



Design and Implementation of an Attribute-Based Encryption to Enhance Privacy in Federated Identity Management (FIM) Systems for Cloud Computing

Andrews Ocran¹

¹Teesside University

(C2372334)

Submitted in Partial Requirements for the Degree of MSc Information and Cybersecurity

Publication Date: 2025/04/09

How to Cite: Andrews Ocran (2025). Design and Implementation of an Attribute-Based Encryption to Enhance Privacy in Federated Identity Management (FIM) Systems for Cloud Computing. *International Journal of Innovative Science and Research Technology*, 10(3), 2384-2421. <https://doi.org/10.38124/ijisrt/25mar657>

ABSTRACT

With the increasing adoption of cloud-based services and distributed systems, securing user identity and sensitive data in Federated Identity Management (FIM) systems has become a critical challenge. Traditional authentication and authorization mechanisms often fall short in ensuring fine-grained access control, especially when dealing with large-scale, dynamic environments. This study explores the enhancement of security in Federated Identity Management (FIM) systems through the integration of Attribute-Based Encryption (ABE), a promising cryptographic technique that offers advanced access control based on user attributes rather than single user identity. The proposed model utilizes Ciphertext-Policy ABE (CP-ABE) to ensure dynamic encryption of user data while also ensuring that only users with the appropriate matching attributes can decrypt and access specific information.

By deploying Attribute-Based Encryption, the system enhances privacy, reduces the risk of unauthorized access, and addresses common vulnerabilities in federated systems, such as credential theft and unauthorized privilege escalation.

Through a series of experiments, this study evaluates the feasibility and effectiveness of the proposed system in real-world scenarios. The findings suggest that integrating Attribute-Based Encryption into Federated Identity Management systems significantly strengthens security, provides more flexible and granular access control, and mitigates risks associated with traditional identity management approaches. This work will contribute to the field by offering a novel approach to securing federated identity systems in dynamic and complex environments, with implications for both academia and industry in cloud computing, cybersecurity, and privacy-preserving technologies.

TABLE OF CONTENT**CONTENTS**

ABSTRACT:	2385
TABLE OF FIGURES:	2387
LIST OF ABBREVIATIONS	2388
CHAPTER ONE	2389
1.Introduction:	2389
1.1. Background of the Study	2389
1.2. Statement of the Problem	2389
1.3. Research Objectives	2389
1.4. Research Questions	2389
1.5. Significance of the Study:	2390
1.6. Scope of the Study:	2390
CHAPTER TWO	2391
2.Literature Review	2392
2.1. Overview of Federated Identity Management System	2391
2.2. Core Components of Federated Identity Management System	2391
2.3. Federated Management System Architecture	2392
2.3.1 Authentication Process in Federated Identity Management	2392
2.3 Existing Cryptographic Techniques in Federated Identity Management Systems	2393
2.4 Security and Privacy Challenges in Federated Identity Management Systems	2393
2.5 Proposed System Model	2393
2.5.1 Types of Attribute Based Encryption.	2395
2.5 Benefits of The Proposed System	2395
2.6 Review of Related Work.	2395
2.7. Identification of Gaps in The Literature Review	2396
2.8 Connection to Current Research Question.	2397
CHAPTER THREE	2398
3. Methodology	2398
3.1 Introduction	2398
3.2 Design and Create Process Model	2398
3.3 Tools, Software, and Resources	2399
3.3.1 Development Tools and Environment:	2399
3.4 Threat Model.	2399
3.4.2 Potential System Adversaries.	2399
3.4.5 Security Model.	2400
3.4.6 Proposed Security Model.	2400
3.5 Ethical Considerations.	2401
CHAPTER FOUR	2402
4. Implementation	2402
4.1 Introduction.	2402
4.2. Algorithm for Implementation of The Proposed System	2403
4.3 Program Implementation.	2405
4.3.1 Component of The Proposed System Model.	2405
CHAPTER FIVE	2408
5. Results and Discussions	2408
5.1 Introduction	2408
5.2. Install Required Software.	2408
5.3. Setup database connection to MySQL.	2408
5.4. Access the Application:	2409
5.5. Discussion	2415
5.6 Future Research Direction.	2416
CHAPTER SIX	2417
6.Conclusion	2417
6.1 Introduction	2417
6.2 Summary of Findings	2417
6.3 Contribution to Knowledge in the Field.	2418
6.4 Recommendation for Practitioners and Policy Makers.	2418
6.5 Research Limitation	2418
REFERENCES	2419

TABLE OF FIGURES

Figure 1: Federated Identity Management System	2391
Figure 2: Federated Identity Management Architecture (Raj, 2022)	2392
Figure 3: Attribute Based Encryption System.	2394
Figure 4: Oates Design and Creates Model	2398
Figure 5: Threat Model Network Diagram.	2400
Figure 6: pseudocode to implement the proposed algorithm	2402
Figure 7: System Implementation Flowchart	2403
Figure 8: Connection to MySQL Database	2405
Figure 9: Data User Login Backend	2405
Figure 10: Backend Data User Registration	2406
Figure 11: Encrypted File Upload with Access Policy	2406
Figure 12: Retrieving User Data Based on Attributes	2407
Figure 13: Dynamic Policy Update	2407
Figure 14: XAMP to host resources locally	2408
Figure 15: XAMP Interface	2408
Figure 16: MySQL Database Interface	2409
Figure 17: Application folder -htdocs	2409
Figure 18: Application folder - ABE folder	2409
Figure 19: System interface	2410
Figure 20: User Registration	2410
Figure 21: User not approved to register to the system	2410
Figure 22: User login	2411
Figure 23: User Dashboard	2411
Figure 24: Data user file search	2412
Figure 25: Approved files	2412
Figure 26: Data Owner Homepage	2412
Figure 27: File Upload	2415
Figure 28: Data Owner File Request Approval	2415
Figure 29: Data Owner view Attribute	2416
Figure 30: Attribute Authority Dashboard	2416
Figure 31: Attribute Provider Approval of Registration	2417
Figure 32: Cloud Storage	2417
Figure 33: Gantt Chart Showing Timeline for The Project	2419

LIST OF ABBREVIATIONS

(Application Program Interface).	
API	7
(Security Assertion Markup Language)	
SAML.....	6
Advance Encryption Standard	
AES	8
Attribute-Based Access Control	
(ABAC)	16
Attribute-Based Encryption	
(ABE)	1
Attribute-Based Encryption (ABE) Method with Verifiable Outsourced Encryption and Decryption	
(ABE-VOED).....	14
ciphertext	
(`CT`)	28
Ciphertext-Policy Attribute-Based Encryption	
(CP-ABE).....	4
Circle of Trust -	
(CoT)	6
Collaborative-Ciphertext Policy-Attribute Role-based Encryption	
(C-CP-ARBE)	14
Elliptic Curve Digital Signature Algorithm	
(ECDSA)	15
Enabling Fine-Grained Access Control	
(FGAC)	13
Federated Identity Management	
(FIM)	1
Fully Homomorphic Encryption	
(FHE)	4
General Data Protection Regulation	
(GDPR)	9
identity providers	
(IdPs)	1
Integrated Development Environment	
(IDE)	20
Internet of Things	
(IoT)	13
Master Secret Key	
(`MSK`).....	28
Multi-Factor Authentication	
(MFA)	8
Organization-Based Access Control	
(OrBAC).....	14
Rivest–Shamir–Adleman	
RSA	8
service providers	
(SPs)	1

CHAPTER ONE

INTRODUCTION

➤ *Background of the Study*

The evolution of cloud computing has brought significant benefits such as cost efficiency, scalability, and convenience for both organisation and individuals. Cloud computing has reshaped how organizations manage their critical IT infrastructure in the cyberspace by providing scalable, on-demand access to computing resources, storage, and applications (Bogataj Habjan and Pucihar, 2017). With its widespread adoption, many organisations are migrating their services to the cloud platforms (Linthicum, 2019). However, with these advantages comes the critical challenge of maintaining security and privacy, especially concerning how user identities and access to data are managed in cloud environments (Abdulsalam and Hedabou, 2021).

Federated Identity Management (FIM) systems have emerged to streamline the management of user identities across multiple domains, enabling users to authenticate once and gain access to different services and resources across organizational boundaries. Federated Identity Management relies on established trust between identity providers (IdPs) and service providers (SPs) to share authentication and authorization data (Naik and Jenkins, 2017). While Federated Identity Management systems enhance user experience by reducing the need for multiple logins, they introduce security and privacy concerns. These systems often involve sharing sensitive user data across multiple service providers, increasing the potential for unauthorized data access and data breaches.

Given the growing concerns over privacy and security in cloud-based federated identity environments, Attribute-Based Encryption (ABE) has gained attention as a robust solution. Attribute-Based Encryption is an encryption algorithm where the decryption of data depends on user attributes rather than specific user identities. This provides a flexible and fine-grained access control mechanism, making it ideal for complex, distributed systems like cloud computing (Li et al., 2020).

➤ *Statement of the Problem*

The increase in cloud services has made it easier for organizations to collaborate and share resources, but it has also heightened the need for secure and privacy-preserving identity management systems. The traditional Federated Identity Management (FIM) systems often fall short in addressing critical privacy issues. When user identity information is shared between different service providers in the cloud, the potential for privacy breaches increases. This can occur in several ways, such as unauthorized access to sensitive data, identity correlation attacks, and data leakage due to the centralization of user identity information Wang et al. (2022).

Many Federated Identity Management systems rely on a centralized trust model, where identity providers are responsible for securely managing and sharing user credentials. If a single identity provider is compromised, the privacy of all users associated with that provider can be endangered.

Although various cryptographic techniques have been proposed to improve privacy in cloud computing, few have been successfully integrated into Federated Identity Management. One promising approach is Attribute-Based Encryption (ABE), which offers fine-grained access control by associating encryption keys with user attributes, such as roles, departments, or access levels (Mohammad, 2022).

Despite its potential, Attribute-Based Encryption has not been widely adopted in Federated Identity Management systems, and its implementation in cloud computing environments poses technical and performance challenges. Therefore, there is a need for an in-depth exploration of how Attribute-Based Encryption can be effectively implemented to enhance privacy in Federated Identity Management systems.

➤ *Research Objectives*

The main objective of this research is to design and implement Attribute-Based Encryption (ABE) to enhance privacy in Federated Identity Management (FIM) systems for cloud computing. The research also seeks to achieve the following objectives.

- To analyze the privacy risks and challenges mitigated by ABE in Federated Identity Management, and compare its effectiveness with traditional encryption techniques in cloud computing.
- To access the security benefits of integrating ABE in FIM systems, specifically focusing on its ability to protect sensitive user information from unauthorized access and data leakage.
- To develop a prototype FIM system integrated with ABE, and evaluate its effectiveness in ensuring secure and controlled access to sensitive data in cloud-based applications.

➤ *Research Questions*

To address the objectives outlined above, the following research questions will guide this study:

- What privacy risks and challenges are mitigated by Attribute Based Encryption in Federated Identity Management, and how does it compare to traditional encryption techniques in cloud computing?

- How does integrating Attribute Based Encryption in Federated Identity Management systems enhance security, specifically in protecting sensitive user data from unauthorized access and data leakage?
- How effective is a prototype FIM system integrated with ABE in ensuring secure and controlled access to sensitive data in cloud-based applications?

➤ *Significance of the Study:*

The significance of this study lies in its contribution to enhancing privacy in Federated Identity Management Systems, which are becoming increasingly critical in cloud computing. By leveraging Attribute-Based Encryption, this study seeks to address privacy challenges in cloud-based services. Attribute-Based Encryption uses user attributes rather than personal identifiers to control access to sensitive data (Fu et al., 2022). This ensures that only users possessing the required attributes can access specific information. As a result, it reduces the risk of unauthorized access and prevents identity correlation between services, maintaining a high level of user privacy in multi-party cloud environments (Annane et al., 2022). ABE minimizes data exposure by limiting the attributes disclosed during authentication and also provides granular control over who can access what data, making it difficult for malicious actors or unauthorized third parties to link user identities across different services. The anonymity and obfuscation features of ABE protect personal information while ensuring that services can still function efficiently (Yan et al., 2024).

ABE enables fine-grained access control, allowing access to data based on specific attributes rather than specific individual identities (Wang et al., 2024). This flexibility supports scalable management as organizations grow, ensuring that as user needs evolve, access control policies can dynamically change without overhauling the entire system. It also minimizes the administrative overhead typically associated with managing complex, hierarchical access policies in cloud systems. In dynamic cloud environment, where users' roles may shift frequently and access permissions are often temporary, ABE's attribute-based structure simplifies policy updates (Reshma Siyal and Long, 2024).

➤ *Scope of the Study:*

The scope of this study is limited to the design and implementation of Ciphertext-Policy Attribute-Based Encryption (CP-ABE) in Federated Identity Management (FIM) systems. The research focuses on designing and evaluating a system where user attributes, such as role and location, are used to control access to encrypted data stored in the cloud.

CHAPTER TWO

LITERATURE REVIEW

A. Overview of Federated Identity Management System

Federated Identity Management (FIM) is a framework, tools and resources that allows users to access several independent services with a single set of credentials (Aldosary and Alqahtani, 2021). Federated Identity Management is based on a trust relationship between Service Providers (SPs) and Identity Providers (IdPs) which allows users to authenticate once and obtain access to a wide range of resources across domains. This solution streamlines identity management for enterprises, improves user experience, and enhances security by centralizing authentication and access controls (Keltoum and Samia, 2017).

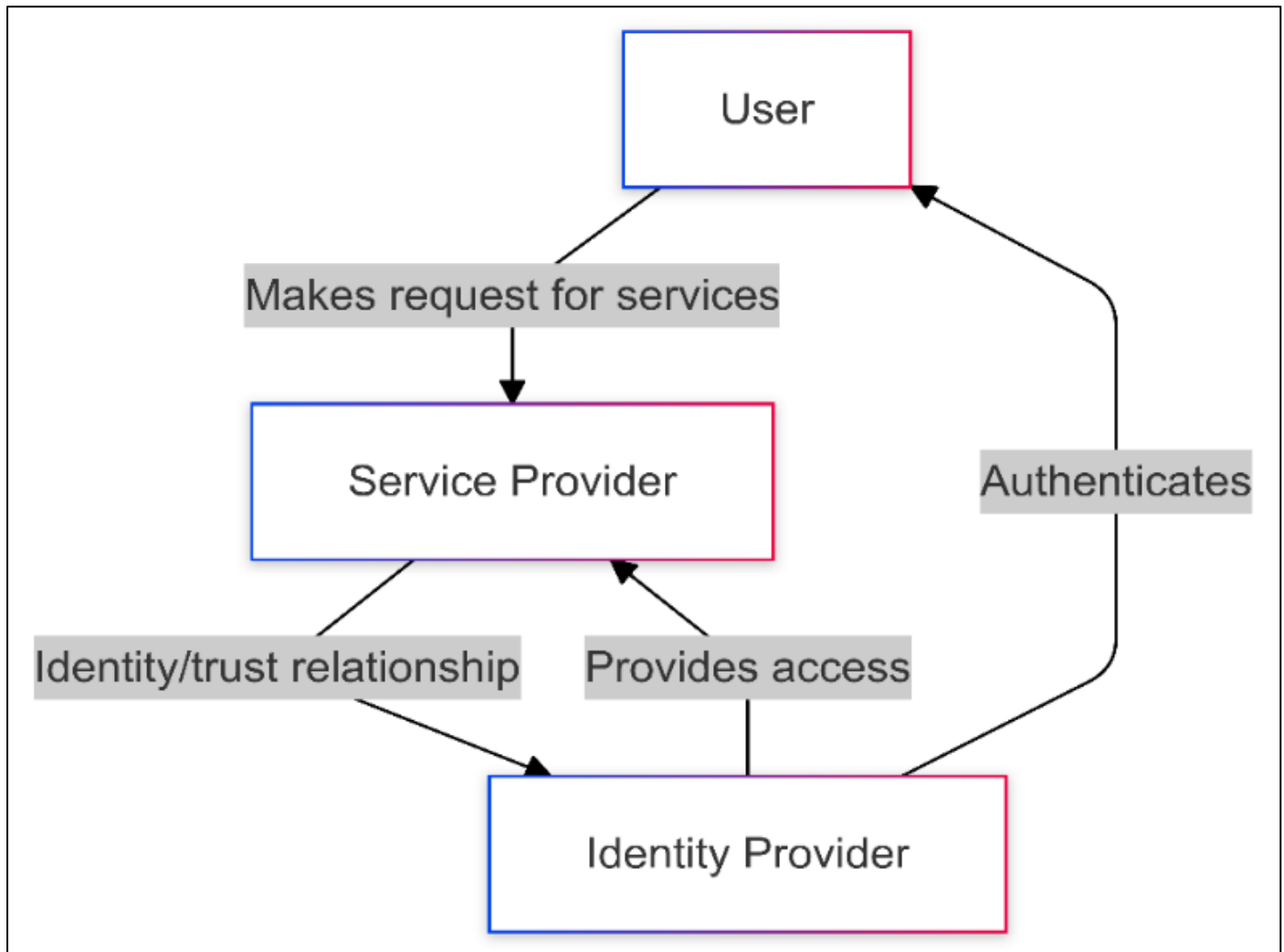


Fig 1 Federated Identity Management System

B. Core Components of Federated Identity Management System

Federated Identity Management systems are built on several key components that establish and manage the trust relationships among multiple entities:

- **Identity Provider (IdPs):** An Identity Provider (IdP) is a trusted authority in charge of authenticating user identities. In a cloud environment, the Identity Provider is responsible for authenticating user credentials (such as usernames, passwords, or biometric data) and confirming the user's identity. Following successful authentication, the Identity Provider generates an assertion, often a token containing authorized user information (Premarathne et al., 2017).
- **Service Provider (SP):** A Service Provider is the entity that provides the actual services or resources that a user wants to access (e.g., SaaS applications, cloud storage, or cloud-hosted web services). The Service Provider trusts the Identity Provider to authenticate the user and grants access based on the provided identity assertion (Panathula, 2024).
- **Authentication Protocols:** Authentication protocols provide a safe communication between the Identity Provider (IdP) and the Service Provider. They guarantee that user authentication and authorization data is transferred securely across several security domains without revealing sensitive information (Panathula, 2024). The three most widely used authentication protocols in FIM are SAML (Security Assertion Markup Language), OAuth, and OpenID Connect:

- **Assertion:** An Assertion is a message statement created by the Identity Provider (IdP) that confirms a user's identity and other related information to the Service Provider (SP). The assertion message vouches for the identity of the user, acting as proof that the user has been authenticated. Assertions include user attributes that define various characteristics of the user, such as their name, role, department, email, or any other information relevant to the SP (Panathula, 2024).
- **Trust Relationship:** A trust relationship is fundamental in federated identity management systems, its act as an agreement between participating entities (usually IdPs and SPs) to mutually recognize and accept each other's authentication and authorization decisions. This connection is necessary for federated systems to work because it allows one organization to trust the identity information and credentials certified by another.

