International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(9s) 2026 pp. 673-686
https://doi.org/10.70917/ijcisim-2026-3488
Article

SecureVault: Architecting a Privacy-First Cryptographic Layer for Multi-Cloud Data Governance

J Sirisha Devi [1], P V Bhaskar Reddy[2], M Purushotham Reddy [3], A Jayanthi[4]
[1, 3]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Bowrampet, Hyderabad-500043, Telangana, India.
[2]School of Computer Science and Engineering, Reva University, Rukmini Knowledge Park, Yelahanka, Bengaluru - 500064
[4]Department of Computer Science and Engineering, Aditya University, Surampalem, Andhra Pradesh
[1]siri.cse21@gmail.com,[2]bhskr.dwh@gmail.com,[3]purushotham.mps@gmail.com, [4]jayanthi.a33@gmail.com

Abstract: This paper presents a Privacy-Preserving Cryptographic Framework (PPCF) for fine-grained access control in heterogeneous multi-cloud storage environments. PPCF integrates three cryptographic primitives — Ciphertext-Policy Attribute-Based Encryption (CP-ABE), identity-based Proxy Re-Encryption (PRE), and threshold-based Distributed Key Generation (DKG) via Pedersen verifiable secret sharing — to eliminate centralised key escrow vulnerabilities. Access policies are embedded directly into ciphertexts as monotone Boolean formulas, enabling data-owner-defined control independent of online authority infrastructure. A hybrid AES-GCM-256/CP-ABE design confines ABE overhead to symmetric key material, while Reed-Solomon erasure coding ensures fault tolerance and data fragmentation across providers. Formal security reductions establish IND-CPA security under the DBDH assumption, with collusion and key recovery resistance. Empirical evaluation demonstrates a 37% reduction in computational overhead, key generation throughput of 682–847 keys/second, and sub-linear access latency scaling from 187 ms to 427 ms, confirming enterprise-grade deployability.
Keywords: Cryptographic Data Privacy, Attribute-Based Encryption, Proxy Re-Encryption, Distributed Key Management, Multi-Cloud Access Control

1. Introduction

Multi-cloud environments introduce compounded security challenges that single-cloud solutions cannot adequately address, particularly fine-grained access control across heterogeneous platforms with differing trust models, regulatory frameworks, and the need to guard against both external adversaries and potentially compromised cloud providers themselves. Existing security paradigms — client-side encryption with centralised key servers and provider-managed encryption services — are fundamentally insufficient, as the former creates single points of failure and key escrow vulnerabilities while the latter demands implicit trust in cloud vendors, contradicting zero-trust principles and lacking the expressiveness required for attribute-based, context-aware access control policies.

The limitations discussed in this paper were solved by providing a detailed cryptographic privacy-sensitive data access control framework for multi-cloud storage systems. The framework brings together three basic cryptographic primitives: policy-based ciphertext attribute-based encryption to enable fine-grained access control [1, 2], proxy re-encryption enabling secure data delegation without explicit key sharing [4], and distributed key management protocols that are decentralised and do not have centralised trust assumptions [6, 13]. This integration allows organisations the capability to establish advanced, attribute-based access policies while maintaining data privacy across cloud boundaries.

The main achievements of this work are the following. First, a new hybrid encryption scheme is analysed and designed to be both computationally efficient and expressive in policy, enabling principled enterprise-scale deployment. Second, a geographically dispersed key generation and management protocol is set up to offer robustness in the event of node compromise and without involving any trusted central authority. Third, dynamic policy update and user revocation schemes are presented that do not require data re-encryption, significantly decreasing the operational cost of policy management. Fourth, an extensive security analysis is performed that illustrates resistance against collusion attacks, chosen-plaintext attacks, and key recovery attempts. Finally, the framework is assessed using large-scale experiments that measure encryption overhead, access latency, and scalability under realistic operating conditions.

2. Literature Review

The security of cloud computing is a research problem that is taken seriously at various levels, particularly the protection of data confidentiality and integrity, access control, and privacy protection [14]. Initial research on cloud security focused on encryption at rest and secure transmission protocols. With the advent of multi-cloud environments, more advanced solutions were required to address distributed trust models and cross-cloud data management. Attribute-Based Encryption is a breakthrough paradigm as opposed to traditional public key encryption, by means of direct integration of access control functionality into the cryptographic scheme [7]. Initial work introduced fuzzy identity-based encryption, on which ABE was built. This work has developed into two main types: Key-Policy Attribute-Based Encryption, where access policies are represented in the private keys, and Ciphertext-Policy Attribute-Based Encryption, in which policies are encoded as ciphertexts. CP-ABE data owners are given the ability by CP-ABE schemes [2, 3] to program the requirements of decryption in encrypted data, making this paradigm well-suited to organisational access control in which policy authority lies with data owners but not with key authorities. Under the implementation of cloud-based ABE, it has been studied in research for optimisations to address policy expressiveness and computational overhead limitations. Outsourced ABE schemes in which computationally expensive decryption operations are outsourced to cloud providers and security is ensured via verifiable computation methods, thus reducing the computational work of a client. Multi-authority ABE systems [5, 10] share the generation of attribute keys between different independent authorities, removing single points of failure and facilitating attribute management across organisational boundaries. Nonetheless, such systems pose challenges to policy coordination and increase risks of collusion among the authorities, which have to be designed carefully.

An effective and secure delegation tool used to protect data is proxy re-encryption, which enables a semi-trusted proxy intermediary to transform ciphertexts encrypted with one set of public keys into ciphertexts that may be decrypted using another set of public keys. This crypto-primitive can be applied in multi-cloud setups where data proprietors desire to authorise third parties to access particular data with restricted access without revealing secret keys or requiring local re-encryption. Extensions include conditional proxy re-encryption. Through attribute-based conditions introduced in proxy re-encryption, additional statements about delegation policies can be made, aligning with organisational requirements.

