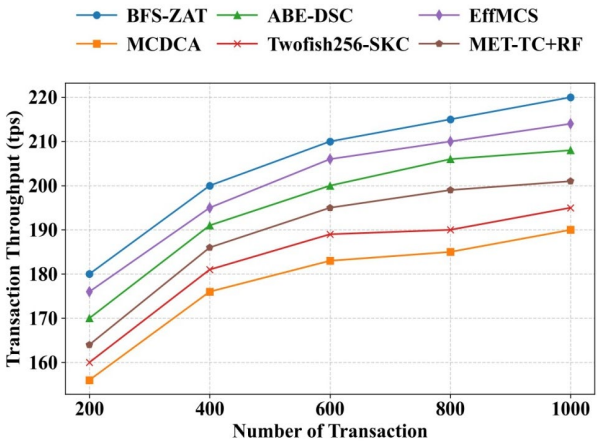

Fig. 6. Transaction throughput comparison between the proposed and the existing techniques.

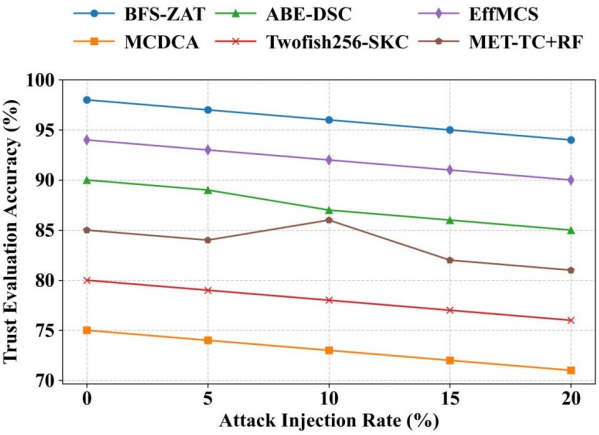

Fig. 7. Trust evaluation accuracy comparison between the proposed and the existing techniques.

Fig. 8 illustrates the attack resilience rate of the BFS-ZAT technique under three critical attack scenarios: impersonation, replay, and malicious CP attacks. This method achieves the highest attack resilience rate at 95%, 94%, and 96% respectively, in three attack types. In contrast, the other methods exhibit lower values of MCDCA (75%, 67%, and 78%), ABE-DSC (87%, 82%, and 89%), Twofish256-SKC (80%, 74%, and 81%), EffMCS (91%, 88%, and 92%) and MET-TC+RF (84%, 77%, and 85%). The framework provides a more robust and reliable defense compared to conventional approaches in multi-cloud environments.

Fig. 9 depicts the service availability of the BFS-ZAT technique against other approaches. The BFS-ZAT technique achieves the service availability of 95.32%. In contrast, the other methods exhibit lower values of MCDCA (71.46%), ABE-DSC (86.44%), Twofish256-SKC (76.24%), EffMCS (90.26%) and MET-TC+RF

(81.45%). This method ensures the users access the service when needed, with common causes of unavailability including planned maintenance and unexpected outages.

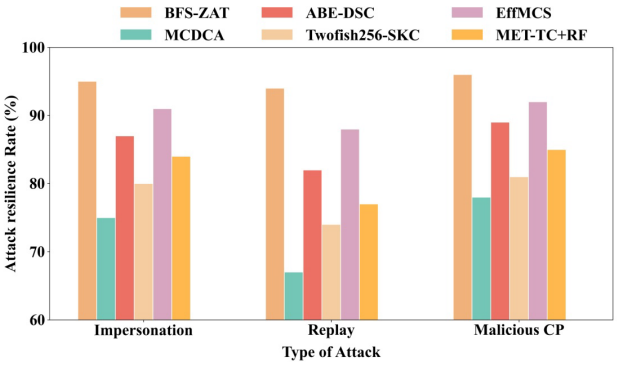

Fig. 8. Attack resilience rate comparison between the proposed and the existing techniques.