C. Federated Management System Architecture

Before a user can access services in a federated environment, a trust relationship (Circle of Trust -CoT) must be established between the Identity Provider and Service Provider. This relationship is formalized using digital certificates, metadata exchanges, or signed agreements (Keltoum and Samia, 2017). The Identity Provider and Service Provider exchange public keys, metadata, and security configurations, often involving secure protocols such as TLS (Transport Layer Security) to protect the communication channel. This ensures that the Service Provider can validate identity assertions issued by the Identity Provider using cryptographic signatures.

➤ Authentication Process in Federated Identity Management

- **User Requests Access:** The process starts when a user tries to access a resource or service managed by a Service Provider (SP). This could be, for example, accessing a cloud-based application, an internal web portal, or an API (Application Program Interface). The user's browser sends an initial request to the Service Provider's endpoint. At this point, the user is not yet authenticated, so the Service Provider recognizes the need for identity verification before granting access.
- **Redirect to Identity Provider:** The Service Provider does not handle authentication directly instead, it redirects the user to an Identity Provider that is trusted to verify the user's credentials. This redirection includes an authentication request, which is typically encrypted to protect sensitive information (Kiourtis et al., 2023).
- **Authentication at Identity Provider:** At this stage, the user is redirected to the Identity Provider's authentication page. The user provides their credentials and the Identity Provider validates these credentials based on its internal database or directory service. Upon successful verification, the Identity Provider creates an identity assertion essentially a secure, verifiable statement about the user's identity (Mortágua, Zúquete and Salvador, 2024).
- **Return to Service Provider:** After successful authentication, the Identity Provider sends the identity assertion back to the Service Provider. This can occur in two ways: either through the user's browser or through direct backend communication between the Identity Provider and Service Provider.

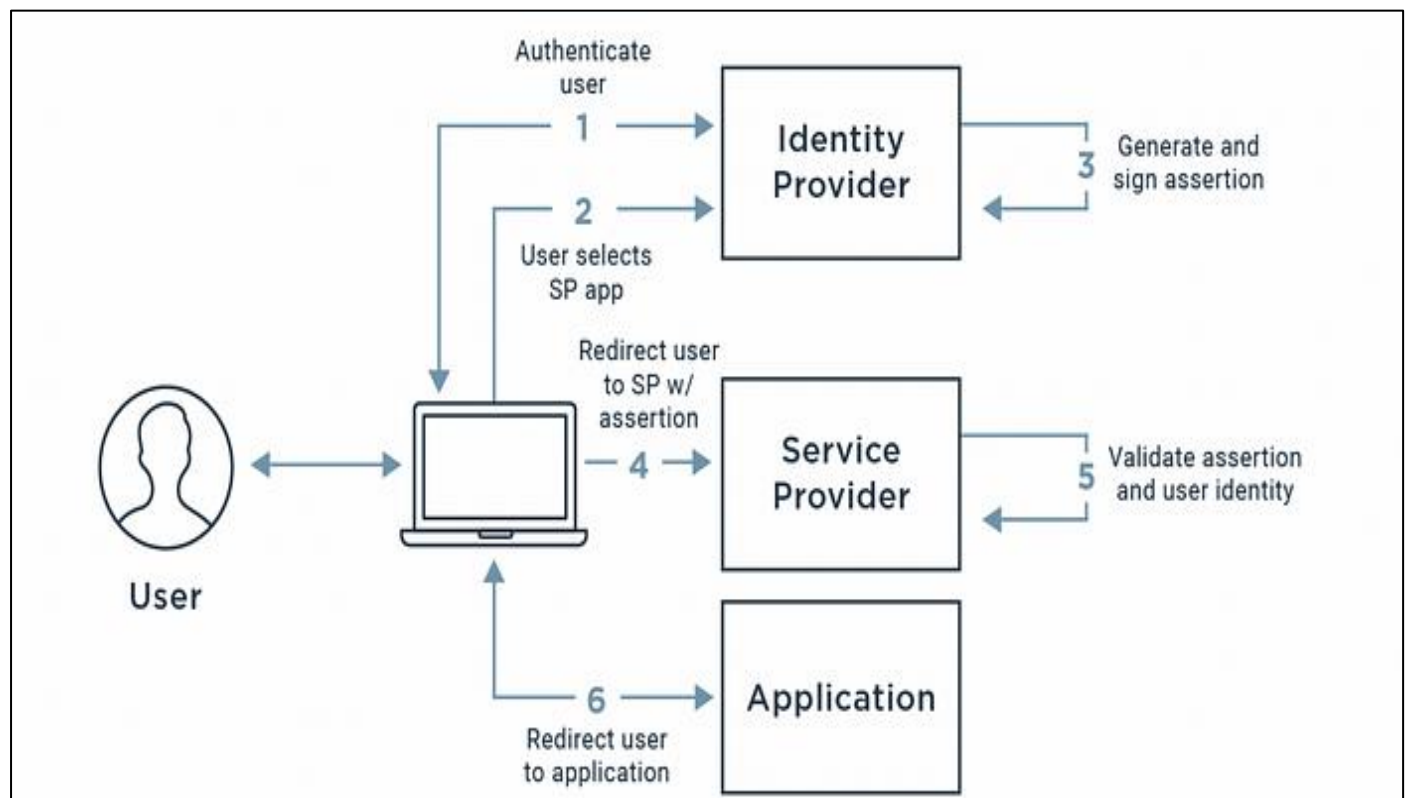


Fig 2 Federated Identity Management Architecture (Raj, 2022)

D. Existing Cryptographic Techniques in Federated Identity Management Systems

Existing security solutions for Federated identification Management (FIM) focus on managing identification, authentication, and access control across trusted domains. While they provide a solid basis of security, they have limits, especially in cloud and multi-domain environments where scalability, privacy, and fine-grained access control are critical.

- **AES (Advance Encryption Standard) and RSA (Rivest-Shamir-Adleman):** Traditional encryption standards like AES and RSA uses shared or public/private key pairs to encrypt and decrypt data, guaranteeing that only authorized parties may access it. While standard encryption is good at encrypting data, it lacks fine-grained access control (Hamza and Kumar, 2020). Access is frequently binary: either a user possesses the decryption key or they don't. This adds overhead in systems where data is often accessed and exchanged between domains.
- **Single Sign-On (SSO):** SSO enables users to authenticate once with a trusted Identity Provider (IdP) and access numerous services across domains without logging in again. SSO is based on protocols such as Security Assertion Markup Language (SAML), OAuth, and OpenID Connect. SSO improves user ease, it may introduce a single point of failure. If an IdP is hacked, unauthorised access to all resources on the federated platform is possible, potentially exposing sensitive data on the federated system (Abhijeet Thakurdesai et al., 2022).
- **Multi-Factor Authentication (MFA):** Multi-Factor Authentication increases security by requiring users to submit several forms of verification, such as a password and a fingerprint. Multi-Factor Authentication in Federated Identity Management may be used to secure access to resources across federated domains, helping to prevent illegal access to confidential data. However, it might degrade user experience by introducing unnecessary stages to the login process. MFA does not safeguard data on its own, hence it is commonly used in conjunction with additional security measures (Almadani et al., 2023).
- **Role-based Access Control (RBAC):** RBAC is an access control mechanism that allocates rights to users based on preset roles. In Federated Identity Management systems, RBAC controls which resources users may access depending on their responsibilities. RBAC is static and inflexible in contexts with changing user roles and access needs. Changes in access requirements frequently necessitate manual action to adjust roles and permissions. RBAC lacks the granularity to define access based on particular user data making it difficult to implement complicated, conditional access restrictions that need customization (McCarthy, 2023).

E. Security and Privacy Challenges in Federated Identity Management Systems

- **Data Exposure:** Federated Identity Management systems allow users to access multiple cloud-based services using a single set of credentials managed by an Identity Provider (IdP). While this simplifies authentication and improves user experience, it also introduces privacy and security risks, particularly around data sharing and access control. Service providers often request more attributes than necessary to fulfill their functions. This goes against the principle of data minimization, which suggests only the minimum amount of personal information necessary for a given purpose should be disclosed (Naik and Jenkins, 2017).
- **Lack of Fine-Grained Control:** Federated Identity Management often provides broad access based on generalized user roles, such as "employee" or "contractor," but lacks detailed control over specific data. This can result in over permissioning, where users have access to more resources than necessary. This leaves users vulnerable to privacy breaches if sensitive attributes are unnecessarily exposed to third-party service providers (Ma et al., 2020). Fine-grained access control could limit access to only the specific records relevant to system users.
- **Privacy Concerns with Data Sharing:** In Federated Identity Management, sharing personal information between IdPs and SPs raises privacy concerns, especially in regions with strict data privacy laws such as General Data Protection Regulation (GDPR). If sensitive user data is exchanged without adequate safeguards, it may lead to data breaches or unauthorized access to personal information (Alansari et al., 2017). If SPs do not adequately protect this data, it could be exposed or misused, raising compliance and privacy risks concerns.

F. Proposed System Model

The proposed system leverages Attribute-Based Encryption (ABE) to provide fine-grained access control to secure systems on cloud platform. Using a public key cryptosystem, the system associate's data access with user attributes such as positions, departments, or location rather than individual identities. This allows encryption rules to be consistent with organizational needs, enabling only authorized individuals with matching attributes to decrypt the data. The system will support Ciphertext-Policy ABE, which allows data owners to embed policies directly in encrypted data as shown in figure 3 below. ABE allows for fine-grained access control over encrypted data based on attributes rather than user identities.

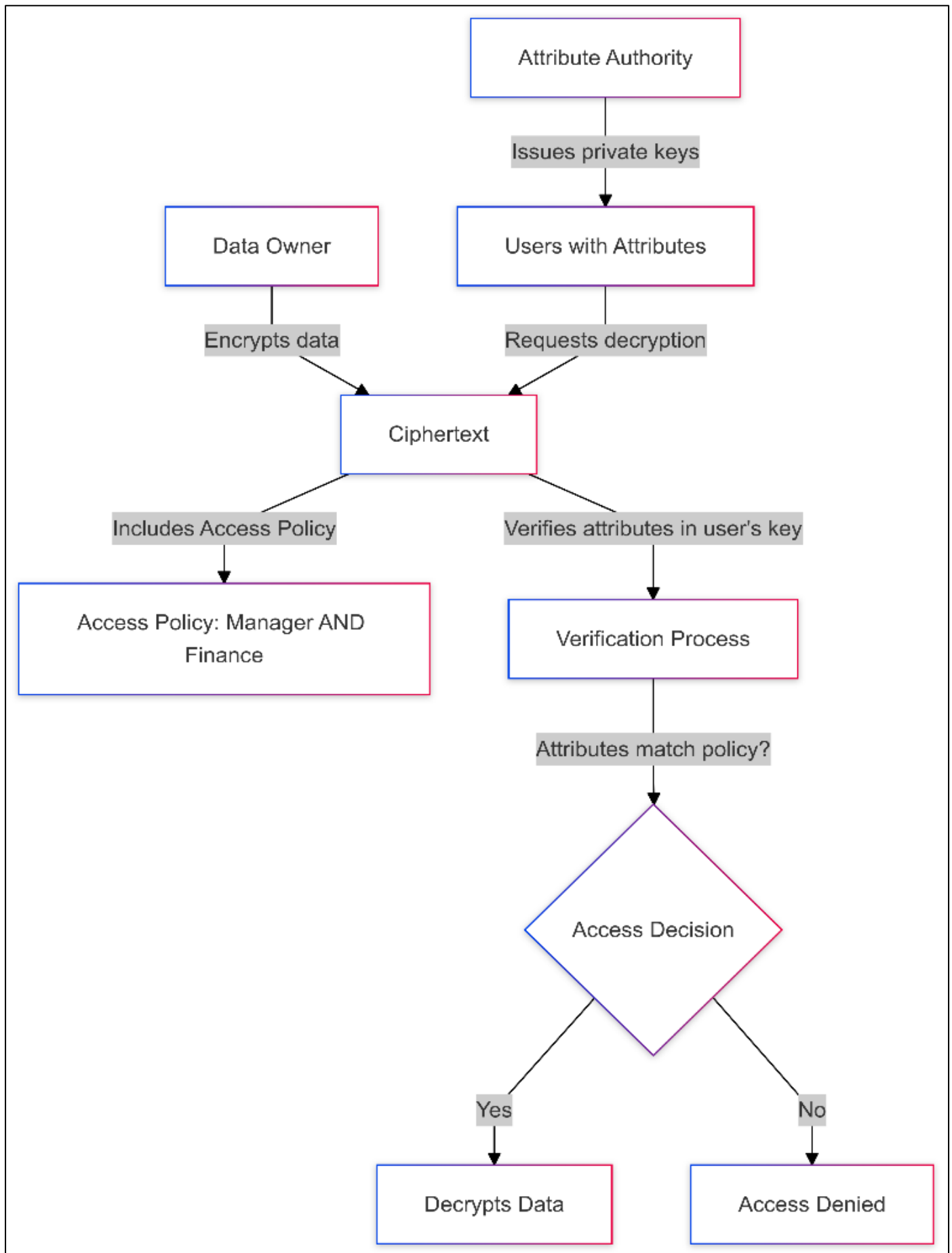


Fig 3 Attribute Based Encryption System.

➤ *Types of Attribute Based Encryption.*

Generally, Attributes-Based Encryption is classified into two main types: Ciphertext-Policy ABE (CP-ABE) and Key-Policy ABE (KP-ABE).

➤ *Ciphertext-Policy ABE (CP-ABE):*

Ciphertext-Policy ABE encryption algorithm allows data owner to specify a specific access policy in the ciphertext, and only users with the necessary set of attributes defined by their decryption keys can decrypt it. CP-ABE enables fine-grained access control in a safe and flexible manner, making it useful for applications that users must access encrypted data depending on criteria such as role, department, and location (Wang et al., 2023). A data owner might encrypt a file and set an access policy that restricts access to the content to individuals who possess the set attributes. The decryption procedure verifies if the attributes in the user's key meet the access policy included in the ciphertext when a user tries to decrypt it. A high degree of flexibility is also provided by CP-ABE's attribute-based design, which allows access decisions to be determined dynamically based on user attributes rather than static roles or identities as user responsibilities or rules change over time (Suryawanshi and Sural, 2024)

➤ *Key-Policy Attribute-Based Encryption (KP-ABE):*

In KP-ABE, ciphertexts are linked to a collection of attributes that characterize the data, and the access policy is explicitly encoded into the user's private key. Data owners can encrypt data using this structure without defining who has direct access to decrypt it. Rather, the policy included in the private key is used by the system to dynamically determine access. The data owner chooses a set of descriptive attributes that characterize the encrypted data at the start of the encryption process in KP-ABE (Luo et al., 2024). The access policy that is incorporated into a user's private key is usually expressed as an access tree or a Boolean formula that specifies which attribute combinations allow access (Wang et al., 2024).

G. Benefits of the Proposed System

- **Minimal Data Exposure:** The proposed system will ensure that only approved data users can access the resources necessary for a specific task, as defined by the encryption policy. Access to encrypted data is allowed in the ABE systems based on assigned attributes, eliminating the need to divulge sensitive user information. This selective exposure guarantees that only the attributes required for decryption are given, reducing the danger of information leakage while improving privacy and security. (Li et al., 2024)
- **Fine-Grained Access Control:** The system supports very detailed access controls based on user attributes like department, role, or location. This allows for exact control over who has access to data, rather than depending on static roles or identity-based permissions. Fine-Grained Access Control allows data owners to specify precise access policies, enabling a high level of customization in data sharing (Liu et al., 2018).
- **Data-centric Security:** Attribute-Based Encryption (ABE) enables safe data access by incorporating access controls directly into the encryption process. Data-centric security encrypts critical information so that only users with specified qualities may decrypt and access it, allowing for fine-grained control over data access (Amanowicz, Szwaczyk and Wrona, 2024). This is especially effective in situations where data is shared by several departments or individuals with varying access privileges, the system implements access restrictions directly at the data layer, lowering the danger of unauthorized exposure. The model adheres to data protection rules by ensuring that only eligible people, depending on specified attributes, gain access to sensitive data (Fun and Samsudin, 2017).
- **Enhanced Privacy:** ABE's encryption algorithm allow data owners to design exact access control policies without disclosing users' true identities. This technique reduces needless exposure of user identities and guarantees that sensitive information is only available to those who have matching attributes, hence improving privacy and preserving security and compliance (Yin et al., 2024).
- **Scalability and Flexibility:** Scalability in ABE originates from its structure, which embeds access controls in the encryption process based on attributes rather than assigning them explicitly to individual users (Amajuoyi, Nwobodo and Adegbola, 2024). This attribute-based solution allows a single encryption policy to be applied to many users without requiring direct control of each user's rights, making it extremely scalable with a large number of users or hierarchies. This flexibility is especially useful in applications like cloud storage and Internet of Things (IoT), where data access control might constantly change owing to dynamic data sharing requirements (Vignesh and Naresh, 2020).
- **Dynamic Policy updates.** In traditional encryption methods, modifying access permissions requires re-encrypting data and redistributing keys, which is time-consuming and prone to errors. However, the proposed ABE system allows policies to be updated dynamically without directly impacting the underlying data encryption.