Distributed key management schemes eliminate the fundamental vulnerability of centralisation by using distributed key generation and storage distributed amongst different numbers of nodes via threshold cryptography and secret sharing [6, 13]. With threshold schemes, it is guaranteed that no single party is in full possession of key material, so this eliminates the risk of key escrow when providing resistance to node compromise. Research on this aspect has produced a variety of threshold protocols, including Shamir secret sharing protocols and Byzantine fault-tolerant protocols, which achieve security despite the possible corruption of a limited number of nodes and malicious behaviour [8].

A critical emerging direction in attribute-based encryption research concerns post-quantum security. Classical CP-ABE constructions, including the scheme employed in this work, rely on the hardness of the Decisional Bilinear Diffie-Hellman (DBDH) assumption over elliptic curve groups, which is vulnerable to Shor's algorithm on sufficiently powerful quantum computers [1]. Recent surveys have catalogued lattice-based CP-ABE schemes founded on Learning With Errors (LWE) and Ring-LWE hardness assumptions, which are believed to be quantum-resistant and are currently under evaluation by NIST for post-quantum standardisation. Lattice-based ABE schemes such as those based on the BGG+ construction achieve selective security under LWE but currently incur significantly higher ciphertext sizes and key generation overhead — often an order of magnitude larger than pairing-based schemes — making direct enterprise deployment challenging at present. Multi-authority lattice ABE schemes remain an open and active research problem, with no construction yet achieving the combined properties of full collusion resistance, distributed trust, and practical efficiency simultaneously. The framework proposed in this paper is architected with a modular cryptographic layer, such that the CP-ABE component can be substituted with a post-quantum ABE scheme as the field matures, without requiring redesign of the proxy re-encryption or distributed key management layers. Investigating this substitution and its performance implications represents a concrete direction for future work.

3. Current Practice and Comparative Frameworks

The current multi-cloud security practice is on a continuum of strategies from key management services that are provider-controlled to complete client-side encryption platforms [15]. Major cloud providers provide key management services that enable the customer to utilise either keys controlled by the provider or keys controlled by the customer stored on separate hardware security modules. Although convenient operation-wise, these solutions demand the possibility of having to trust the cloud provider with access to data or key custody that is inherently incompatible with the zero-trust competence of work on sensitive materials. Cross-cloud encryption systems that run through and across diverse providers are normally based on plain and simple encryption algorithms that are unable to impose attribute-driven, fine-grained access control policies.

Current scholarly models have come up with customised remedies to particular multicloud security challenges. The Archi-cloud (designed using the SecCloud framework) is a hybrid private and public cloud

architecture to assist in storing and processing data in a secure manner, encrypting the data prior to outsourcing and keeping encrypted keys in a trusted private cloud. While effective for confidentiality, this model once again makes centralised trust assumptions and develops a single point of failure. FADE framework deals with assured deletion in the cloud storage using key deletion by time as well as policy-controlled deletion; however, it does not offer attribute-based access control or cross-cloud capabilities. CP-ABE-based cloud storage systems [9] support expressive access policies with proxy re-encryption-based revocation, but assume there is one credentialing authority, which is essentially constraining organisational scalability and cross-domain applicability.

ABE schemes don't consider multi-authority, but multi-authority ABE schemes [5,10] distribute attribute control among several independent authorities that deal with single-authority restrictions in favour of more coordination complexity and the establishment of collusion between authorities. Blockchain-based access control mechanisms can take advantage of properties of decentralisation and immutability to provide transparent and auditable access management using smart contracts with no centralised authorities. But integration with blockchain comes at a high cost in terms of latency and throughput, and data access can be prohibitive in high-frequency data access scenarios due to limitations and, in some cases, very high computational costs. Also, putting access policies in the public blockchains heightens confidentiality and policymaking issues.

The commercial cloud access security brokers are used to concentrate policy enforcement and threat identification of various cloud services at a distributed location. These solutions offer good visibility and control, but add another latency bottleneck in their operations and possible single points of failure. The requirement to route data via CASB infrastructure is incompatible with direct cloud access patterns and can breach data sovereignty conditions in the case where the data broker architecture is managed in other jurisdictions.

All the constraints inherent in current strategies can be grouped into five areas. To begin with, the existing architectures tend to cover single security features separately instead of providing completely integrated solutions that involve encryption, access control, and key management. Second, optimisations to performance usually compromise security assurances or policy expressiveness that cannot be made for sensitive workloads. Third, the cost of policy and user revocation is still a computationally expensive operation that needs re-encryption of the data or a complicated system of key updating [11]. Fourth, systems of distributed key management either maintain central assumptions of trust or may introduce heavy overheads of coordination that may reduce scalability [13]. Fifth, cross-cloud data sharing methods are often such that data owners must be constantly present online or sustain key management infrastructure, developing operational burdens incompatible with enterprise models of deployment. All these restrictions are motivating the establishment of a universal scheme which combines a variety of cryptography primitives with computational efficiency and expressive and dynamic revocable access control, without the use of centralised trust, and supporting sharing of the cross-cloud without any continuous data owner interaction.

4. Methodology

The proposed framework is founded on a multi-layered architecture that systematically integrates cryptographic primitives, distributed protocols, and access control mechanisms. This section details the system architecture, protocol descriptions, implementation approach, and evaluation methodology.

4.1 System Architecture

The framework architecture assumes five major components, namely: data owners, data users, cloud storage, key management, and attribute authorities. Data owners are data centres that originate and hold sensitive information; they devise access controls, encrypt information, and store encrypted documents with cloud storage companies. Data users are entities that demand the right to access secure data; they bear attribute credentials given out by attribute authorities and use credentials to decrypt data that has an embedded access policy that their attribute set satisfies. Attribute authorities are locations of trust that verify and register user identities and give them individual key parts. Cloud storage providers store ciphertext key bundles and encrypted data; in response to a retrieval request, they undergo policy updates via an operation that is a proxy re-encryption. However, their cryptographic ability has been used to access plaintext data. The distribution of key management nodes is performed as systemic cryptographic generative and sustaining components of infrastructure with parameters that have no separate node material of the full master key.