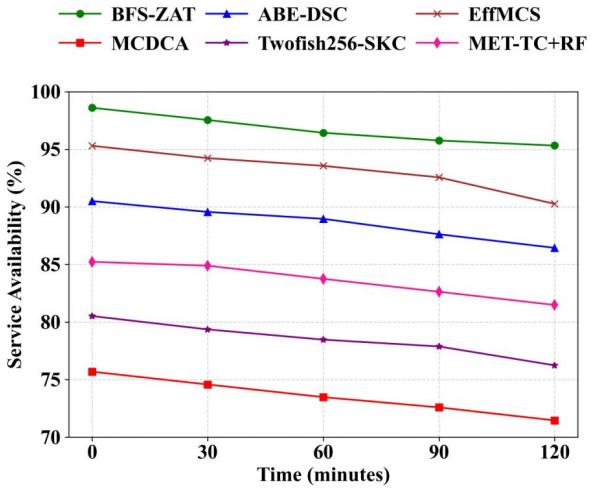

Fig. 9. Service availability comparison between the proposed and the existing techniques.

VI. Conclusions and Future Work

This proposed system suggested the BFS-ZAT to offer highly secure, transparent, and privacy-preserving access control on federated multi-clouds. This framework proposal integrates blockchain-safe federated identity, adaptive trust scoring, zero-knowledge authentication, hybrid HE-ABE encryption, and trust-aware proxy re-encryption to offer policy-compliant and auditable cross-cloud data sharing. The BFS-ZAT framework addresses the key drawbacks of the current multi-cloud security solutions, which comprise the lack of dynamically managed trust, credential leakage, and lack of interoperability, which cannot be overcome by relying solely on decentralized trust management. The results of experiments demonstrate that the BFS-ZAT framework can be significantly better than the existing ones, with 180 ms authentication delay, 220 tps transaction rate, and 94% trust evaluation accuracy, proving its effective and applicable use in secure multi-cloud operations.

The future work will focus on the large-scale and real-time implementation of systems on more widespread multi-cloud environments, the integration of AI-based anomaly detection to further enhance adaptive trust and resilience to attacks, and the system optimization via lightweight cryptography to reduce the overhead in resource-constrained environments.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Prabakaran K: Conceptualization, methodology, formal analysis, resources, data analysis, visualization, writing—original draft. Sivakumar B: Conceptualization, data analysis, review & editing, supervision, investigation, verification and validation. All authors had approved the final version.

References

[1] A. Alli, N. Raj, S. Sahoo *et al*., "Load balancing and intrusion detection techniques for enhanced security in multi-cloud environments," in *Proc. 2025 International Conference on Networks and Cryptology (NETCRYPT)*, 2025, pp. 1770–1775.

[2] C. Diningrat, and B. Rahardjo, "Security issues in multi-cloud: A systematic literature review," *IEEE Access*, vol. 13, pp. 70006–70017, 2025.

[3] M. Madanan, P. Patel, P. Agrawal *et al*., "Security challenges in multi-cloud environments: Solutions and best practices," in *Proc. 2024 7th International Conference on Contemporary Computing and Informatics (IC3I)*, 2024, vol. 7, pp. 1608–1614.

[4] S. Ali, D. B. Talpur, A. Abro *et al*., "Security and privacy in multi-cloud and hybrid cloud environments: Challenges, strategies, and future directions," *Computers & Security*, vol. 157, 104599, 2025.

[5] A. Punia, P. Gulia, N. S. Gill *et al*., "A systematic review on blockchain-based access control systems in cloud environment," *Journal of Cloud Computing*, vol. 13, no. 1, 146, 2024.

[6] S. R. Gopireddy and A. D. Engineer, "Securing multi-cloud environments with azure: Challenges and solutions," *IJCEM*, vol. 6, no. 12, 2021.