H. Review of Related Work.

The traditional CP-ABE model, introduced by (Bethencourt, Sahai and Waters, 2007) revolutionized secure data sharing by enabling fine-grained access control based on user attributes. However, it faces ongoing issues with attribute plaintext visibility and its inability to authenticate users, leaving it susceptible to unauthorized access and data leakage. Attribute-Based Encryption (ABE) has received a lot of attention over the last decade as a privacy-preserving cloud security approach. Because of its fine-grained access control and data-centric encryption, ABE has been intensively researched for its use in securing sensitive data in cloud environment, addressing significant difficulties in privacy, scalability, and adaptability.

In their study on Enabling Fine-Grained Access Control (FGAC) in information sharing environments, (Niskanen and Salonen ,2023) emphasize the importance of secure and efficient sharing of sensitive information, particularly within complex multi-organizational networks such as supply chains and critical infrastructures. Their methodology tackles the difficulty of selectively providing access permissions to users by implementing access control at the granular level using structured data formats such as JSON. The authors offered a structured data sharing architecture for the marine logistics industry that secures cyber incident information while demonstrating Fine-Grained Access Control using Attribute-Based Encryption methods. The study investigates several access control techniques, including Role-Based Access Control (RBAC) and Organization-Based Access Control (OrBAC), to find the best strategy for Fine-Grained Access Control in distributed systems. They also explored the disadvantages of standard RBAC in multi-organization contexts, such as role explosion and inflexibility, and provide ABE as a suitable solution for circumstances that require dynamic and decentralized access control.

The researchers, (Li et al. ,2019) provided an efficient Attribute-Based Encryption (ABE) Method with Verifiable Outsourced Encryption and Decryption (ABE-VOED) to solve computational overhead concerns in cloud environments, particularly among mobile users. Traditional ABE methods have limitations due to the high computational cost of encryption and decryption, which grows with policy complexity. (Li et al. ,2019) addressed issues by providing a safe outsourcing model that outsources both encryption and decryption to untrusted servers, decreasing the load on data owners and consumers. The approach proposes an efficient algorithm that allows data owners and users to check the accuracy of outsourced calculations while maintaining security. The authors based their approach on the Brent Waters Ciphertext-Policy ABE (CP-ABE) concept, which they modified to include verifiable outsourced encryption and decryption. Their technology facilitates safe data exchange in mobile cloud by requiring users to conduct a limited number of lightweight actions, making it appropriate for resource-constrained devices.

Fugkeaw and Sato (2016) address significant limitations in managing encryption policies for ciphertext policy attribute-based encryption (CP-ABE) systems, especially in data outsourcing environments where access policies require frequent updates. In their study, they introduced a Collaborative-Ciphertext Policy-Attribute Role-based Encryption (C-CP-ARBE) model, which combines CP-ABE with role-based access control (RBAC) to streamline access policy management. By incorporating a very lightweight proxy re-encryption (VL-PRE) approach, they reduce the computational and communication burdens traditionally placed on data owners, allowing them to efficiently manage policy updates. This work builds upon prior research in CP-ABE and access control, where (Bendiab, Shiaeles and Samia , 2018) had already explored CP-ABE's dynamic policy updating capabilities for large-scale cloud data proposing solutions that handled attribute addition and removal without requiring complete data re-encryption. However, their approach required substantial computational resources for each update, posing challenges in scalability.

Similarly, (Yi et al., 2024) investigated Attribute-Based Proxy Re-Encryption to improve flexibility in policy updates, though their approach depended on key generation authority for re-encryption key generation, which constrained its efficiency in environments with frequent policy changes.

Imam et al. (2022) address these challenges in their systematic review, which looks at Attribute-Based Encryption (ABE) as a strong framework for controlling security in Electronic Health Records (EHR) and other sensitive health data. Attribute-Based Encryption, noted for its fine-grained access control, appears to be a potential option for data security in cloud computing, where standard cryptographic solutions frequently fail to provide the flexibility and scalability needed by healthcare applications.

(Zhang et al. ,2023) proposed an improved Ciphertext-Policy Attribute-Based Encryption (CP-ABE) algorithm to address critical vulnerabilities in the traditional CP-ABE model, such as user privacy leakage, identity forgery, and computational inefficiencies. These challenges are rooted in the plaintext exposure of attributes in private keys and ciphertexts, a lack of robust user identity verification mechanisms, and significant overhead during key generation and decryption. To mitigate these shortcomings, Zhang et al. (2023) integrate cryptographic advancements and innovative access control techniques. They introduce a hybrid signature system combining the Elliptic Curve Digital Signature Algorithm (ECDSA) with Dilithium, a post-quantum cryptographic algorithm, to provide robust identity verification. This hybrid signature ensures resistance to both classical and quantum attacks, enhancing the system's security against identity forgery. They further enhance data security by employing permutation-based encryption to conceal plaintext attributes, thereby protecting access policies from unauthorized users.

Furthermore, Zhang et al. incorporate a role and attribute-based access control model, which restricts data access to users meeting specific roles and attribute conditions, aligning with the work of (Zhou et al. ,2020) on Multi-Authority CP-ABE systems that support flexible policy enforcement.

I. Identification of Gaps in the Literature Review

- **Scalability of Access Control Policies:** Role-Based Access Control (RBAC), the mainstay of traditional FIM solutions, is challenging to scale in federated systems because it requires frequent policy changes to support additional users, roles, or services across systems. This lack of scalability affects system performance and causes administrative bottlenecks as federations expand. Although some academics suggest Attribute-Based Access Control (ABAC) as a more adaptable option, the computing

complexity necessary to assess intricate attribute-based policies in real time makes large-scale practical implementations difficult (He et al., 2024).

- **Data-Centric Security:** Traditional FIM systems prioritize access authorization and authentication over data protection. In the absence of inherent data-centric security prevention, sensitive data is nonetheless susceptible to failure or circumvention of access constraints. Attribute-Based Encryption (ABE) techniques can help with this, but their successful integration in federated identity system is hindered by performance issues and additional system complexity (Das, 2024).
- **Inadequate Dynamic Access Control:** One key limitation identified in the reviewed literature is predefined and static access control tags, which limits the ability of the system to support dynamic access control mechanisms. In the case of (Niskanen and Salonen ,2023), while their framework effectively enforces fine-grained access based on structured data formats, it lacks provisions for real-time modification of access policies in response to dynamic changes. This poses challenges in situations where access permissions must adapt rapidly. The absence of dynamic policy enforcement mechanisms restricts the framework's applicability in environments that require flexibility and responsiveness to evolving access requirements.
- **End-User Usability:** (Zhang et al., 2023) proposed algorithm significantly enhances security by tackling issues like identity forgery and privacy leakage, but it doesn't fully address the challenges faced by non-technical end-users in terms of usability. It is crucial to develop user-friendly interfaces that simplify the process of key management and identity verification. Users without technical expertise, may struggle with complex encryption key handling or hybrid signature systems.

J. Connection to Current Research Question.

The review of related works revealed that Federated Identity Management (FIM) systems are vulnerable to serious security flaws even though they are necessary for facilitating smooth authentication and access control across many platforms. These include reliance on centralized identity providers, single points of failure, insider threat exposure, and illegal data access, all of which could result in serious breaches if compromised. Addressing these security issues has been a top priority in both the academic and industrial spheres. Attribute-based encryption (ABE) has emerged as a cryptographic technique for enhancing data security. By enabling fine-grained access control based on attributes rather than identities, ABE ensures that only authorized users with specific attributes can access sensitive resources. Despite its potential, the application of ABE in FIM systems remains underexplored. Current research has largely focused on theoretical models or limited use cases, without addressing critical real-world challenges such as scalability, computational overhead, and the dynamic nature of attribute management in federated environments. This research aims to bridge these gaps by exploring how ABE can be integrated into FIM systems to enhance their privacy and security.

CHAPTER THREE METHODOLOGY

➤ Introduction

This chapter outlines the systematic approach employed to achieve the objectives of this research, focusing on enhancing privacy in Federated Identity Management (FIM) through Attribute-Based Encryption (ABE). The research adopts a combination of methodologies, incorporating design and creation strategies within the framework of qualitative study. Data and findings are analyzed qualitatively to provide insights into the effectiveness and feasibility of the proposed model.

The methodology includes the design and construction of technical artifacts that integrate ABE into FIM systems. This process is informed by a detailed case study, which serves as the foundation for qualitative analysis.

(Oates, 2006) Design and Creation Process model will service as guide throughout the program development cycle. This model comprises five key stages: awareness, suggestion, development, evaluation, conclusion. This structured approach ensures a rigorous and iterative process, fostering both practical and theoretical contributions.

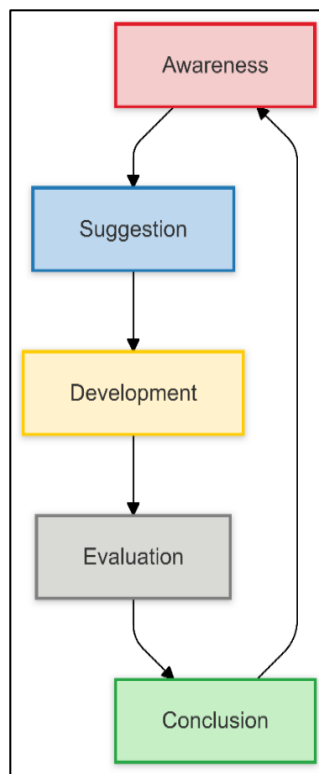


Fig 4 Oates Design and Creates Model

➤ Design and Create Process Model

- **Awareness:** Traditional FIM systems often relies on coarse-grained access controls and centralized trust, making them vulnerable to unauthorized access, privacy breaches, and single points of failure. Existing studies highlight the need for privacy-preserving solutions that address these vulnerabilities without compromising system performance or user experience (Deshmukh, Yadav and Bhandari, 2021). Attribute-Based Encryption (ABE) emerges as a viable approach, providing fine-grained access control by encrypting data based on user attributes.
- **Suggestion:** To address the identified challenges, this research proposes the integration of a customized Ciphertext ABE scheme into FIM systems to enhance privacy and access control. The proposed approach involves designing a flexible encryption policy that supports dynamic attributes while preserving scalability and minimizing computational overhead. By incorporating ABE, the solution aims to prevent unauthorized access, reduce reliance on centralized trust, and enable secure data sharing across federated domains.
- **Development:** The proposed system will be implemented as an artifact, leveraging modern cryptographic libraries for ABE and integrating them into an FIM framework. The development will use PHP and MySQL programming language and associated cryptographic libraries for implementing the encryption and decryption processes.
- **Evaluation:** The effectiveness of the system will be assessed through rigorous evaluation. The prototype will be tested to measure its ability to enhance privacy, enforce attribute-based access control policies, and prevent unauthorized access.

Performance metrics such as computational overhead, scalability, and encryption/decryption times will be analyzed to determine the system's practicality.

- **Conclusion:** After implementing the proposed system and conducting a comprehensive evaluation, conclusions will be drawn regarding the effectiveness of ABE in enhancing privacy within FIM. The findings will be analyzed in the context of previous research to highlight improvements, challenges, and future opportunities.

➤ *Tools, Software, and Resources*

The implementation of the proposed system model of Attribute-Based Encryption (ABE) requires a combination of software tools, libraries, and development environments that facilitate cryptographic operations, secure data handling, and seamless integration with a federated identity system in a cloud environment.

• *Development Tools and Environment:*

- ✓ **PHP (Hypertext Preprocessor):** PHP serves as the core programming language for implementing the ABE algorithms. Its versatility and extensive library support make it ideal for handling encryption, decryption, and system integration. It is a powerful and flexible language that enables developers to create dynamic and interactive web applications.
- ✓ **Integrated Development Environment (IDE):** VS Code is widely used to write, debug, and manage PHP codebases efficiently, offering flexibility and support for modern development practices. Its robust features, extensive library of extensions, and intuitive user interface make it an ideal choice for PHP development.
- ✓ **XAMP (Apache/MySQL):** MySQL is ideal for implementing the proposed system because it efficiently stores and manages user attributes and policies in structured tables. It supports complex queries and indexing for fast retrieval of attributes, enabling dynamic access control. The databases ensure secure storage and retrieval of sensitive data with encryption and access control features.
- ✓ **PHP OpenSSL Extension:** The PHP OpenSSL Extension allows for encryption, decryption, digital signatures, and secure communication using the OpenSSL library. It supports a wide range of cryptographic algorithms, including RSA, AES, and SHA, which are required for developing secure systems such as Attribute-Based Encryption (ABE).

➤ *Threat Model.*

A healthcare organisation, AgesCare, stores and processes sensitive patient data in a cloud environment. The organization must comply with stringent regulations, such as the General Data Protection Regulation Act (GDPR), which requires ensuring that only authorized personnel have access to specific data based on their roles, responsibilities and location. AgesCare faces challenges in implementing fine-grained access control while maintaining scalability and efficiency.

• *Potential System Adversaries.*

- ✓ **External Adversaries:** Unauthorized entities such as hackers may try to obtain private information or interfere with system functions. These adversaries may take advantage of weaknesses in the system's encryption, access control, or communication channel. Their goals frequently involve stealing confidential data, altering data, or interfering with the system's functionality (Si-Ahmed, Ali Al-Garadi and Boustia, 2023).
- ✓ **Insiders, or internal adversaries:** Employees, administrators, and other entities having insider access may purposefully change policies to allow illegal access or disclosing private information to outside parties (Nabil et al., 2019). They might give unauthorized users access to their decryption keys or credentials, allowing them to get beyond the system's access control measures. Adversaries possess in-depth knowledge of the system's architecture and security protocols, they can more easily take advantage of internal vulnerabilities, which makes the insider threat deadlier (Si-Ahmed, Ali Al-Garadi and Boustia, 2023).

• *Adversary Capabilities.*

- ✓ **Eavesdropping on Communications:** Adversaries may intercept data transmitted between nodes, including ciphertexts, public parameters, and access policy updates. By capturing this data, they aim to gather information about the system's operations, user attributes, or even attempt to decrypt sensitive information through brute force or pattern analysis. Without secure communication channels, such as those based on TLS, this vulnerability becomes a critical point of failure (Garnaev and Trappe, 2022).
- ✓ **Compromising User Credentials or Attributes:** Adversaries may steal user credentials, private keys, or attribute information through phishing, malware, or exploiting poorly secured devices. Once credentials are compromised, attackers can impersonate legitimate users, bypass access policies, and access sensitive data (Crane, 2023).
- ✓ **Exploiting Weaknesses in the ABE Scheme:** Adversaries may target specific vulnerabilities in the attribute-based encryption framework. Examples include exploiting outdated attributes that have not been revoked or updated, leading to unauthorized access even after a user's privileges should have been revoked. Key leakage, another critical vulnerability, can occur through side-channel attacks, poorly implemented cryptographic operations, or insider compromises, allowing attackers to decrypt data without satisfying access policies (Prantl et al., 2023).

• Security Model.

Attribute-Based Encryption (ABE) chosen to address the AgesCare challenge. In this cryptographic technique, access rights are determined based on user characteristics rather than specific identities. For example, attributes such as "Role: Nurse" and "Location: London" can be used to define access policies. The network process diagram Figure 5, demonstrates the security model designed to mitigate the identified threats.

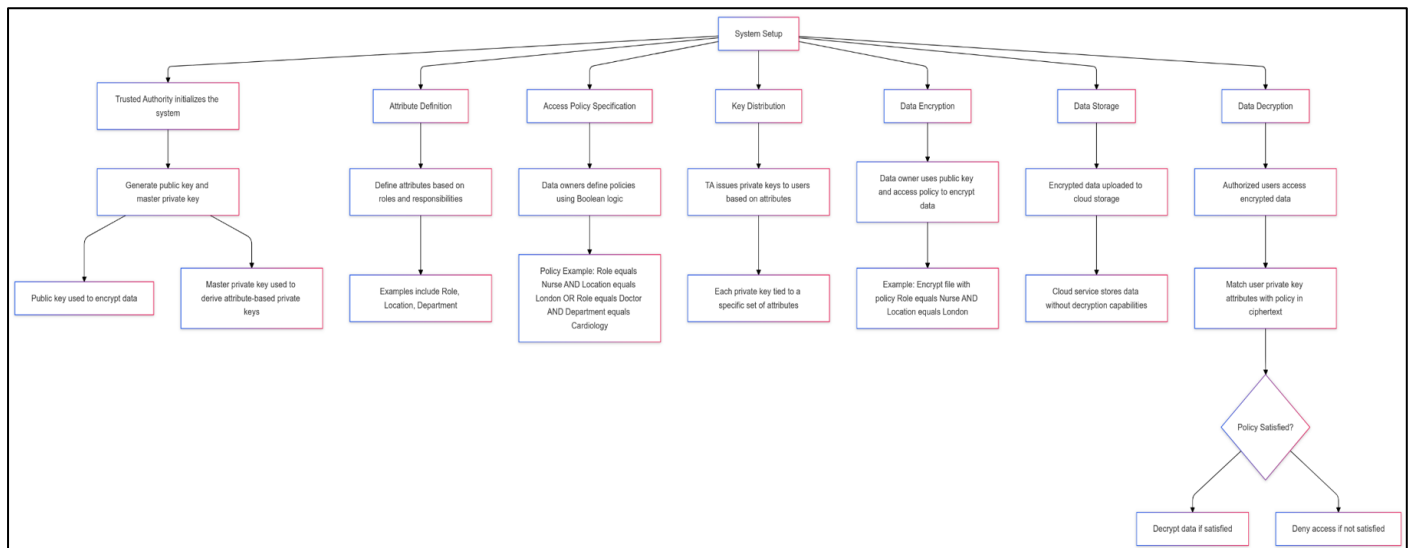


Fig 5 Threat Model Network Diagram.

• Proposed Security Model

✓ System Setup.

A Trusted Authority (TA) initializes the system by generating a public key and a master private key. The public key is used to encrypt data, while the master private key is used to derive attribute-based private keys for authorized users.

✓ Define Attribute.

Define attributes based on organizational roles and responsibilities.

- Role: Nurse, Doctor, Administrator
- Location: London, New York
- Department: Cardiology, Oncology

✓ Access Policy Specification.