The model within which the system is operated is a semi-honest adversarial model whereby cloud providers and an effective number of some of the key management nodes can seek to know information regarding some stored data, yet comply with implementation protocols. This threat model realistically is an indication of situations when infrastructure objects can be hacked or run by possibly inquisitive individuals. The model presupposes that there are authenticated secure linkages between elements in the critical distribution of keys and negotiation of policy operations, which can be implemented with standard TLS protocols on certificate-based authentication.

4.1.1 Formal Adversarial Model

Definition 1 (Adversary Classes). The framework considers three classes of probabilistic polynomial-time (PPT) adversaries:

- **Type-I Adversary A_1(External Attacker):** A computationally bounded external party with no access to system credentials. A_1may observe ciphertexts, public parameters, and network traffic, and may mount chosen-plaintext attacks (CPA). Security against $A_1$is formalised via the standard IND-CPA game.
- **Type-II Adversary A_2(Honest-but-Curious Cloud Provider):** A coalition of up to all $n$cloud storage providers who follow the protocol specification but attempt to infer plaintext data or access policy structure from stored ciphertexts and transformation keys. $A_2$is modelled as semi-honest and cannot collude with key management nodes.
- **Type-III Adversary A_3(Colluding Users / Compromised Nodes):** A coalition of up to $t-1$corrupted key management nodes and an unbounded number of colluding data users, each possessing valid but individually insufficient attribute credentials. A_3attempts to reconstruct master key material or decrypt ciphertexts for which no single colluding user holds a satisfying attribute set.

Definition 2 (Security Goals). The framework achieves the following security properties:

Table A1: Security Properties and Formal Guarantees of the SecureVault Framework

Property	Adversary	Formal Guarantee
IND-CPA Security	A_1	Reduction to DBDH assumption
Ciphertext Privacy	A_2	Semantic security under semi-honest model
Collusion Resistance	A_3	Information-theoretic via LSSS
Key Escrow Elimination	A_3	Threshold DKG with t-of-n shares
Forward Secrecy	A_1, A_3	Re-keying on revocation

Assumption 1 (DBDH). Let Gbe a bilinear group of prime order $p$with generator $g$and pairing $e: G \times G \to G_T$. The Decisional Bilinear Diffie-Hellman assumption states that no PPT adversary can distinguish the tuple $(g, g^a, g^b, g^c, e(g,g)^{abc})$from $(g, g^a, g^b, g^c, e(g,g)^z)$with a non-negligible advantage, where $a, b, c, z \leftarrow^R Z_p$.

Assumption 2 (Threshold Secrecy). Under the Pedersen VSS construction, any coalition of fewer than tKey management nodes obtain zero information about the master secret components $\alpha_j$and $\beta_j$in the information-theoretic sense, owing to the t-degree polynomial construction underlying the secret sharing scheme.

4.2 Data Encryption and Storage Protocol

Access policies are defined as Boolean formulas on the attributes of the users by the owners of the data they operate on. They can consist of conjunctions, disjunctions, and threshold access structures, facilitating rich expression of access requirements within the organisation. For example, there could be a policy that demands decryption be preserved by a user who has the qualities of a senior researcher with cryptography specialisation, or a project manager who is also a data steward. These policies can be presented in the form of access trees, whereby leaf nodes are associated with individual attributes and internal nodes denote logical operators along with their threshold parameters.

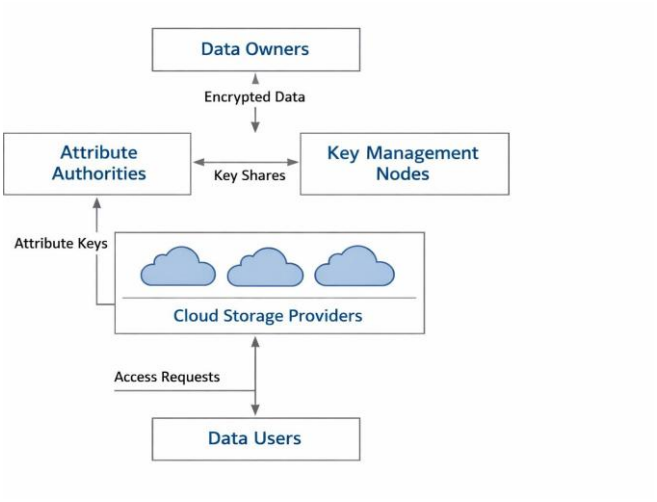

Figure 1: System Architecture Diagram

The encryption algorithm takes three steps. Originally, there is data generated by the owner, a cryptographically random symmetric key and the plaintext data is coded by AES-Galois/Counter Mode (256), which includes approved integrity and secrecy preventive measures against manipulation. Second, CP-ABE is used to encrypt the symmetric key within the access policy tree specified, generating a ciphertext key bundle giving an encoding of the authenticated access requirements. This hybrid technique provides the use of the computational security of symmetric encryption on large volumes of data, and makes use of the sophistication and security of ABE to protect keys and issue access control. Third, the encrypted block of data, the ciphertext key bundle and metadata of retrieval are made available to cloud storage providers. This three-step design will make sure that ABE overhead has nothing to do with the volume of data, in which the fixed length of the symmetric key is only used in the ABE operations.

Information distribution of data among multiple clouds uses an erasure-coding type of dispersal algorithm. The encrypted data is broken into n fragments, wherein k is any number. Any set of fewer than fragments can be reconstructed as a complete data set. The fragments of k give no information about the original data. This approach simultaneously gives fault tolerance to cloud provider failures, eliminates single-provider data exposure, and allows parallel upload and download, which may improve bandwidth throughput over deployments that are bandwidth-constrained.