[7] M. S. Saleh, "Blockchain for secure and decentralized artificial intelligence in cybersecurity: A comprehensive review," *Blockchain: Research and Applications*, vol. 5, no. 3, 100193, 2024.

[8] M. Waseem, A. Ahmad, P. Liang *et al*., "Containerization in multi-cloud environment: Roles, strategies, challenges, and solutions for effective implementation," *Journal of Systems and Software*, vol. 230, 112558, 2025.

[9] V. Baladari, "Enhancing performance and security in multi-cloud and hybrid-cloud environments," *International Journal of Core Engineering and Management*, vol. 7, no. 11, pp.53–265, 2024.

[10] J. Brown, and S. Adams. (2025). The power of centralized security: Best practices for managing security policies across multi-cloud environments. [Online]. Available: https://www.researchgate.net/profile/Bouchra-Hocine/publication/393007623_The_Power_of_Centralized_Security_Best_Practices_for_Managing_Security_Policies_Across_Multi-Cloud_Environments/links/685c216607d6d53e82ee75d8/The-Power-of-Centralized-Security-Best-Practices-for-Managing-Security-Policies-Across-Multi-Cloud-Environments.pdf

[11] N. Karnik, A. Kumar, P. Mahajan *et al*., "An efficient technique for securing a multi-cloud storage environment," *International Journal of System Assurance Engineering and Management*, pp.1–12, 2025.

[12] S. Bansal, M. S. Nidhya, K. Chheda *et al*., "An efficient strategy for ensuring multi-cloud information security," *International*

Journal of System Assurance Engineering and Management, pp.1–12, 2025.
[13] K. O. Chauke, N. Makondo, and T. Muchenje, "Enhancing network security in multi-cloud environments through adaptive threat detection," *Int. J. Cloud Secur.*, vol. 5, no. 1, pp. 66–82, 2025.
[14] Yang, P. Li, K. Xiao *et al*., "An efficient attribute-based encryption scheme with data security classification in the multi-cloud environment," *Electronics*, vol. 12, no. 20, 4237, 2023.
[15] A. Jin, Y. Xu, W. Qin *et al*., "A blockchain-based auditable deduplication scheme for multi-cloud storage," *Peer-to-Peer Networking and Applications*, vol. 17, no. 5, pp.2870–2883, 2024.
[16] N. Bharot, N. Mehta, J. G. Breslin, and P. Verma, "Cloudlock: Secure data sharing using a hybrid cryptosystem in multi-cloud data storage," *Cluster Computing*, vol. 28, no. 7, 464, 2025.
[17] N. Witanto, B. Stanley, and S. G. Lee, "Distributed data integrity verification scheme in multi-cloud environment," *Sensors*, vol. 23, no. 3, 1623. 2023.
[18] Noh, S. Ji, Y. Go *et al*., "Resilient and fast block transmission system for scalable hyperledger fabric blockchain in multi-cloud environments," *IEEE Transactions on Network and Service Management*, vol. 21, no. 5, pp. 5118–5134, 2024.
[19] S. A. Jebakumari, S. Mahajan, H. Raichura *et al*., "Innovative model for security of multi-cloud platform: data integrity perspective," *International Journal of System Assurance Engineering and Management*, pp. 1–8, 2024.
[20] W. Huang, Y. Chen, D. Jing *et al*., "A multicloud collaborative data security sharing scheme with blockchain indexing in industrial internet environments," *IEEE Internet of Things Journal*, vol. 11, no. 16, pp. 27532–27544. 2024.
[21] Zeydan, J. Baranda, J. Mangues-Bafalluy *et al*., "A trustworthy framework for multi-cloud service management: Self-sovereign identity integration," *IEEE Transactions on Network Science and Engineering*, vol. 11, no. 3, pp. 3135–3147, 2023.
[22] S. Bhatt, A. Shivarudra, S. Kavuri *et al*., "Building scalable and secure data ecosystems for multi-cloud architectures," *Letters in High Energy Physics*, 2024.