Data owners define access policies using a Boolean logic structure to specify who can access data. Example Policy: (Role: Nurse AND Location: London) OR (Role: Doctor AND Department: Cardiology)

✓ Key Distribution.

The TA issues private keys to users based on their attributes. Each private key is tied to a specific set of attributes, ensuring access control is tightly enforced.

✓ Data Encryption:

When uploading data, the data owner uses the public key and defines an access policy to encrypt the data using CP-ABE.

- Example: A file containing sensitive cardiology patient data is encrypted with the policy (Role: Nurse AND Location: London).

✓ Data Storage:

Encrypted data is uploaded to the cloud storage. The cloud service provider is responsible only for storing encrypted data and has no capability to decrypt it, ensuring GDPR compliance.

✓ Data Decryption:

- Authorized users attempt to access data.
- Their private keys are matched against the policy embedded in the ciphertext.
- If their attributes satisfy the policy, the ciphertext is decrypted; otherwise, access is denied.

➤ *Ethical Considerations.*

Conducting research on the topic raises several ethical considerations that must be considered to ensure that the research is conducted responsibly, particularly given its focus on privacy, data security, and encryption technologies.

- Ethical clearance for the research was sought and obtained from Teesside University Ethics Committee. After thorough review and consideration of the research methodology, approval was granted, allowing the study to commence in adherence to ethical standards. The study adheres to the approved ethical guidelines, ensuring compliance with all relevant standards.
- The project complies with all relevant data protection laws and standards, such as:
 - ✓ GDPR (General Data Protection Regulation) in the European Union, which emphasizes user rights regarding data privacy, including data encryption and protection.
 - ✓ ISO/IEC 27001 standards for information security management systems, which provide a framework for implementing encryption in ways that ensure data security and compliance with best practices.

CHAPTER FOUR IMPLEMENTATION

A. Introduction

This chapter delves into the practical implementation of the proposed system model to enhance privacy and improves security in federated identity management systems using Attribute-Based Encryption (ABE) in a cloud computing. The implementation step connects the theoretical concepts introduced in earlier chapters to their real-world applications. It describes the technological architecture, tools, algorithms, and procedures utilized to implement the proposed system.

```
pseudocode.php
1
2
3 class ABE:
4     def __init__(self):
5         self.PP, self.MSK = self.setup()
6
7     def setup(self):
8         # Generate system parameters and master secret key
9         PP = generate_public_parameters()
10        MSK = generate_master_secret_key()
11        return PP, MSK
12
13    def key_generation(self, attributes):
14        # Generate private key for a user based on their attributes
15        SK_user = derive_private_key(attributes, self.MSK, self.PP)
16        return SK_user
17
18    def encrypt(self, message, access_policy):
19        # Encrypt a message with a given access policy
20        CT = encrypt_message(message, access_policy, self.PP)
21        return CT
22
23    def decrypt(self, ciphertext, user_attributes, SK_user):
24        # Check if user attributes satisfy the access policy in the ciphertext
25        if satisfies_policy(user_attributes, ciphertext.policy):
26            decrypted_message = decrypt_ciphertext(ciphertext, SK_user, self.PP)
27            return decrypted_message
28        else:
29            raise Exception("Decryption failed: Attributes do not satisfy the policy.")
30
```

Fig 6 Pseudocode to Implement the Proposed Algorithm

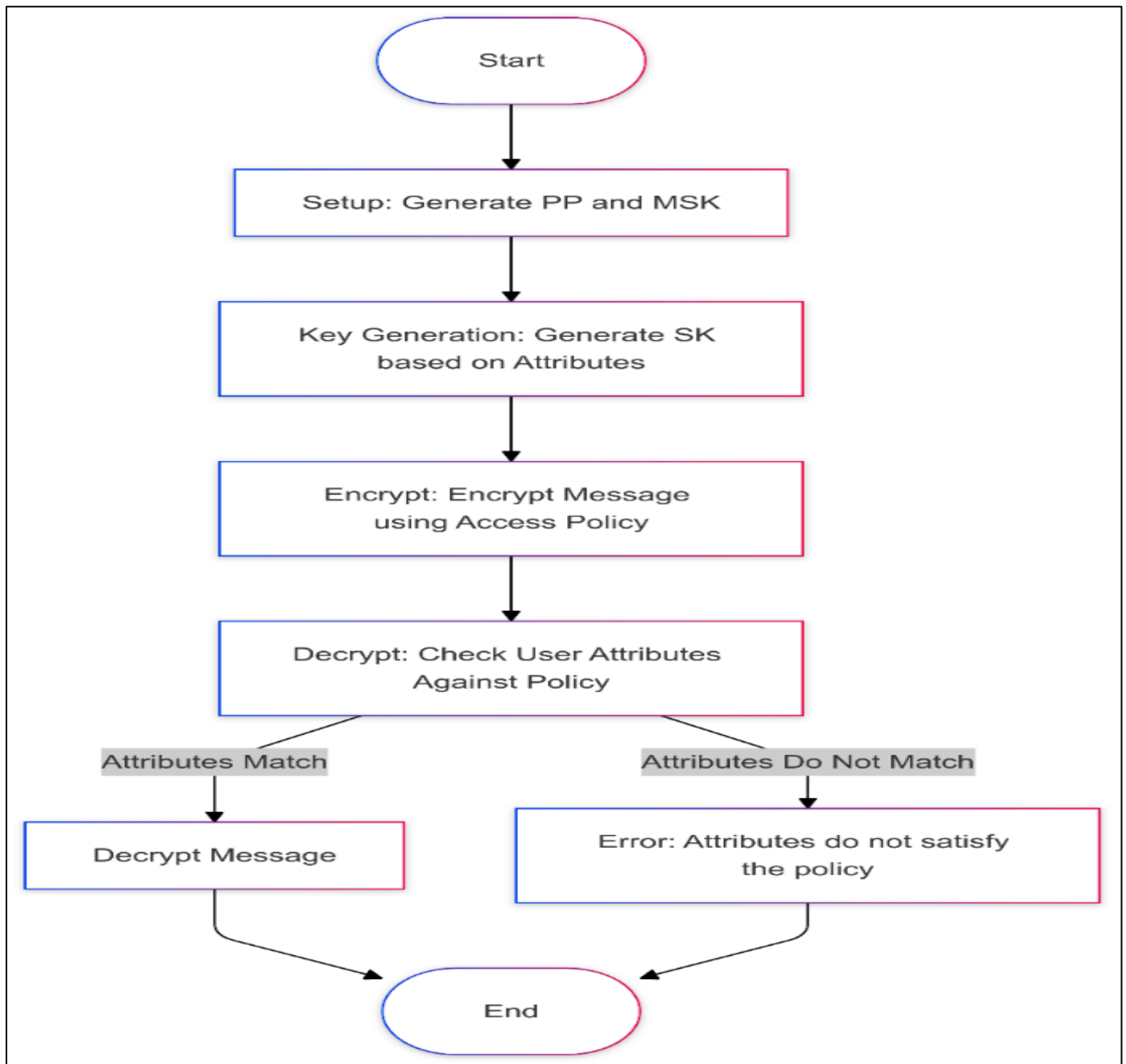


Fig 7 System Implementation Flowchart

B. Algorithm for Implementation of the Proposed System

➤ System Setup:

The system initializes by generating the global public key ('P') and a master secret key ('MSK'). The public keys are shared with users with matching attributes and are used for encryption and decryption of data. These keys establish a secure cryptographic framework. The master secret key is kept private and is critical for deriving user-specific private keys.

- **Input:** A security parameter λ .
- **Output:** Master secret key MSK and public key PK.

• Algorithm

- ✓ Choose a bilinear group G with two slots.
- ✓ Input two numbers e.g. (A, B) .
- ✓ Output a unique value $e(A, B)$ (bilinear map).
- ✓ Select a random generator $g \in G$. A master number g (generator) to create keys.

- ✓ Choose a hash function $H: \{0,1\}^* \rightarrow G$ for attributes. The (hash function
- ✓ H) converts words (e.g., "Doctor") into unique arbitrary values.
- ✓ Generate two keys:
- ✓ Public Key (PK): $(g, g^a, e(g, g))$. Where:
- ✓ g (master number).
- ✓ g^a (master number raised to a secret power a).
- ✓ $e(g, g)$ (a precomputed value).
- ✓ Master Secret Key (MSK): Hidden. Contains the secret power a .

➤ *Key Generation:*

The user's attributes are input into the system. Based on these attributes, the system uses the Master Secret Key ('MSK') to generate a unique private key ('SK') for the user. This private key is tied to the user's attributes, ensuring that it can only be used by the specific user it was issued to. This step ensures that only authorized users can attempt decryption based on their attributes.

- **Input:** Master secret key MSK, user attributes $A = \{a_1, a_2, \dots, a_n\}$
- **Output:** User's private key SK.

• *Algorithm:*

- ✓ For each attribute $a_i \in A$ in the Set A
- ✓ Compute $H(a_i)$ using the hash function. Hashes attributes (e.g., "Doctor") into a unique number in a cryptographic group G .
- ✓ Generate the private key component $K_i = H(a_i)^a$.
- ✓ a : A secret number (part of the system's master key). K_i : A "digital badge" for the attribute a_i .
- ✓ Collect all badges K_1, K_2, \dots, K_n into a single private key SK. Example: $SK = \{K_{\text{Doctor}}, K_{\text{London}}\}$.
- ✓ Combine all K_i values to form the user's private key $SK = \{K_i\}_{i=1}^n$

➤ *Encryption:*

The data owner specifies the message to be encrypted and defines an access policy. The access policy outlines the attributes or combination of attributes required to decrypt the message. The system encrypts the message using the public key ('P') and the access policy. The resulting ciphertext ('CT') is secured such that only users with matching attributes can decrypt it.

- **Input:** Public key PK, message M, and access policy P.
- **Output:** Ciphertext CT.

• *Algorithm:*

- ✓ Encode the access policy p as a monotonic Boolean formula (AND, OR).
- ✓ Randomly select $s \in \mathbb{Z}_p$ as the secret. Where \mathbb{Z}_p is integers modulo a prime p .
- ✓ Compute $C = M \cdot e(g, g)^{as}$ to bind the message.
- ✓ For each attribute a_i in the policy:
- ✓ Compute $C_i = g^{s \cdot w_i}$. where w_i is a weight derived from the policy.
- ✓ Compute $D_i = g^{w_i}$. Used to reconstruct the secret during decryption.
- ✓ The ciphertext is $CT = (C, \{C_i, D_i\}_{i \in P})$. C : Encrypted message bound to $e(g, g)^{as}$.

➤ *Decryption:*

Data user provides the ciphertext, their attributes, and their private key ('SK'). The system evaluates the user's attributes against the access policy embedded in the ciphertext. If the attributes satisfy the policy, the system uses the private key to decrypt the message. The user then gains access to the original content. If the attributes do not satisfy the policy, the decryption fails, and the user cannot access the message.

- **Input:** Ciphertext CT, user private key SK, and attributes A.
- **Output:** Decrypted message M or failure.

• *Steps:*

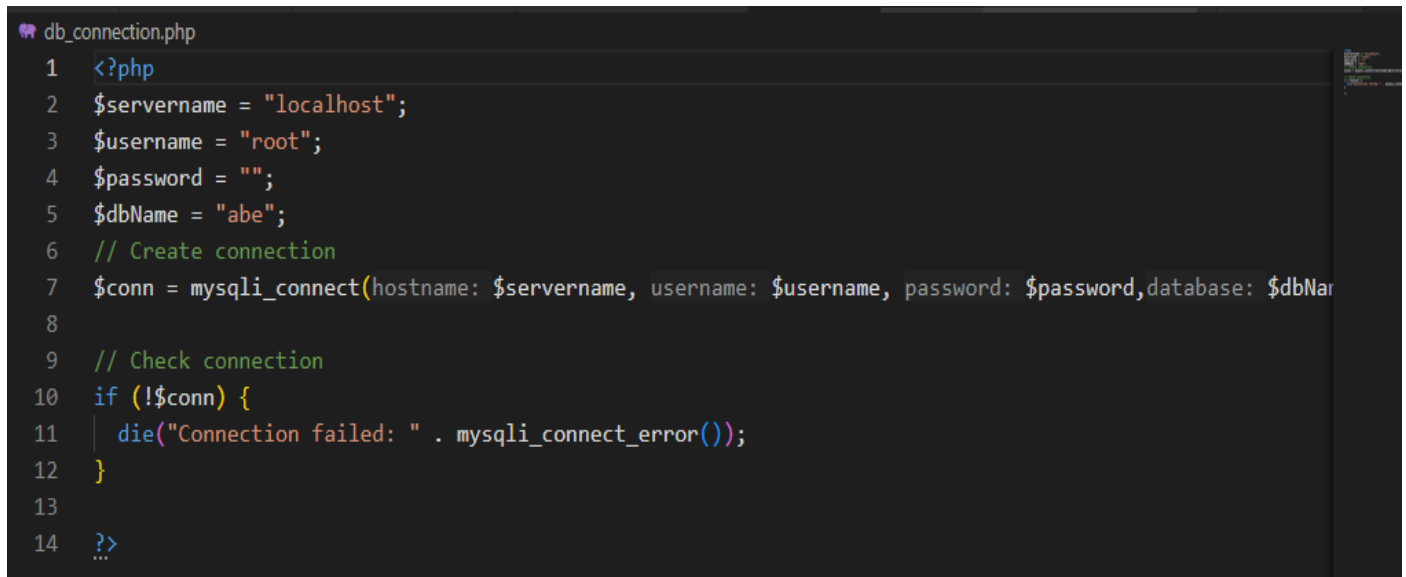
- ✓ Verify that the user's attributes A satisfy the access policy P.
- ✓ If satisfied, reconstruct the secret s using the attribute keys $\{K_i\}$ and policy weights $\{w_i\}$.

$$\text{Compute } M = \frac{C}{e(g, g)^{as}}$$

C. Program Implementation.

➤ Database Setup.

The database connection is important in implementing the proposed system. it serves as the backbone for securely storing and retrieving data required for access control and encryption. It allows the application to store and access user-related data, such as keys, encryption policies, and attributes. The system model program will dynamically query attributes to check user rights against encryption policies by establishing a dependable connection to the database. This ensures that only authorized users with the appropriate attributes are able to decrypt data.



```

db_connection.php
1  <?php
2  $servername = "localhost";
3  $username = "root";
4  $password = "";
5  $dbName = "abe";
6  // Create connection
7  $conn = mysqli_connect(hostname: $servername, username: $username, password: $password,database: $dbName);
8
9  // Check connection
10 if (!$conn) {
11     die("Connection failed: " . mysqli_connect_error());
12 }
13
14 ?>

```

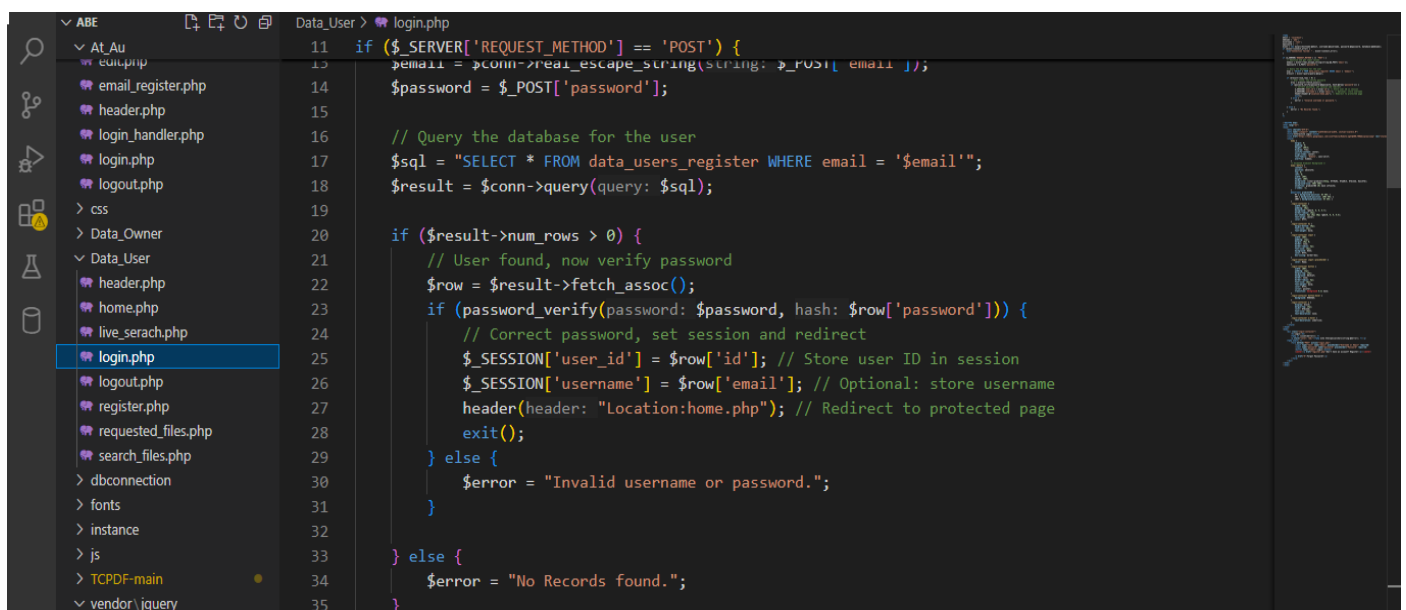
Fig 8 Connection to MySQL Database

➤ Component of the Proposed System Model.

The User Dashboard serves as the primary interface for controlling access to information and resources within the system. The purpose of this dashboard is to give data owners and users a safe, user-friendly environment in which to access their required resources.

• User Login.