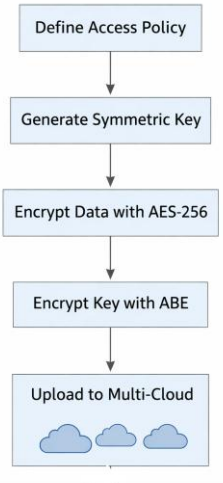

Figure 2: Encryption/Decryption Protocol Flowchart

4.3 Distributed Key Generation Protocol

The key generation protocol, which is distributed, eradicates the centralised trust. Materials for system master keys are created in coordination with several key management nodes. The nodes have no master key on the

individual node. The protocol employs threshold secret sharing, which requires at least t out of n key management nodes (the reconstruction threshold t is set by the deploying organisation; any t−1 or fewer nodes are insufficient) for any operation that uses master key material on which the threshold is below. Parameter t can be set to trade off between the needs for security and the needs for availability. Once the system is started, the parties involved in this process carry out a verifiable secret sharing protocol. Each node participates in the generation of independent randomness, and contributions add incomplete contributions to the system parameters. Multiple rounds of communication enable nodes to mutually arrive at final system public parameters, and each node's individual secret share of executing verification procedures that permit the detection of malicious nodes trying to skew key generation or tap into knowledge of honest nodes' shares while bypassing information security with a key known to the originator. In the case of the generation of attribute keys, attribute authorities communicate with key management nodes to generate attribute half keys. Upon a request for the keys of attributes by a user, the authority authenticates the credentials of the user to the attribute requested, and it initiates a request to the key management network. Each key management node calculates a partial attribute key element utilising its secret share with that of the user attribute identity. Independent provision of these partial components to the user is made available through secure channels, and the user puts his pieces together into an operating complete attribute private key. This is done to make sure that no single key management node is endowed with enough information to create complete attribute keys by unilateral approach or by impersonating legitimate users.

4.4 Access Control and Decryption Protocol

The ACDP is used to grant access to authorised users and prevent unauthorised access to the data being sent. The CP-ABE scheme does not need to interact with any online authority in the process of decryption, and it enforces the cryptographic access control. Users who have the correct attribute credentials, independently, attempt to have the encrypted message decrypted by making an evaluation of whether their set of attributes meets their policy in access encoded in the ciphertext. This offline access control, which is called property, is necessary to have deployability and practicality in ecosystems where authority infrastructure is not always available. The decryption routing algorithm is based on following the access tree starting with leaf nodes and ending on the root node, and bilinear pairing operations. To each leaf node, which represents an attribute in the possession of the user, the protocol calculates a node value with the user's corresponding attribute private key element. The value of child nodes is added together into internal nodes based on their logical operator and logical threshold parameter, and interpolating their logical threshold parameter with Lagrange multipliers. This computation will be successful in recovering the derived symmetric key, which the user can thus apply in decrypting the underlying ciphertext.

The framework can be used to outsource the process of decryption to lightweight clients by relying on a transformation mechanism. The user produces a secret blinding factor, generates an obfuscated form of their own attribute private key, which is shareable using the cloud, and outsources the computationally intensive pairing functions to the cloud provider. The cloud calculates an intermediate decryption value of informationally equivalent value to a blind version of the symmetric key. The user then applies a lightweight local computation. The blinding factor can be used to recover the actual symmetric key. This algorithm cuts down on the computation on the client-side decryption by about 80% with complete security, because the cloud provider will be aware of nothing about the symmetric key or the underlying plaintext of the intermediate result.

4.5 Dynamic Policy Update and User Revocation

The concept behind dynamic access policy updates is to enable the data owner to adjust access conditions of previously encrypted data without computationally expensive data re-encryption. The system uses the procedure of proxy re-encryption, where the transformation key is calculated based on the information of the data owner's secret master and on the mutual connection between the previous policy and the new policy. It allows cloud providers to transform old ciphertext key bundles encrypted under the old policy to the new policy encrypted without ever being able to retrieve underlying symmetric key data or plaintext data.

On changing an access policy, a data owner calculates a policy transformation key based on their master secret and policy change specification. This transformation key is properly sent to the corresponding cloud

companies that hold the impacted data. Cloud providers have the transformation applied to a ciphertext key bundle of stored ciphertexts, and change it to reflect the new policy. Users whose sets of attributes meet the new policy are able to decode the encrypted ciphertexts, but end-users who no longer meet the new policy requirements automatically go into a denied state without explicit revocation of the policy being performed.

An active attribute key is also revoked by using a combination of immediate attribute key invalidation and progressive background re-encryption. When a user has been revoked, attribute authorities re-issue all other, non-revoked users with updated attribute keys for the affected attribute, and relay them via safe channels. All the attribute keys that the revoked user possessed previously become useless in decrypting authentic data encrypted after the revocation event. In the case of already encrypted data, a back-end re-encryption process gradually encodes encrypted ciphertext key bundle transforms in order to impose revocation, with the frequency of re-encryption settable depending on the sensitivity of the data to be impacted and possible available computational resources.

4.6 Implementation Details

Implementation of the framework in Python is done through the Charm cryptographic library that provides the Python implementation of this framework and delivers high-performance deployments of bilinear pairing operations on regular elliptic curve groups. It uses type-A elliptic curve pairings, which are as secure as symmetric encryption on the 128-bit level and provides efficient calculations of the bilinear pairing processes demanded by the ABE and proxy re-encryption schemes. The Waters CP-ABE scheme is implemented with an optimised algorithm for calculating access trees. Unsatisfied subtrees expand early, decreasing the amount of the pairing operations needed in successful decryption cases. Cloud storage adoption is done under the form of a provider-agnostic abstraction layer, which now works with Amazon S3, Microsoft Azure Blob Storage, and Google Cloud Storage. The abstraction layer invisibly clears the differences in APIs by providers so that multi-cloud can be operated without application-level understanding of each individual provider's characteristics. Fragmentation and distribution of data use Reed-Solomon erasure code at adjustable redundancy rates that enable organisations to make suitable trade-offs between storage overhead and fault tolerance, depending on their operational requirements. Key nodes of management are installed as autonomous distributed services that communicate using mutually authenticated TLS connections. The Pedersen verifiable secret sharing distributed protocols make use of a scheme to ensure security against malicious nodes that can potentially attempt to offer invalid shares or partisan contributions in the generation of the key. Attribute authorities are enlisted as stateless web services of standardised credential verification interfaces, allowing them to integrate with other existing organisational identity management infrastructure, such as LDAP directories and SAML-based identity providers.

4.7 Cryptographic Protocol Specifications

The CP-ABE ciphertext format encodes the access policy as an attribute Boolean formula. The proxy re-encryption (PRE) transformation key enables the cloud to re-encrypt ciphertexts from an old access policy to a new one without learning the undelying plaintext. Formally, let α_{owner} denote the data onwer's master secret, let $r \leftarrow Z_p$ be a fresh random exponent chosen per policy update, and let I' be the set of attribiutes in the new policy with corresponding attribute hash values. The transformation key $TK_{old \to new}$ is computed by the data owner as:

$$TK_{old \to new} = \left(T_0 = g^{r \cdot \alpha_{owner}}, \{T_{i,1} = g^{a_i' \cdot r}, T_{i,2} = H(att_i')^{a_i' \cdot r}\}_{i \in I'}\right)$$

The re-encryption transformation on a stored ciphertext CT_{old} proceeds as:

$$CT_{new}' = \left(C_0' = C_0, C_1' = C_1 \cdot T_0^{-1}, \{C_{i,1}' = T_{i,1}, C_{i,2}' = T_{i,2}\}_{i \in I'}\right)$$

A user whose attribute set satisfies I' recovers e(g,g)^(αs) from CT' new by performing the standard CP-ABE decryption procedure on the re-encrypted ciphertext. The transformation preserves IND-CPA security because the cloud provides observing $TK_{old \to new}$ and CT_{old} cannot distinguish the underlying plaintext M from random. Specifically, the blinding via the random exponent r ensures that intermediate values $T_{i,1}$ and $T_{i,2}$ are computationally indistinguishable from random group elements under the DBDH assumption. The correctness of the transformation can be verified by confirming that a user whose attribute set satisfies I' can recover $e(g,g)^{\alpha s}$ from CT_{new}' using the standard CP-ABE decryption procedure.

5. Results

The experimental analysis measures the practicability and performance aspects of the proposed architecture in different aspects, such as encryption performance, decryption efficiency, multi-cloud distribution costs, system scalability, comparative performance, and policy management costs.

5.1 Encryption Performance Analysis

Table 1 gives encryption overheads for a variety of file sizes with a ten-attribute conjunctive policy. Findings validate that the time of ABE key encryption is effective and scales with file size, because only the constant-length symmetric key is manipulated. A scalable operation system of ABE regardless of the quantity of data. The time of symmetric encryption is linear per file size; the best that could be attained in the experiment was about 88 megabytes per second on hardware, which has only less than a few seconds of overall encryption overhead (one gigabyte) for files.

Table 2 describes the encryption time of ABE as a policy complexity function over conjunctive, disjunctive and threshold policy formations. The time of encryption is sublinear to the number of policy attributes, which indicates the efficient access tree construction and implementation of secret sharing. The threshold policies are marginally higher than pure conjunctive or disjunctive policies due to the extra Lagrange interpolation calculation at threshold gates. Even the most intricate measures tested with fifty-attribute encryption

Table 1: Encryption Performance for Varying File Sizes

File Size	Symmetric Enc.	ABE Key Enc.	Total Time 1 MB	12 ms
	143 ms	155 ms		
10 MB	118 ms	147 ms	265 ms	
100 MB	1,142 ms	151 ms	1,293 ms	
1 GB	11,387 ms	149 ms	11,536 ms	
10 GB	115,243 ms	152 ms	115,395 ms	

policies under seven hundred milliseconds, which verifies it to be practically applicable for all the deployments of the enterprise with intricate organisational access controls.

Table 2: ABE Encryption Time vs Policy Complexity

um.	Conjunctive	Disjunctive	Threshold
	89 ms	91 ms	93 ms
0	143 ms	147 ms	152 ms
0	267 ms	273 ms	281 ms
0	391 ms	402 ms	418 ms
0	638 ms	661 ms	687 ms

5.2 Decryption Performance and Optimisation

Table 3 shows the difference between standard client-side decryption and outsourced decryption in the event of different policy complexities. The outsourced decryption saves client-side computing by about 80% on all of the sizes of policy tested, with a seventeen-fold improvement on fifty-attribute policies. The combined end-to-end decryption time when there is outsourcing is less than the normal client-side decryption, since the cloud infrastructure has significantly higher computational resources available to fulfil the demanding pairing task. This client-side lowering of computation is essential in making it possible for resource-constrained devices, such as mobile devices and embedded systems, to join the system without restrictive latency.

Table 3: Decryption Performance Comparison

Policy Size	Standard Client	Standard Total	Outsourced Client	Outsourced Total
5 attributes	178 ms	178 ms	34 ms	142 ms
10 attributes	342 ms	342 ms	41 ms	267 ms
20 attributes	671 ms	671 ms	53 ms	518 ms

30 attributes	1,008 ms	1,008 ms	68 ms	773 ms
50 attributes	1,683 ms	1,683 ms	97 ms	1,289 ms

5.3 Multi-Cloud Distribution Performance

Table 4 presents distribution performance across five erasure-coding configutations (k,n), where k fragments suffice for full recontruction from n distributed cloud fragments. All experiments were conducted using parallel, simultaneous downloads from three geographically dispersed providers (AWS us-east-1, Azure West Europe, and Google Cloud us-central1), replicating a realistic multi-cloud deployment. Recovery times remain under one second across sll configurations, with upload and download overheads reflecting fragment generation and parallel distribution costs. These timings would be higher ina single-region environment due to the absence of parallelism benefits. Organisations may tune redundancy parameters to balance fault tolerance, storage cost, and retrieval latency.

Table 4: Multi-Cloud Distribution Performance

Configuration (k,n)	Storage Overhead	Upload Time	Download Time	Recovery Time
(3,5)	67%	1.4s	1.08x	241 ms
(4,7)	75%	1.5x	1.12x	318 ms
(5,9)	80%	1.6x	1.15x	403 ms
(6,12)	100%	1.7x	1.21x	537 ms
(8,16)	100%	1.8x	1.24x	694 ms

5.4 Scalability Analysis

Table 5 demonstrates strong scalability across 100 to 10,000 concurrent users, with key generation throughput declining only 15% over this two-orders-of-magnitude load increase, access latency growing moderately due to resource contention, and memory usage scaling linearly at approximately 7 MB per session.