To gain access to the system, users must authenticate themselves by login into the system. Upon successful login to the systems, users are granted access to resources according to the attributes allocated by the Attribute Provider. After authentication, the ABE system ensures that access to particular files or data is limited unless the user's attributes meet the policies established by the Data Owner or Attribute Provider. Users without login credentials need to register to be granted access to the system.



```

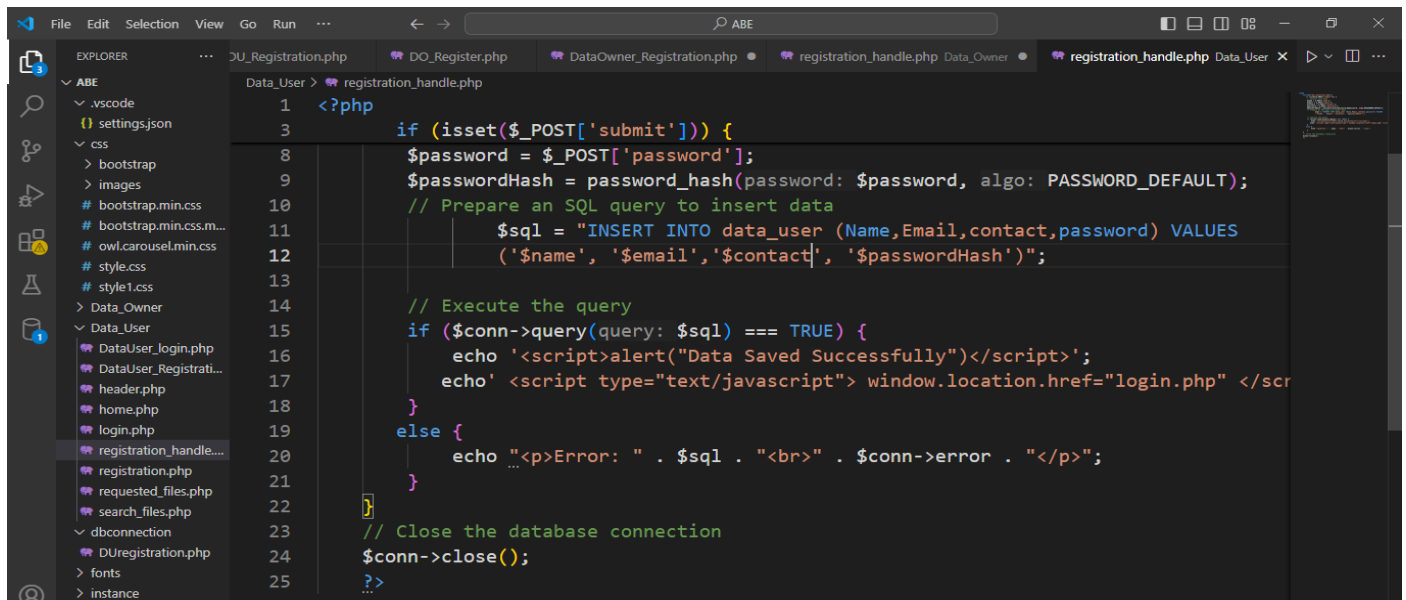
Data_User > login.php
11 if ($_SERVER['REQUEST_METHOD'] == 'POST') {
12     $email = mysqli_real_escape_string($conn, $_POST['email']);
13     $password = $_POST['password'];
14
15     // Query the database for the user
16     $sql = "SELECT * FROM data_users_register WHERE email = '$email'";
17     $result = $conn->query($sql);
18
19     if ($result->num_rows > 0) {
20         // User found, now verify password
21         $row = $result->fetch_assoc();
22         if (password_verify($password, $row['password'])) {
23             // Correct password, set session and redirect
24             $_SESSION['user_id'] = $row['id']; // Store user ID in session
25             $_SESSION['username'] = $row['email']; // Optional: store username
26             header("Location:home.php"); // Redirect to protected page
27             exit();
28         } else {
29             $error = "Invalid username or password.";
30         }
31     } else {
32         $error = "No Records found.";
33     }
34 }
35

```

Fig 9 Data User Login Backend

- **Registration Page:**

Users must register and provide the needed identifying information before they can access any files. After registering, access is subject to the Data Owner's consent and the Attribute Provider's assignment of attributes. These characteristics specify the Data User's function and system-wide permissions. Resource access is not given at random; rather, it depends on whether the attributes that the Data User has been assigned meet the access rules that the Data Owner has set.



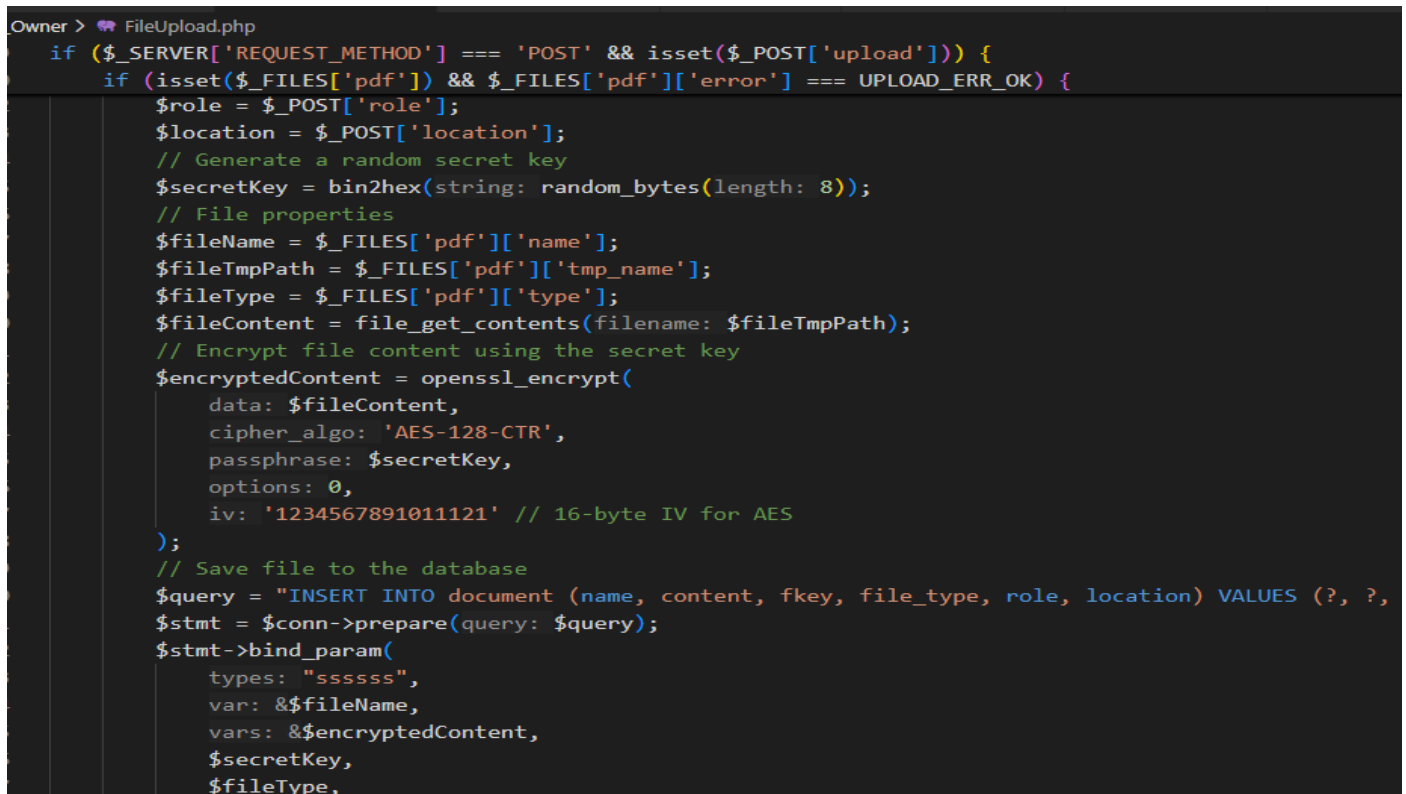
```

1  <?php
3  if (isset($_POST['submit'])) {
8      $password = $_POST['password'];
9      $passwordHash = password_hash($password, PASSWORD_DEFAULT);
10     // Prepare an SQL query to insert data
11     $sql = "INSERT INTO data_user (Name,Email,contact,password) VALUES
12     ('$name', '$email', '$contact', '$passwordHash')";
13
14     // Execute the query
15     if ($conn->query($sql) === TRUE) {
16         echo '<script>alert("Data Saved Successfully")</script>';
17         echo '<script type="text/javascript"> window.location.href="login.php" </scr
18     }
19     else {
20         echo "<p>Error: " . $sql . "<br>" . $conn->error . "</p>";
21     }
22 }
23 // Close the database connection
24 $conn->close();
25 ?>
  
```

Fig 10 Backend Data User Registration

- **File Upload:**

Figure 10 illustrates the implementation of the proposed system to ensure secure file storage and controlled access. By associating specific attributes like role and location when uploading file, the system integrates ABE principles where attributes govern access rights. The file is encrypted using a symmetric encryption algorithm (AES-128-CTR) and protected with a unique secret key generated for each file upload. The key serves as important security component, allowing only authorized users who possess the correct key to decrypt and access the file.



```

Owner > FileUpload.php
if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['upload'])) {
    if (isset($_FILES['pdf']) && $_FILES['pdf']['error'] === UPLOAD_ERR_OK) {
        $role = $_POST['role'];
        $location = $_POST['location'];
        // Generate a random secret key
        $secretKey = bin2hex(random_bytes(8));
        // File properties
        $fileName = $_FILES['pdf']['name'];
        $fileTmpPath = $_FILES['pdf']['tmp_name'];
        $fileType = $_FILES['pdf']['type'];
        $fileContent = file_get_contents($fileTmpPath);
        // Encrypt file content using the secret key
        $encryptedContent = openssl_encrypt(
            data: $fileContent,
            cipher_algo: 'AES-128-CTR',
            passphrase: $secretKey,
            options: 0,
            iv: '1234567891011121' // 16-byte IV for AES
        );
        // Save file to the database
        $query = "INSERT INTO document (name, content, fkey, file_type, role, location) VALUES (?, ?, ?, ?, ?, ?)";
        $stmt = $conn->prepare($query);
        $stmt->bind_param(
            types: "ssssss",
            var: &$fileName,
            vars: &$encryptedContent,
            $secretKey,
            $fileType,
  
```

Fig 11 Encrypted File Upload with Access Policy

- *Query user Data Based on Attributes.*

This component implements Searchable Encryption which a critical component of an Attribute based encryption system. By enforcing access based on session variables such as `Location` and `Role`, the system ensures that users can only view records that align with their attributes as illustrated in figure 11. This approach aligns with ABE's core concept, where access policies are defined by user attributes rather than individual identity. By associating each document with specific attributes and validating these against user sessions, the system implements a policy-driven access mechanism. This ensures confidentiality by restricting access to sensitive records. The session management protect against unauthorized access, ensuring only authenticated users with the correct attributes can interact with the data.

```
// Check and retrieve session variables
~ if (!isset($_SESSION['Location']) || !isset($_SESSION['Role'])) {
    die("Session variables for location or role are not set.");
}
$location = $_SESSION['Location'];
$role = $_SESSION['Role'];
$email = $_SESSION['Email'];

// Fetch matching records
$searchQuery = "SELECT * FROM document WHERE role = ? AND location = ?";
$stmt = $conn->prepare(query: $searchQuery);
$stmt->bind_param(types: "ss", var: &$role, vars: &$location);

~ if (!$stmt->execute()) {
    die("Query failed: " . $stmt->error);
}

$result = $stmt->get_result();
```

Fig 12 Retrieving user Data Based on Attributes

- *Dynamic Attributes Update.*

The fundamental idea behind ABE is that access to encrypted data is granted or denied based on attributes, rather than traditional key management system. As demonstrated in figure 12, the system checks whether the attributes associated with a user meet the necessary requirements outlined in the resource's access policy. The policy determines the conditions under which a user can access the resource, typically based on attributes. By dynamically updating, the policy the system ensures that access control remains flexible and easily modifiable. If at any time the user's attributes change, their access can automatically be updated without needing to issue new keys or re-encrypt data.

```
$policyQuery = "SELECT access_policy FROM resource_policies WHERE resource_id = ?";
$policyStmt = $conn->prepare(query: $policyQuery);
$policyStmt->bind_param(types: "i", var: &$resourceId);
$policyStmt->execute();
$policyResult = $policyStmt->get_result();

if ($policyResult->num_rows === 0) {
    return ["status" => "revoked", "message" => "Resource policy not found."];
}
$accessPolicy = $policyResult->fetch_assoc()['access_policy'];

// Evaluate access policy
$policyParts = explode(separator: " AND ", string: $accessPolicy);
foreach ($policyParts as $part) {
    [$key, $value] = explode(separator: "=", string: $part);
    $key = trim(string: $key);
    $value = trim(string: $value);
    if (!isset($userAttributes[$key]) || $userAttributes[$key] !== $value) {
        return ["status" => "revoked", "message" => "Access policy not satisfied."];
    }
}
```

Fig 13 Dynamic Policy Update

CHAPTER FIVE

RESULTS AND DISCUSSIONS

A. Introduction

To implement the proposed system to enhance privacy and security in Federated Identity Management System, some important software tools and resources must be installed and configured. These tools will provide the necessary environment for developing, testing, and deploying the system.

B. Install Required Software.

XAMPP will be used to locally host the systems files and resources. XAMPP is preferred because it offers a cross-platform, free, and open-source package that makes setting up a local web server environment easier. It's also provides an integrated environment with MySQL, PHP, and Apache web server pre-configured for local development.

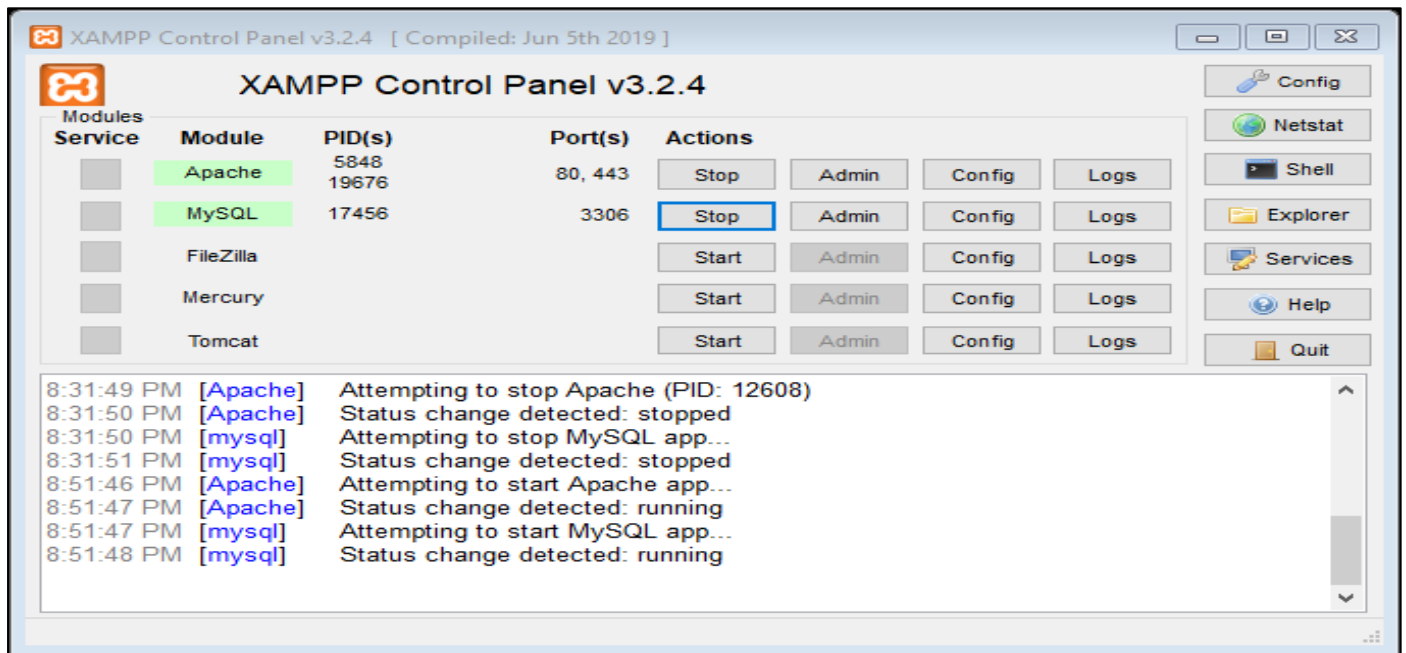


Fig 14 XAMP to Host Resources Locally

The Apache and MYSQL modules must be started before localhost server can be accessed. The Apache functions as web server that processes client requests and delivers web pages or backend response and MySQL offers an organized method for storing attributes, access logs, encryption policies, and user data. MySQL will be configured to runs on the default port **3306**.

C. Setup Database Connection to MySQL.

- Open a web browser and type `http://localhost` to access the localhost dashboard.

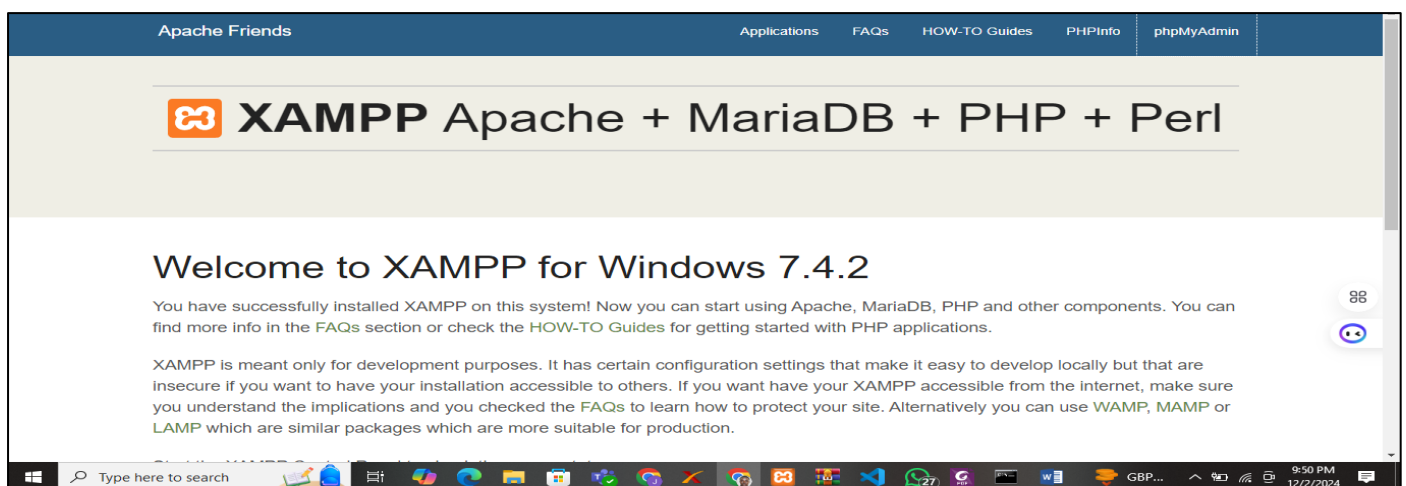


Fig 15 XAMP Interface

- Click on phpMyAdmin to configure the database table necessary for the system setup.