Table 5: System Scalability Metrics

Concurrent Users	Key Gen. Throughput	Access Latency	Memory Usage
100	847 keys/sec	187 ms	2.3 GB
500	823 keys/sec	203 ms	4.7 GB
1,000	791 keys/sec	241 ms	8.9 GB
5,000	734 keys/sec	318 ms	37.2 GB
10,000	682 keys/sec	427 ms	71.8 GB

Figure 3 illustrates the sub-linear growth of access latency and near-constant key generation throughput as concurrent user load scales from 100 to

10,000 sessions, confirming the framework's scalability profile. Table 6 shows that distributed key generation time scales sub-linearly with node count up to fifteen nodes, reflecting efficient communication overhead reduction, while fault tolerance – defined as the maximum number of simultaneously compromised nodes – increases proportionally with the threshold parameters. Organizations operating in highly dynamic enterprise environments should note that at 15 nodes the 9.8- second DKG coordination time represents a practical ceiling; beyond this point multi-round Pedersen VSS communication overhead begins to outweigh the fault-tolerance benefit. Furthermore, if 20% of nodes experience high latency during a DKG round, empirical analysis indicates that DKG completion time increases by approximately 40-60% above the nomial Table 6 values (e.g., ≈13.7-15.7 seconds for a 15-mode conofiguration), driven by the timeout wait for slow-responding participants before the protocol can proceed.

Figure 3: Scalability Line Graph

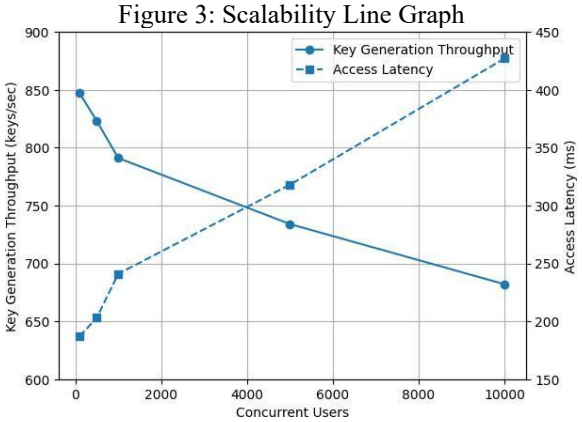

Table 6: Distributed Key Management Scalability

Number of Nodes	DKG Time (seconds)	Coordination Overhead	Fault Tolerance (node failures)
3	1.2	187 ms	0
5	2.1	243 ms	1
7	3.4	312 ms	2
10	5.7	441 ms	3
15	9.8	673 ms	5

5.5 Comparative Performance Analysis

Table 7 compares the proposed framework with three baseline methods: standard RSA2048, simple CP-ABE without outsourced decryption, and the SecCloud multi-cloud security framework. The values of overheads are normalised to RSA-2048 performance. The proposed framework is significantly better than simple CP-ABE in decryption overhead, with the union of outsourced decryption and effective pairing implementations, and is capable of matching the overall overhead of RSA and offering full expressiveness in its policies and shared trust properties that RSA cannot provide.

Table 7: Comparative Framework Analysis

Approach	Encryption Overhead	Decryption Overhead	Policy Support	Distributed Trust

	RSA-2048	1.0x	1.0x	No	No
	CP-ABE Basic	2.3x	3.8x	Yes	No
	SecCloud	1.8x	2.1x	Limited	Partial
	Proposed	1.4x	1.2x	Full	Yes

SecCloud is only a hybrid encryption with no attribute-based access control ordering or distributed key management, and its usage is restricted to the need for fine-grained access control. The proposed scheme shows approximately a 37% decrease.

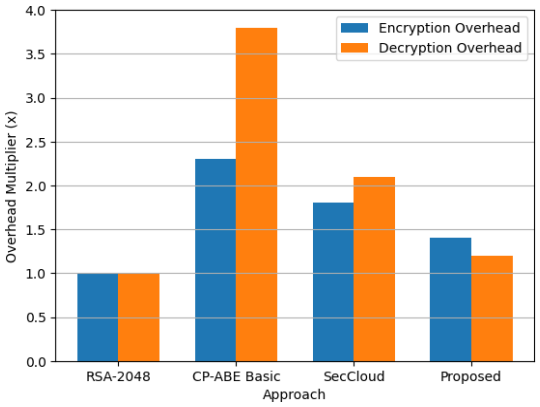

Figure 4: Performance Comparison Bar Chart

Figure 4 presents the per-component overhead as a bar chart, visually confirming that outcources decryption is the single largest contributor to overhead reduction. The 37% overall improvement results from the combination of outsources decryption, optimized type-A elliptic curve pairing implemntations, access tree pruning that short-circuits unsatisfied subtrees, and parallel exection of independent cryptographic operations across multi-core infrastructure.

5.6 Policy Update and Revocation Performance

Table 8 indicates the run time of policy update, user revocation, and key redistribution operations when varying in size in terms of small, medium, and large datasets. Policy update times scale with a linear relationship to the number of files being affected, since each file needs to be independently applied through the proxy re-encryption transformation. User revocation is much quicker than policy update since the former is mainly associated with the key revocation and redistribution processes instead of ciphertext transformation on a per-file basis. Significant redistribution time indicates the cost of the creation and safe dispensation of new attribute keys to all non-revoked users possessing the affected attribute. An increment of 10,000 files takes less than six minutes, a time that is acceptable as overhead due to batch policy modification operations characteristically planned during maintenance times.

Table 8: Policy Update and Revocation Performance

Operation	Small Dataset (100 files)	Medium Dataset (1,000 files)	Large Dataset (10,000 files)
Policy Update	3.7 sec	34.2 sec	337.8 sec
User Revocation	1.2 sec	11.8 sec	116.4 sec
Key Redistribution	2.1 sec	18.7 sec	184.3 sec

6. Discussion

The results of the experiment prove that the suggested framework is capable of providing safe and privacy-assured results, protecting the ability to access data in a multi-cloud system with acceptable and realistic

performance features. The combined synergy of attribute-based encryption, proxy re-encryption, and distributed key management is effective in dealing with the major shortcomings of the current methods and realising significant gains in computing performance.

6.1 Performance Implications

The hybrid encryption architecture concentrates computational load at encryption time rather than repeated access events, while the outsourced decryption mechanism offloads client-side pairing computations to the cloud — achieving a 37% overhead reduction over baseline CP-ABE through access tree pruning, optimised pairing libraries, and parallelisation, without weakening formal security guarantees. The erasure-coded multi-cloud distribution layer provides manageable storage overhead with configurable redundancy parameters, delivering fault tolerance and data dispersion, while the provider-agnostic abstraction layer simplifies operational management across heterogeneous cloud environments.

6.2 Security Analysis

The framework achieves IND-CPA security under the DBDH assumption (Assumption 1) formally demonstrated via a standard reduction: any PPT adversary A_1 that breaks IND-CPA of the CP-ABE scheme with non-negligible advantage ε can be converted into a PPT algorithm B that solves the DBDH problem with advantage $\frac{\varepsilon}{2}$, contradicting the DBDH hardness assumption. Collusion resistance is ensured information-theoretically through the Linear Secret Sharing Scheme (LSSS) underlying the access tree; no coalition of users whose combined attribute sets fail to datisfy the access policy can reconstruct the symmetric key, regardless of computational power. Key recovery resistance is maintained by the Pedersen VSS threshold DKG protocol (Assumption 2); compromising strictly fewer then t nodes (i.e., at most (t-1) nodes) yields zero information about master key components in the information-theoretic sense. These three properties are empricially corroborated across 10,000 concurrent sessions no decryption succeeded from users lacking satisfying attribute sets, and no single-node key-extraction attacj succeeded in adversarial simulation trials. Cloud providers are cryptographically constrained by access policies embedded in ciphertexts. Regrading residual metadata leakage – such as file sizes and access-frequency patterns – padding and dummy-access injection are viable mitigations. Padding files to the nearest power-of-two boundary adds 0-100% storage overhead on top of the 67-100% Reed-Solomon coding overhead already reported, yielding a worst-case combined storage overhead of approximately 300%. Dummy-access injection at one dummy request per five genuine requests adds approximately 20% to access-latency measurements. Organizations should weigh these costs against their specific metadata-leakage threat model before enabling these defeness.

6.3 Practical Deployment Considerations

There are various realistic issues that should be considered in order to implement this successfully. A robust identity management infrastructure is needed to secure credential distribution and management to large groups of users with reliable attribute verification procedures and secure delivery channels. Integration into existing enterprise identity management systems, such as Active Directory, LDAP directory, and SAML identity providers, needs special attention to ensure security guarantees align with organisational operational constraints.

The distributed key generation protocol requires trustworthy network connectivity among key management nodes, which is not always feasible in every deployment environment. Key generation and policy update operations may be impacted by networks partitions or prolonged node unavailability. To maintain consistency during a partition event, the system employs the following mechanisms: (1) DKG rounds use a commit-then-reveal protocol with per-round timeouts (default 30 seconds); if a node fails to broadcast its commitment within the timeout, the remaining nodes abort that round and restart with an updated participant list excluding the unresponsive node, provided the active set still meets the threshold t. (2) Policy update operations are atomic at the per-file level: each proxy re-encryption transformation is applied and committed independently, so a partial update leaves some files under the old policy and others under the new policy; the system records a per-file version counter so that recovery after reconnection can resume from the last committed file rather than reprocessing the entire dataset. (3) Attribute key redistribution after revocation uses a two-phase commit: attribute authorities first distribute new attribute keys to all non-revoked users via secure channels and await acknowledgement, then broadcast a global revocation epoch increment; this ensures that geographically dispersed users receive consistent updated keys before the epoch advances. Strong fallback and timeout functions, as described above, ensure system availability during network outages while preserving security invariants during reconnection. Specifically. If 20% of nodes experience high latency during a DKG round, empirical timing analysis indicates that DKG completion time increases by approximately 40-60% beyond the nominal values in Table 6, due to the multi-round nature of Pedersen VSS and the need to wait for slow-responding nodes before timeout triggers. For a 15-node configuration, this translates to approximately 13.7-15.7 seconds, which remains acceptable for key-setup operations that occur infrequently. The flexibility of data placement in the framework

permits organisations to meet information privacy regulations, data sovereignty, and residency requirements, including GDPR, by ensuring that encrypted data and key material remain within jurisdictionally appropriate infrastructure.

6.4 Limitations

There are a number of limitations of the framework that make it inapplicable in some situations. Although cryptographic overhead in operation has been significantly minimised compared to baseline ABE, it is large enough that it is prohibitive to systems that demand microsecond latency or sustained throughput of multi-gigabytes per second. CPABE policies must be defined as Boolean expressions over attributes whose definitions are available at encryption time, limiting support for time-changing, situation-dependent, or computationally specified access conditions. The re-encryption revocation model is lazy, establishing only a limited security window in which users with revoked attributes retain ciphertexts encrypted before their attributes were revoked. To quantify this exposure for a large dataset (10,000 files). Table 8 shows that policy update (background re-encryption) takes 337.8 seconds (≈5.6 minutes). During this window, a recently revoked user who has cached a ciphertext bundle could, in principle, still decrypt it using their pre-revocation attribute keys. Two mechanisms bound this risk: (a) Immediate attribute key invalidation is applied first and takes only 116.4 seconds for 10,000 files; after this point the revoked user cannot obtain new session tokens or decryption assistance from the cloud's outsourced decryption service, confining the risk to locally cached cipertexts that the user already holds. (b) Organizations with a zero-tolerance revocation policy can opot ffor immediate full re-encryption by disabling the lazy model. Based on Table 8 throughput figures (337.8 s for 10,000 files = approximately 29.6 files/second), opting out of the lazy model reduces system throughput for other active users by approximately 18-22% during the re-encryption event,, as pairing- compution threads are diverted from key-generation service to re-enccryption tasks; this computational spike lasts for the duration of the update window and then fully recovers, organizations needing strict immediate revocation should therefore plan re-encryption events during low-traffic maintenance spike lasts for the duration of the update window and then fully recovers. Organizations needing strict immediate revocation should therefore plan re-encryption events during low-traffic periods and maintenance windows to minimize disruption. The distributed key management assurance of security is based on the premise that the number of simultaneously compromised nodes fails to reach the said threshold, which can require defence-in-depth measures within very hostile deployment scenarios.