Fig 16 MySQL Database Interface

D. Access the Application:

- To access the application, ensure that local server (XAMPP) is running and the system files are stored in the 'htdocs' folder of XAMPP.

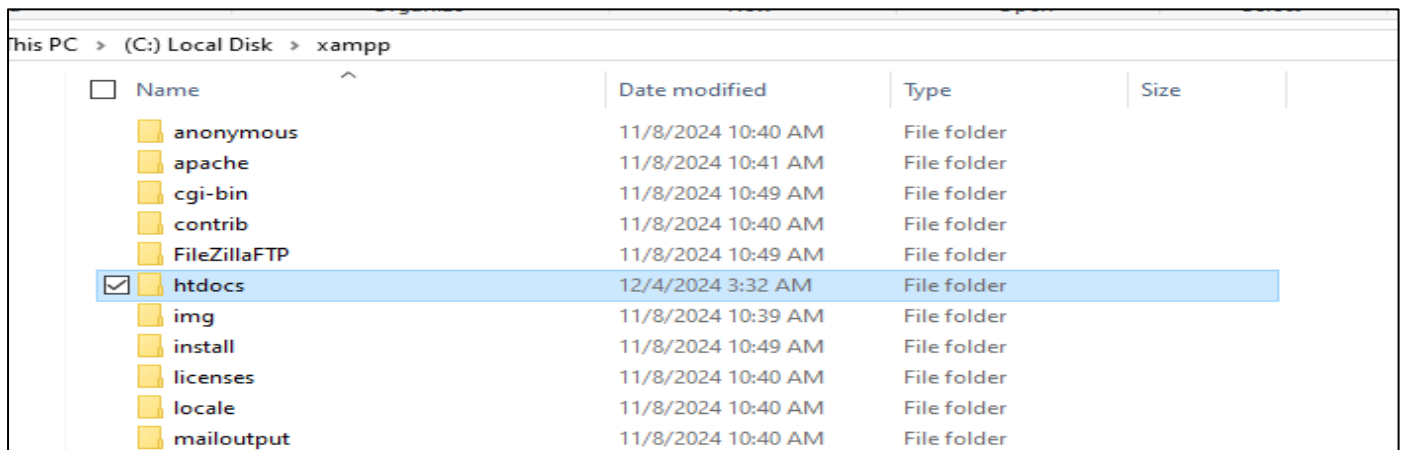


Fig 17 Application Folder -HTDOCS

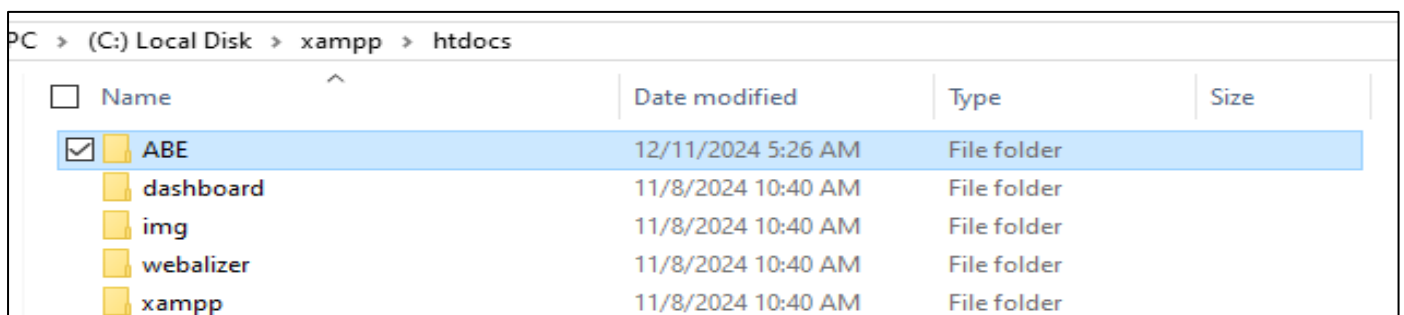


Fig 18 Application folder - ABE Folder

- Then proceed to enter the path to the application's folder (<http://localhost/ABE>) to display the system interface.

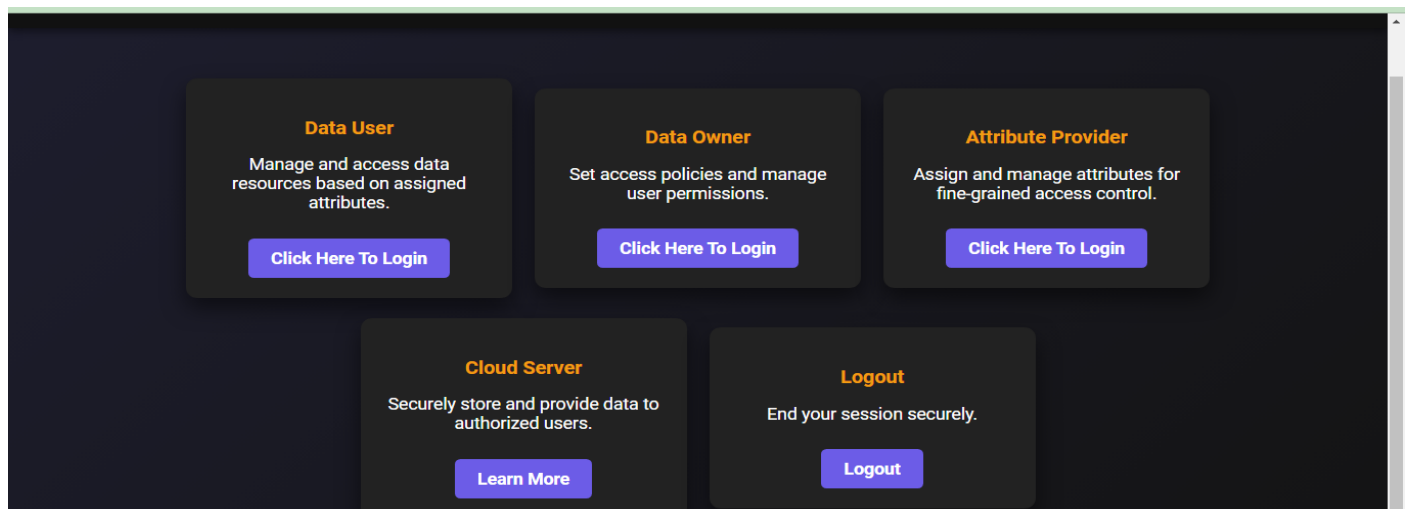


Fig 19 System Interface

- *Step 1- User Registration:*

To access resources on the system approved users by the Attribute Provider must first register to gain access to system. Attribute provider must accept the data user by entering their email into the approved to register database.

The registration form is centered on a bright pink background. It features a dark grey box with the title 'Register' in white. Below the title is a red error message: 'Email already exists. Please use a different email.' The form contains five input fields with placeholder text: 'Enter your name', 'Enter your email', 'Enter your Phone Number', 'Create a password', and 'Confirm password'. At the bottom of the form is a green 'Register' button and a link that says 'Already have an account? Login in Here'.

Fig 20 User Registration

- Users cannot register until their request has been approved by the Attribute Provider.

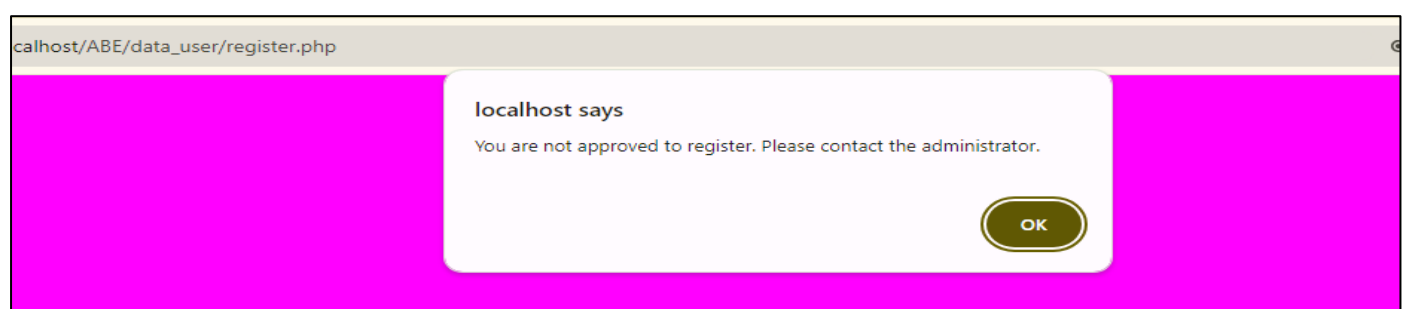


Fig 21 User not approved to register to the system

- *Step 2- User Login*

After successful registration users can now proceed to login by providing a matching credentials used during registration. If login credentials supplied is not found in the system, access to the system files and resources will be denied.

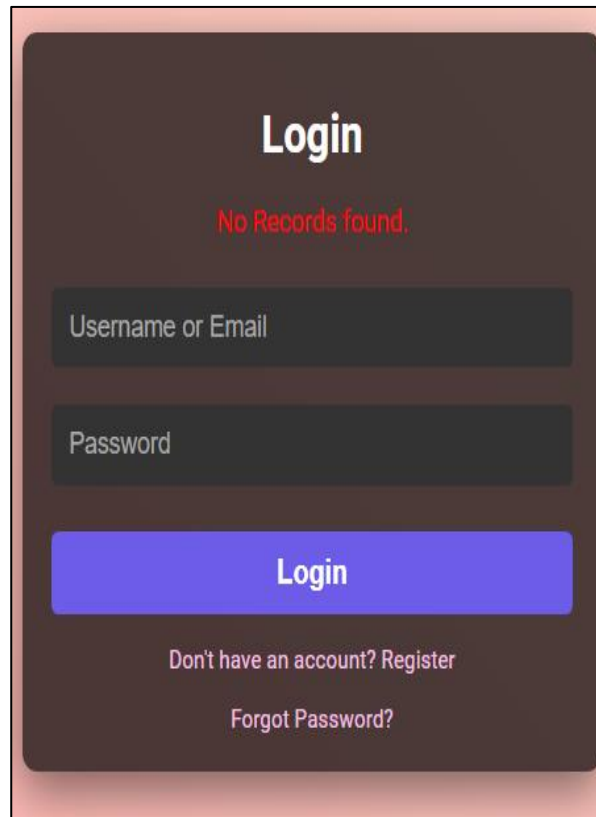


Fig 22 User Login



Fig 23 Data User Dashboard

- *Step 3- File Search.*

After successfully logging in, users will have access to a search feature that allow them to locate files based on their assigned attributes. File must be request for and approved before it can be viewed by users. This search feature is governed by the user's assigned access policies, ensuring that each user can only view and retrieve files they are authorized to access.

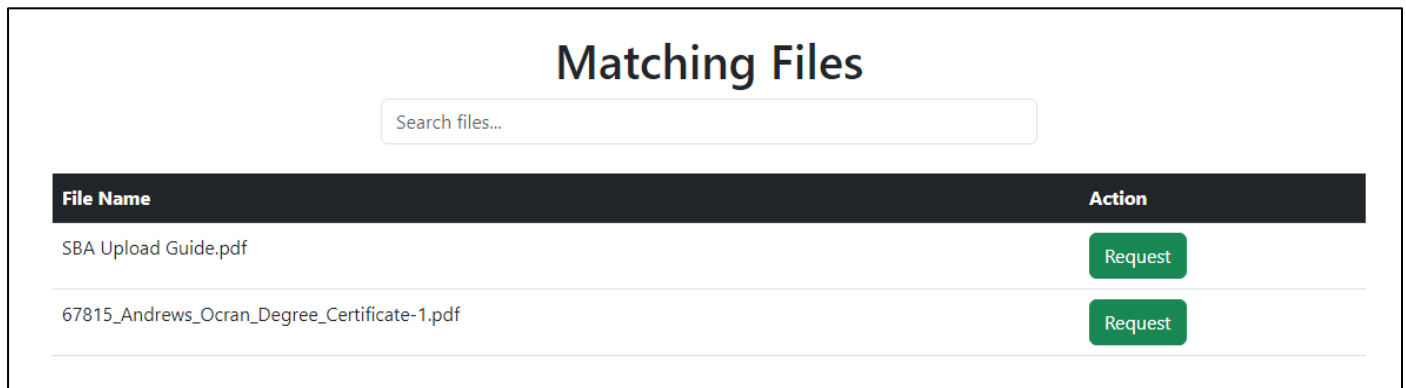


Fig 24 Data user File Search

• *Step 2 - View files.*

Requested and approved files can be viewed under the requested file page. Along with access to these files, they receive the corresponding decryption keys. To access the file contents, data users must enter the provided key, which is used to decrypt the encrypted data securely. This ensures that even if the file is accessed, its contents remain protected and unreadable without the correct decryption key, maintaining confidentiality and enforcing strict access control.

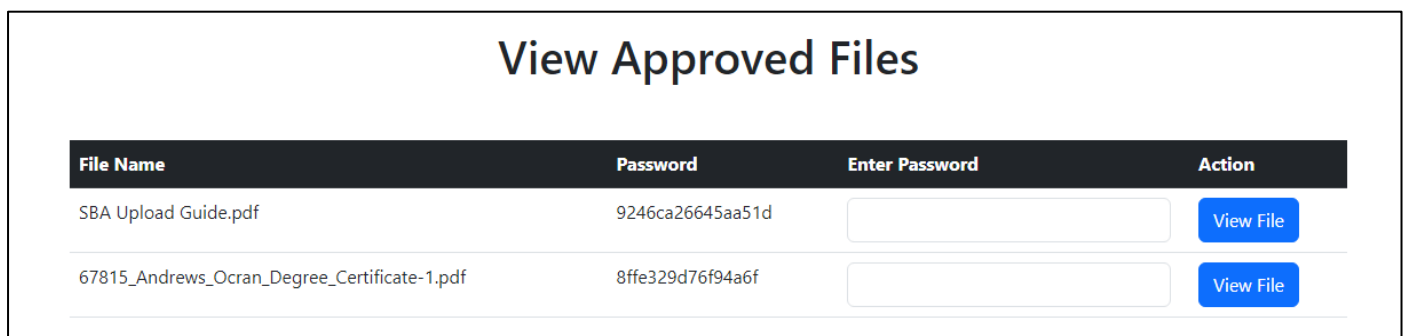


Fig 25 Approved Files

➤ *Data Owner.*

Upon successful login, the data owner can perform the following functions on the system.

- Upload file.
- Approve file
- Trace file
- View Attributes/Policy
- Add Attributes/Policy

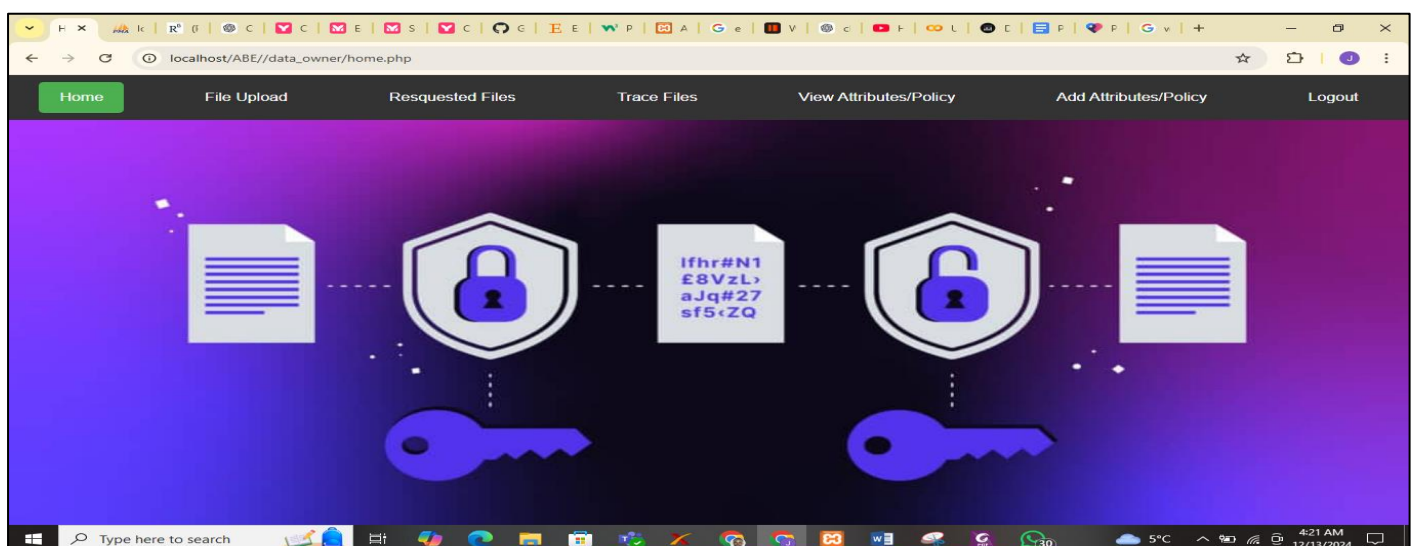


Fig 26 Data Owner Homepage

- *File Upload:*

The data owner can upload files and set access policy that data users must satisfy before access is granted to the file. Users whose attributes match the assigned attributes will have the cryptographic keys necessary to decrypt and access the file. This ensures fine grained access control.