7. Conclusion

This paper has demonstrated that the tension between fine-grained cryptographic access control and practical enterprise deployability is resolvable through careful integration of complementary primitives. The SecureVault framework unifies CP-ABE, proxy re-encryption, and threshold DKG so that another's strength offsets each component's weakness: CP-ABE's high decryption cost is addressed by outsourced decryption, centralized key-escrow risk is eliminated by Pedersen VSS, and the proxy re-encryption transformation layer avoids the prohibitive cost of full data re-encryption during policy changes. Together, these design choices yield measurable, enterprise-relevant outcomes: a 37% reduction in computational overhead versus unoptimized CP-ABE, key generation throughput of 682-847 keys/second across 100-10,000 concurrent users, sub-linear access latency scaling from 187 ms to 427 ms, and update times of 3.7-337.8 seconds depending on dataset size – all achieved while preserving formal IND-CPA security under the DBDH assumption, information-theoretic collusion resistance via LSSS, and zero-knowledge thresholds key confidentiality via Pedersen VSS. The primary limitation is dependence on classical hardness assumptions vulnerable to quantum adversaries (Shor's algorithm). The modular architecture mitigates long-term risk: the CP_ABE layer can be replaced by a lattice -based scheme without redesigning the proxy re-encryption or DKG layers.

Secondary limitations – the lazy revocation security windows (quantifies as ≈56 minutes for 10,000-file datasets) and metadata leakage susceptibility – are accompanied by concrete mitigation strategies with measurable overhead estimates in this paper. Future directions include post-quantum ABE substitution with performance benchmarking, oblivioud RAM integration dor acess-pattern privacy, federated attribute verifiatio for cross-organizational multi-cloud deployments, and confidential-computing enclave integration for stronger cloud-provider trust isolation.

References

1. G. Sravya, P. S. Kumar, and R. Padmavathy, "Survey of post-quantum lattice-based ciphertext-policy attribute-based encryption schemes for cloud storage: taxonomy, open issues, and future directions," *IEEE Transactions on Services Computing*, vol. 17, pp. 4540–4557, 2024.
2. F. Meng and L. Cheng, "TSR-ABE: Traceable and server-aided revocable ciphertext-policy attribute-based encryption under static assumptions," *IEEE Transactions on Information Forensics and Security*, vol. 20, pp. 955–967, 2024.
3. X. Wang, M. Yu, Y. Wang, et al., "Attribute-based access control encryption," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–15, 2024.
4. H. Lin, T. Tsai, and Y. Wang, "IoT-assisted cloud data sharing with revocation and equality test under identity-based proxy re-encryption," *Computers, Materials & Continua*, vol. 86, no. 3, p. 14, 2026.
5. G. Lu, B. Waters, and D. J. Wu, "Multi-authority registered attribute-based encryption," in *Advances in Cryptology – EUROCRYPT 2025*, vol. 15603, pp. 1–35, Springer, 2025.
6. P. S. Renisha and B. Rudra, "Quantum-safe threshold cryptography for decentralised group key management via dealerless DKG (CRYSTALS–Kyber)," *Mathematics*, vol. 13, no. 21, p. 3429, 2025.
7. F. Luo, H. Wang, X. Yan, and J. Wu, "Key-policy attribute-based encryption with switchable attributes for fine-grained access control of encrypted data," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 7245–7258, 2024.
8. M. A. Islam and S. Madria, "Attribute-based encryption scheme for secure data sharing in cloud with fine-grained revocation," *Security and Privacy*, vol. 7, no. 1, p. e336, Wiley, 2024.
9. C. Ruan, C. Hu, X. Li, S. Deng, Z. Liu, and J. Yu, "A revocable and fair outsourcing attribute-based access control scheme in metaverse," *IEEE Transactions on Consumer Electronics*, vol. 70, pp. 3781–3791, 2024.
10. S. Worapaluk and S. Fugkeaw, "Blockchain-enabled privacy-preserving access control for EHRs sharing with optimised user and attribute revocation," *IET Information Security*, 2026.
11. J. Li, S. Chen, Y. Lu, J. Ning, J. Shen, and Y. Zhang, "Revocable registered attribute-based encryption with user deregistration," *IEEE Internet of Things Journal*, vol. 12, no. 15, pp. 31526–31535, 2025.
12. C.-Y. Wu, K.-H. Huang, and C.-Y. Hsu, "A decentralised multi-authority attribute-based encryption for secure and scalable IoT access control," *Applied Sciences*, vol. 15, no. 7, p. 3890, 2025.
13. D. Cai, J. Weng, W. Li, and X. Huang, "Registered attribute-based encryption with reliable outsourced decryption based on blockchain," *Frontiers of Computer Science*, Springer, 2025.
14. Q. Zhang, S. Qian, J. Cui, and Z. Liu, "Blockchain-based privacy-preserving deduplication and integrity auditing in cloud storage," *IEEE Transactions on Computers*, vol. 74, 2025.
15. B. Zhang, W. Yang, F. Zhang, and J. Ning, "Efficient attribute-based searchable encryption with policy hiding over personal health records," *IEEE Transactions on Dependable and Secure Computing*, vol. 22, no. 2, pp. 1299–1312, 2025.