Fig 27 File Upload

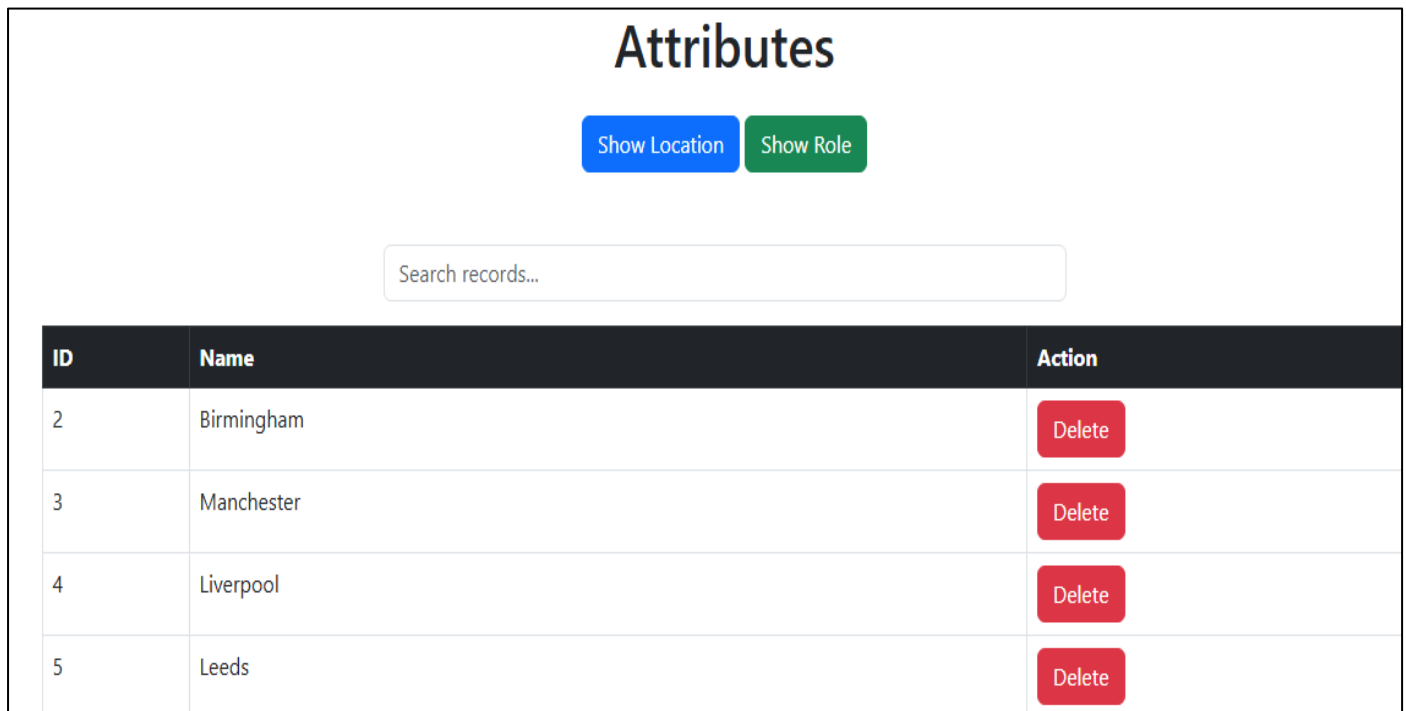
- *Approve File Request:*

This functionality ensures that the data owner retains full control over sensitive files, granting access only to authorized users who meet specific criteria as shown in figure. By reviewing and deciding on requests, the data owner can enforce security policies effectively, ensuring that files are shared only with appropriate users while maintaining data confidentiality and integrity.

Approve Requests		
File Name	User Email	Action
SBA Upload Guide.pdf	kin@gmail.com	<button>Approve</button>
SBA Upload Guide.pdf	kin@gmail.com	<button>Approve</button>
67815_Andrews_Ocran_Degree_Certificate-1.pdf	kin@gmail.com	<button>Approve</button>
SBA Upload Guide.pdf	kin@gmail.com	<button>Approve</button>
67815_Andrews_Ocran_Degree_Certificate-1.pdf	kin@gmail.com	<button>Approve</button>
67815_Andrews_Ocran_Degree_Certificate-1.pdf	kin@gmail.com	<button>Approve</button>

Fig 28 Data Owner File Request Approval

- **View Attributes/Policy:** Under this functionality, the data owner has the option to add, view and modify Attributes.



ID	Name	Action
2	Birmingham	Delete
3	Manchester	Delete
4	Liverpool	Delete
5	Leeds	Delete

Fig 29 Data Owner view Attribute

- *Attribute Authority.*

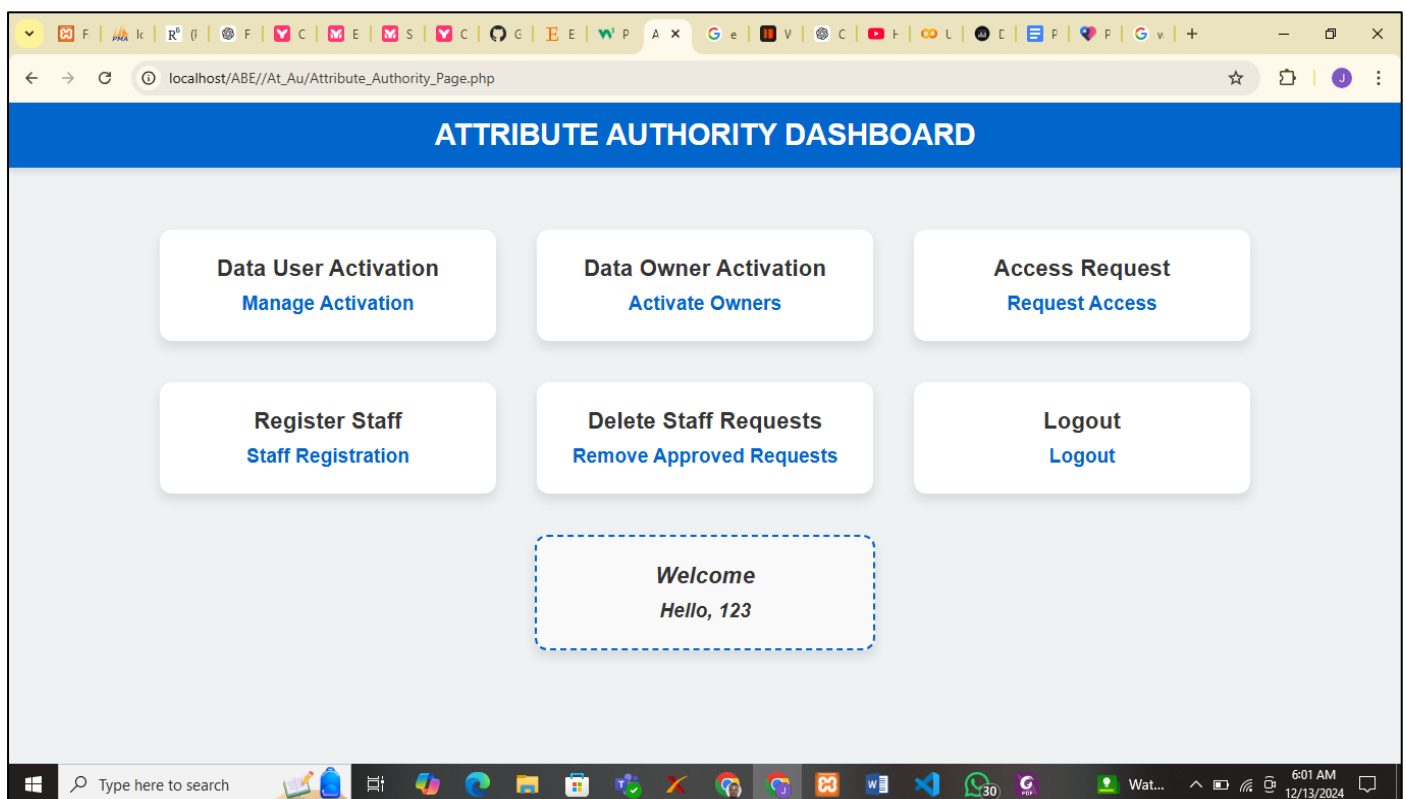


Fig 30 Attribute Authority Dashboard

- The Attribute Provider ensures that data user and data owner activations are managed correctly.
- It facilitates secure access by managing access requests and ensuring only authorized individuals receive appropriate access.

ID	NAME	EMAIL	STATUS	ROLE	LOCATION	ACTION
1	Andrews Ocran	andrews.ocran23@gmail.com	Approved	Pending	Pending	<button>Update Status</button>
2	James Aiden	gem@gmail.com	Approved	Nurse	New York	<button>Update Status</button>
4	Andrews Ocran	andrewsocran32@gmail.com	Approved	Pending	Pending	<button>Update Status</button>
6	kin	kin@gmail.com	Approved	Pediatricians	London	<button>Update Status</button>

Fig 31 Attribute Provider Approval of Registration

- *The Cloud.*

The cloud is the centralized storage for encrypted data. The ciphertext is uploaded by the data owner and later retrieved by authorized users. The cloud storage service does not have access to the decryption keys, ensuring that the service provider cannot access the underlying plaintext data as illustrated in figure 29 below. The data owner has the ability to dynamically update attributes associated with updated files.

Manage Files

ID	Name	Role	Location	Actions
3	Cisco Networking Academy.pdf	Nurse	New York	<button>Delete</button> <button>Edit</button>
4	Cisco Networking Academy 2.pdf	Manager	Tokyo	<button>Delete</button> <button>Edit</button>
5	Florence Academy.pdf	Nurse	New York	<button>Delete</button> <button>Edit</button>
6	ISAAC CV CARE.pdf	Nurse	New York	<button>Delete</button> <button>Edit</button>
9	SBA Upload Guide.pdf	Pediatricians	London	<button>Delete</button> <button>Edit</button>
10	67815_Andrews_Ocran_Degree_Certificate-1.pdf	Pediatricians	Birmingham	<button>Delete</button> <button>Edit</button>

Fig 32 Cloud Storage

E. Discussion

The main objective of the research was to design and implement an Attribute-Based Encryption System to enhance security in a federated identity management system. The proposed system demonstrates several advanced features that effectively address critical gaps in existing review systems.

➤ Research Objective 1.

- *Privacy Risks and Challenges Mitigated the Proposed System*

One of the most significant privacy risks mitigated by the proposed model is Unauthorized Access. In traditional encryption models, unauthorized access can occur if a user gains control over encryption keys or if access policies are too broadly defined. ABE guarantees Granular Access Control by ensuring precise and comprehensive control over who has access to a resource, what they can do, and when. Instead of issuing wide or generalized rights. Data owners can specify access policy a highly granular level. For example, in the healthcare, policies like (Role = Nurse AND Location = London) used to encrypt the data ensures that only users with matching attributes can view the data there enhancing security and privacy of the data stored on the system.

Another privacy risk mitigated by the proposed system is Insider Threat. Insider threats are a significant concern, where users may intentionally or unintentionally misuse their access privileges to compromise a system. The ABE model mitigates this risk by limiting access to data based strictly on the user's attributes, ensuring that no one has blanket access to all resources within the system. This also provides data owners with the ability to revoke access or update encryption policies dynamically and immediately locking out compromised or malicious insiders without needing to re-encrypt the entire dataset.

➤ *Research Objective 2*

- *Security Benefits of Integrating ABE in FIM Systems.*

One benefit that cannot be overemphasized is the inclusion of dynamic update. This feature enables data owners to add, edit, or remove data attributes without affecting the general functionality or security of the system. The searchable encryption is useful in environments where data attributes or user responsibilities regularly change. The dynamic updates improve data integrity by ensuring that stale or irrelevant data is removed promptly, thereby reducing security risks (Zhao et al., 2022). This capability makes the model particularly suitable for applications requiring constant adaptability, such as cloud storage systems, healthcare records, and financial institutions.

The searchable encryption introduces a groundbreaking feature in attribute-based encryption systems. While preserving the privacy of the data and the search queries, searchable encryption allows users to query and receive information based on particular attributes. In large-scale systems, where users require quick access to pertinent data without disclosing sensitive details, this feature is extremely crucial. By enabling users to swiftly search and retrieve data based on their unique attributes without necessitating the complete decryption of all stored data, the approach improves user experience (Zu, Lu and Li, 2023).

Again, Fine-grained access control is an important feature of the proposed model, ensuring that users can access only the information relevant to their roles or attributes. This mitigates risks associated with unauthorized access and over-privileged accounts, which are common vulnerabilities in traditional access control systems. Access is granted based on predefined attributes such as department, user role, or geographic location (Albulayhi et al., 2020). For instance, a doctor would only have access to patients' files, while sensitive financial data would remain inaccessible. This capability positions the proposed model as a robust solution for environments requiring stringent access control, such as government agencies, corporate databases, and research institutions.

The proposed model strengthens security by requiring users to possess valid decryption keys before accessing files. This mechanism ensures that even if an unauthorized user gains access to encrypted data, they cannot decrypt or view the contents without the appropriate keys. The decryption key mechanism facilitates a clear separation of duties, where even system administrators cannot access data without the necessary keys. It minimizes risks associated with compromised credentials or malicious insiders and maintains data confidentiality by keeping sensitive information encrypted until accessed by authorized users. This feature not only enhances security but also builds user trust.

F. Future Research Direction.

Future research regarding attribute-based encryption can be considered under one of the following.

- **ABE for IoT and Edge Devices:** Future research can focus on creating ABE schemes that are especially tailored for these types of devices.
- **Integration with Blockchain:** To enable decentralized access control systems, researchers might investigate how ABE can be combined with blockchain. This involves storing encrypted data or regulations utilizing the immutability and transparency of blockchain technology and enforcing access rights using ABE.
- **Machine Learning Integration:** A promising line of inquiry is to examine how ABE and machine learning may work together. Attempts might involve giving policy-driven access to ML models or employing ABE to protect private datasets used in AI training.

CHAPTER SIX

CONCLUSION

➤ Introduction

The process began with careful planning and developing a clear understanding of the problem. A key factor was defining the research question, which took time to refine as ABE is a complex topic. Once the scope was defined, guidance was sought from the supervisor to ensure the approach was focused and feasible.

The research process was divided into distinct phases, each dependent on various factors such as the complexity of the encryption methods, the availability of necessary tools, and the design of the study. The initial phase focused on reviewing existing literature to gain a deep understanding of the current state of ABE systems and their applications. This took around three weeks, as it was important to analyze a wide array of scholarly works, ranging from foundational theories to the latest advancements. The literature review helped in identifying gaps in current research and areas where ABE could be further optimized or applied.

Following the literature review, the research design was crafted. I selected the appropriate ABE framework to explore, based on the gaps identified in the review. This stage involved deciding on technical aspects such as the encryption model to use, the platform for implementation, and the necessary resources for data collection and testing. The implementation phase was shaped by the tools and programming languages best suited for ABE, as well as the metrics needed to evaluate its efficiency, scalability, and security. The timeline for this stage was flexible, given the potential for unexpected challenges, such as performance bottlenecks or the need to fine-tune the models.

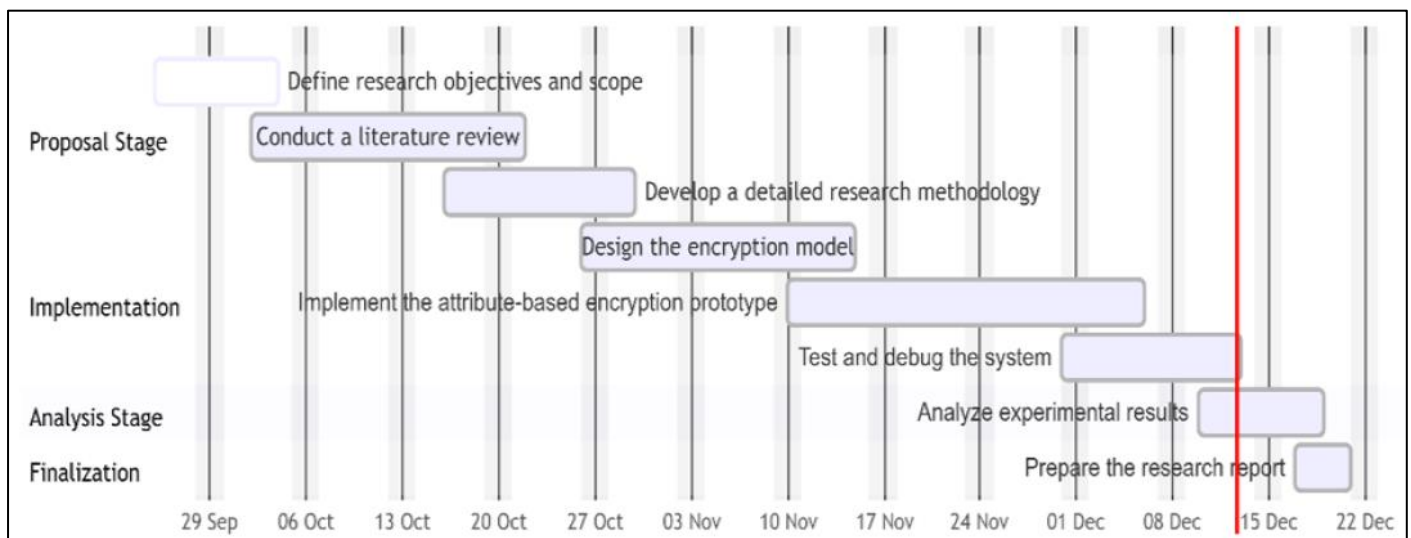


Fig 33 Gantt Chart Showing Timeline for The Project

➤ Summary of Findings

Research and advancements in the field of Attribute Based Encryption have focused on addressing critical aspects such as dynamic updates, usability design, and searchable encryption. ABE systems have been enhanced to support dynamic environments where user roles, attributes, or access policies frequently change. Efficient mechanisms for attribute revocation and policy modifications ensure that the system remains secure and responsive without requiring re-encryption of the entire dataset. These updates cater to real-world applications where access rights need to evolve rapidly, such as in organizational or cloud-based systems.

Efforts in usability design aim to simplify the deployment and management of ABE schemes. User-friendly interfaces and tools for defining and managing access policies, along with streamlined processes for key management, are being developed to reduce the complexity typically associated with cryptographic systems. Such enhancements make ABE accessible to non-expert users and scalable for widespread adoption in industries requiring robust data security.

Integrating ABE with searchable encryption techniques enables users to perform secure searches over encrypted data without compromising confidentiality. This feature is particularly valuable in scenarios like cloud storage, where users must retrieve specific information while preserving data privacy. Searchable ABE schemes allow query execution to align with attribute-based access policies, further enhancing the system's functionality and usability.

These findings underscore the evolving capabilities of ABE in addressing contemporary challenges in secure data management. By focusing on dynamic adaptability, enhanced user experience, and expanded functionalities like searchable encryption, ABE continues to solidify its position as a foundational technology for modern cryptographic applications.

➤ *Contribution to Knowledge in the Field.*

The study contributes significantly to the field of information and data security by addressing key challenges such as unauthorized access and data exposure. This work enhances security models by developing robust encryption algorithm to counteract potential threats. The study also contributes to the field of data privacy by providing advanced solution for fine-grained access control, ensuring that only authorized users with specific matching attributes can access encrypted data. It addresses critical privacy concerns by enabling data sharing without exposing sensitive information to unauthorized parties. By introducing techniques for dynamic policy updates, and attribute revocation, ABE enhances the practical implementation of privacy-preserving systems.

By studying how users interact with attribute-based access controls, researchers can identify design principles that simplify the management of attributes, keys, and policies, reducing cognitive and operational burdens. Insights from ABE usability studies can also inform the development of intuitive interfaces that balance security with accessibility, fostering broader adoption in diverse environments.

➤ *Recommendation for Practitioners and Policy Makers.*

- Policymakers should invest in educational and training initiatives to equip IT professionals, system developers, and organizational leaders with the knowledge and skills required to implement and manage ABE systems. Workshops and Certification Programs offering hands-on training to ensure stakeholders understand ABE concepts, capabilities, and limitations should be implemented.
- To maximize the impact of ABE, policymakers should advocate for the establishment of standards and frameworks that enable seamless integration into existing IT systems. This involves developing uniform specifications for ABE implementations to ensure compatibility across different vendors and platforms and also creating frameworks that allow ABE to be integrated into existing infrastructures without extensive modifications or additional overhead.
- Practitioners should ensure efficient and secure Attribute Revocation in dynamic environment, where user roles and permissions frequently change, practitioners must prioritize solutions that include robust attribute revocation functionalities. Without effective revocation, former employees, contractors, or users with outdated privileges may retain access to sensitive data, posing a significant security risk.

➤ *Research Limitation*

The review of exiting literature on Attribute-Based Encryption (ABE), identified limited in-depth exploration into the practical implementation and optimization of ABE in real-world environments, particularly in large-scale systems like cloud storage or enterprise networks.

- A major limitation of the project was the limited time available to start and complete it. This constraint restricted the scope of exploration and the ability to delve deeper into certain aspects, potentially impacting the comprehensiveness of the results and analysis.
- Another key limitation in the current body of research is the complexity involved in setting up and configuring ABE systems. There is also lack of comprehensive documentation or practical case studies detailing the real-world deployment and integration of ABE systems with existing IT infrastructures.
- Some ABE implementations run the risk of vendor lock-in because some systems could be closely linked to particular platforms or cryptographic libraries, which restricts the freedom to select different solutions.

REFERENCES

- [1]. Abdulsalam, Y.S. and Hedabou, M. (2021). Security and Privacy in Cloud Computing: Technical Review. *Future Internet*, [online] 14(1), p.11. doi:<https://doi.org/10.3390/fi14010011>.
- [2]. Abhijeet Thakurdesai, Nistor, M.S., Bein, D., Pickl, S. and Bein, W. (2022). Single Sign-On (SSO) Fingerprint Authentication Using Blockchain. *Advances in intelligent systems and computing*, pp.195–202. doi:https://doi.org/10.1007/978-3-030-97652-1_24.
- [3]. Alansari, S., Paci, F., Margheri, A. and Sassone, V. (2017). Privacy-Preserving Access Control in Cloud Federations. *2017 IEEE 10th International Conference on Cloud Computing (CLOUD)*. doi:<https://doi.org/10.1109/cloud.2017.108>.
- [4]. Aldosary, M. and Alqahtani, N. (2021). A Survey on Federated Identity Management Systems Limitation and Solutions. *International Journal of Network Security & Its Applications*, [online] 13(03), pp.43–59. doi:<https://doi.org/10.5121/ijnsa.2021.13304>.
- [5]. Almadani, M.S., Alotaibi, S., Hada Alsobhi, Hussain, O.K. and Farookh Khadeer Hussain (2023). Blockchain-based multi-factor authentication: A systematic literature review. *Internet of Things*, 23, pp.100844–100844. doi:<https://doi.org/10.1016/j.iot.2023.100844>.
- [6]. Amajuoyi, C.P., Nwobodo, L.K. and Adegbola, M.D. (2024). Transforming business scalability and operational flexibility with advanced cloud computing technologies. *Computer Science & IT Research Journal*, [online] 5(6), pp.1469–1487. doi:<https://doi.org/10.51594/csitrj.v5i6.1248>.
- [7]. Amanowicz, M., Szwaczek, S. and Wrona, K. (2024). *Data-Centric Security in Software Defined Networks (SDN)*. *Studies in big data*. Springer International Publishing. doi:<https://doi.org/10.1007/978-3-031-55517-6>.
- [8]. Annane, B., Alti, A., Laouamer, L. and Refad, H. (2022). Cx-CP-ABE: Context-aware attribute-based access control schema and blockchain technology to ensure scalable and efficient health data privacy. *SECURITY AND PRIVACY*. doi:<https://doi.org/10.1002/spy2.249>.
- [9]. Bendiab, K., Shiales, S. and Samia, B. (2018). A New Dynamic Trust Model for ‘On Cloud’ Federated Identity Management. *Portsmouth Research Portal (University of Portsmouth)*. doi:<https://doi.org/10.1109/ntms.2018.8328673>.
- [10]. Bethencourt, J., Sahai, A. and Waters, B. (2007). *Ciphertext-Policy Attribute-Based Encryption*. [online] IEEE Xplore. doi:<https://doi.org/10.1109/SP.2007.11>.
- [11]. Bogataj Habjan, K. and Pucihar, A. (2017). The Importance of Business Model Factors for Cloud Computing Adoption: Role of Previous Experiences. *Organizacija*, 50(3), pp.255–272. doi:<https://doi.org/10.1515/orga-2017-0013>.
- [12]. Crane, C. (2023). *Compromised Credentials: 7 Ways to Fight Credential Attacks*. [online] Hashed Out by The SSL Store™. Available at: <https://www.thesslstore.com/blog/compromised-credentials-ways-to-fight-credential-attacks/> [Accessed 7 Dec. 2024].
- [13]. Das, M. (2024). Fine-Grained Access Through Attribute-Based Encryption for Fog Computing. pp.405–424. doi:<https://doi.org/10.1002/9781394175345.ch17>.
- [14]. Deshmukh, J.Y., Yadav, S.K. and Bhandari, G.M. (2021). Attribute-Based encryption mechanism with Privacy-Preserving approach in cloud computing. *Materials Today: Proceedings*. doi:<https://doi.org/10.1016/j.matpr.2021.05.609>.
- [15]. Fu, X., Ding, Y., Li, H., Ning, J., Wu, T. and Li, F. (2022). A survey of lattice based expressive attribute based encryption. *Computer Science Review*, [online] 43, p.100438. doi:<https://doi.org/10.1016/j.cosrev.2021.100438>.
- [16]. Fun, T.S. and Samsudin, A. (2017). Attribute Based Encryption—A Data Centric Approach for Securing Internet of Things (IoT). *Advanced Science Letters*, 23(5), pp.4219–4223. doi:<https://doi.org/10.1166/asl.2017.8315>.
- [17]. Garnaev, A. and Trappe, W. (2022). An eavesdropping and jamming dilemma with sophisticated players. *ICT Express*. doi:<https://doi.org/10.1016/j.icte.2022.06.002>.
- [18]. Hamza, A. and Kumar, B. (2020). *A Review Paper on DES, AES, RSA Encryption Standards*. [online] IEEE Xplore. doi:<https://doi.org/10.1109/SMART50582.2020.9336800>.
- [19]. He, G., Li, C., Shu, Y. and Luo, Y. (2024). Fine-grained access control policy in blockchain-enabled edge computing. *Journal of network and computer applications*, 221, pp.103706–103706. doi:<https://doi.org/10.1016/j.jnca.2023.103706>.
- [20]. Hou, X., Zhang, L., Wu, Q. and Fatemeh Rezaeibagha (2023). Collusion-resistant dynamic privacy-preserving attribute-access control scheme based on blockchain. *Journal of King Saud University - Computer and Information Sciences*, 35(8), pp.101658–101658. doi:<https://doi.org/10.1016/j.jksuci.2023.101658>.
- [21]. Imam, R., Kumar, K., Raza, S.M., Sadaf, R., Anwer, F., Fatima, N., Nadeem, M., Abbas, M. and Rahman, O. (2022). A systematic literature review of attribute based encryption in health services. *Journal of King Saud University - Computer and Information Sciences*. doi:<https://doi.org/10.1016/j.jksuci.2022.06.018>.
- [22]. Keltoum, B. and Samia, B. (2017). A dynamic federated identity management approach for cloud-based environments. *Proceedings of the Second International Conference on Internet of things, Data and Cloud Computing*. doi:<https://doi.org/10.1145/3018896.3025152>.
- [23]. Li, J., Zhang, Y., Ning, J., Huang, X., Poh, G.S. and Wang, D. (2020). Attribute Based Encryption with Privacy Protection and Accountability for CloudIoT. *IEEE Transactions on Cloud Computing*, pp.1–1. doi:<https://doi.org/10.1109/tcc.2020.2975184>.
- [24]. Li, X., Wang, H., Ma, S., Xiao, M. and Huang, Q. (2024). Revocable and verifiable weighted attribute-based encryption with collaborative access for electronic health record in cloud. *Cybersecurity*, 7(1). doi:<https://doi.org/10.1186/s42400-024-00211-1>.

- [25]. Liang, K., Fang, L., Susilo, W. and Wong, D.S. (2013). A Ciphertext-Policy Attribute-Based Proxy Re-encryption with Chosen-Ciphertext Security. *Intelligent Networking and Collaborative Systems*. doi:<https://doi.org/10.1109/incos.2013.103>.
- [26]. Linthicum, D.S. (2019). Approaching Cloud Computing Performance. *IEEE Cloud Computing*, 5(2), pp.33–36.
- [27]. Liu, Y., Zhang, Y., Ling, J. and Liu, Z. (2018). Secure and fine-grained access control on e-healthcare records in mobile cloud computing. *Future Generation Computer Systems*, [online] 78, pp.1020–1026. doi:<https://doi.org/10.1016/j.future.2016.12.027>.
- [28]. Luo, F., Wang, H., Yan, X. and Wu, J. (2024). Key-Policy Attribute-Based Encryption with Switchable Attributes for Fine-Grained Access Control of Encrypted Data. *IEEE Transactions on Information Forensics and Security*, [online] pp.1–1. doi:<https://doi.org/10.1109/tifs.2024.3432279>.
- [29]. McCarthy, M. (2023). *Understanding Role-Based Access Control (RBAC)*. [online] www.strongdm.com. Available at: <https://www.strongdm.com/rbac>.
- [30]. Mohammad, A. (2022). Distributed Authentication and Authorization Models in Cloud Computing Systems: A Literature Review. *Journal of Cybersecurity and Privacy*, 2(1), pp.107–123. doi:<https://doi.org/10.3390/jcp2010008>.
- [31]. Mortágua, D., Zúquete, A. and Salvador, P. (2024). Enhancing 802.1X authentication with identity providers using EAP-OAUTH and OAuth 2.0. *Computer networks*, 244, pp.110337–110337. doi:<https://doi.org/10.1016/j.comnet.2024.110337>.
- [32]. Nabil, M., Bima, M., Alsharif, A., Johnson, W., Gunukula, S., Mahmoud, M. and Abdallah, M. (2019). Priority-Based and Privacy-Preserving Electric Vehicle Dynamic Charging System With Divisible E-Payment. *Smart Cities Cybersecurity and Privacy*, pp.165–186. doi:<https://doi.org/10.1016/b978-0-12-815032-0.00012-3>.
- [33]. Naik, N. and Jenkins, P. (2017). Securing digital identities in the cloud by selecting an apposite Federated Identity Management from SAML, OAuth and OpenID Connect. *2017 11th International Conference on Research Challenges in Information Science (RCIS)*. doi:<https://doi.org/10.1109/rcis.2017.7956534>.
- [34]. Niskanen, T. and Salonen, J. (2023). Enabling fine-grained access control in information sharing with structured data formats. *European Conference on Cyber Warfare and Security*, 22(1), pp.332–340. doi:<https://doi.org/10.34190/eccws.22.1.1143>.
- [35]. Panathula, M. (2024). *Federated Identity Management: A Comprehensive Guide | 2024 | Zluri*. [online] Zluri.com. Available at: <https://www.zluri.com/blog/federated-identity-management> [Accessed 5 Nov. 2024].
- [36]. Prantl, T., Zeck, T., Horn, L., Lukas ffländer, Bauer, A., lexandra Dmitrienko, Krupitzer, C. and Kounev, S. (2023). Towards a cryptography encyclopedia: a survey on attribute-based encryption. *Journal of Surveillance Security and Safety*, [online] 4(4), pp.129–54. doi:<https://doi.org/10.20517/jsss.2023.30>.
- [37]. Premarathne, U.S., Khalil, I., Tari, Z. and Zomaya, A. (2017). Cloud-Based Utility Service Framework for Trust Negotiations Using Federated Identity Management. *IEEE Transactions on Cloud Computing*, 5(2), pp.290–302. doi:<https://doi.org/10.1109/tcc.2015.2404816>.
- [38]. Raj, A. (2022). *SAML vs OAuth : Building Blocks to Federated Identity Management*. [online] Medium. Available at: <https://blog.devgenius.io/saml-vs-oauth-building-blocks-to-federated-identity-management-f36ca58f7aa0> [Accessed 7 Dec. 2024].
- [39]. Reshma Siyal and Long, J. (2024). Secure Cloud Data with Attribute-based Honey Encryption. *Research Square (Research Square)*. doi:<https://doi.org/10.21203/rs.3.rs-4115057/v1>.
- [40]. Si-Ahmed, A., Ali Al-Garadi, M. and Boustia, N. (2023). Survey of Machine Learning based intrusion detection methods for Internet of Medical Things. *Applied Soft Computing*, 140, pp.110227–110227. doi:<https://doi.org/10.1016/j.asoc.2023.110227>.
- [41]. Suryawanshi, V. and Sural, S. (2024). Ciphertext Policy Attribute Based Encryption with Intel SGX. *arXiv (Cornell University)*. doi:<https://doi.org/10.48550/arxiv.2409.07149>.
- [42]. Vignesh, M. and Naresh, Dr. (2020). Exploration of Attribute Based Encryption Schemes on Cloud Computing Storage. *International Journal of Recent Technology and Engineering*, 8(5), pp.5367–5371. doi:<https://doi.org/10.35940/ijrte.e6764.018520>.
- [43]. Wang, J., Liang, J., Ding, Y., Tang, S. and Wang, Y. (2023). Ciphertext-policy attribute-based encryption supporting policy-hiding and cloud auditing in smart health. *Computer Standards & Interfaces*, 84, pp.103696–103696. doi:<https://doi.org/10.1016/j.csi.2022.103696>.
- [44]. Wang, X., Yu, M., Wang, Y., Pi, Y., Xu, P., Wang, S., Jin, H. and Han, M. (2024). Attribute-Based Access Control Encryption. *IEEE Transactions on Dependable and Secure Computing*, pp.1–15. doi:<https://doi.org/10.1109/tdsc.2024.3481497>.
- [45]. Wang, Y., Su, Z., Zhang, N., Xing, R., Liu, D., Luan, T.H. and Shen, X. (2022). A survey on metaverse: Fundamentals, security, and privacy. *IEEE Communications Surveys & Tutorials*, 25(1), pp.1–1. doi:<https://doi.org/10.1109/comst.2022.3202047>.
- [46]. www.okta.com. (n.d.). *What Is Federated Identity? | Okta*. [online] Available at: <https://www.okta.com/identity-101/what-is-federated-identity/>.
- [47]. Yadav, U.C. and Ali, S.T. (2015). Ciphertext policy-hiding attribute-based encryption. pp.2067–2071. doi:<https://doi.org/10.1109/icacci.2015.7275921>.
- [48]. Yan, L., Wang, G., Yin, T., Liu, P., Feng, H., Zhang, W., Hu, H. and Pan, F. (2024). Attribute-Based Searchable Encryption: A Survey. *Electronics*, 13(9), pp.1621–1621. doi:<https://doi.org/10.3390/electronics13091621>.
- [49]. Yi, W., Wang, C., Kuzmin, S., Gerasimov, I. and Cheng, X. (2024). Weighted Attribute-Based Proxy Re-Encryption Scheme with Distributed Multi-Authority Attributes. *Sensors*, 24(15), p.4939. doi:<https://doi.org/10.3390/s24154939>.

- [50]. Yin, H., Zhu, Y., Deng, H., Ou, L., Qin, Z. and Li, K. (2024). Privacy-Preservation Enhanced and Efficient Attribute-Based Access Control for Smart Health in Cloud-Assisted Internet of Things. *IEEE Internet of Things Journal*, [online] pp.1–1. doi:<https://doi.org/10.1109/jiot.2024.3470891>.
- [51]. Zhang, D., Yang, X., Jia, Z., Li, H., Guo, X. and Wang, Q. (2023). Improved CP-ABE Algorithm Based on Identity and Access Control. pp.482–487. doi:<https://doi.org/10.1109/iaecst60924.2023.10503198>.
- [52]. Zhang, Y., Deng, R.H., Xu, S., Sun, J., Li, Q. and Zheng, D. (2020). Attribute-based Encryption for Cloud Computing Access Control. *ACM Computing Surveys*, 53(4), pp.1–41. doi:<https://doi.org/10.1145/3398036>.