SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V11I12P110

Volume 11 Issue 12, 108-118, December 2024
© 2024 Seventh Sense Research Group®

Original Article

A Privacy-Enhanced Framework for Securing User Data on Cloud-Based Social Networks

S. Nasira Tabassum[1], Gangadhara Rao Kancherla[2]

[1,2]*Department of Computer Science & Engineering, Acharya Nagarjuna University, Andhra Pradesh, India.*

[1]*Corresponding Author : nasira.tabassum@gmail.com*

Received: 06 October 2024 Revised: 07 November 2024 Accepted: 05 December 2024 Published: 31 December 2024

Abstract *- Online Social Networks (OSNs) have become integral to modern life, enabling people to communicate, share information, and stay connected over vast distances. However, the rising use of OSNs has sparked significant concerns regarding the privacy and security of user data. This paper presents an innovative method for strengthening data privacy and security in cloud-enabled OSNs utilizing an E-ABE system. The proposed solution employs flow graph analysis to verify relationships, ensuring secure user data exchange while offering fine-grained access control. This approach addresses the limitations of existing methods, which often fail to provide comprehensive privacy and security measures. By leveraging cryptographic techniques and secure communication protocols, the E-ABE model allows for the controlled sharing of sensitive information, ensuring that only authorized users can access data. The system's design includes roles for Cloud Service Providers (CSPs), Trusted Attribute Authorities (TAAs), and end-users (followers and followees), each contributing to the overall security framework. Experimental results demonstrate the effectiveness of the proposed method in reducing key generation time, data encryption and decryption time, and communication costs while maintaining high levels of authorization accuracy. This research contributes to the field by providing a robust solution for protecting user data in cloud-based OSNs, highlighting its potential for broader application in other domains requiring stringent data privacy and security measures.*

Keywords *- OSNs, Data privacy, Data security, Attribute-Based, Encryption (ABE), Cloud computing, Flow graph analysis, Fine-grained access control.*

1. Introduction

1.1. Background on the Proliferation of OSNs

In the digital age, the manner in which individuals communicate and interact and OSNs have revolutionized shared information. Platforms such as LinkedIn, Facebook, Twitter, and Instagram have experienced exponential growth, resulting in millions of active users on a global scale. These networks facilitate social interaction by allowing users to connect with friends, family, and colleagues, share personal updates, and engage with a wide array of content. The pervasive use of OSNs highlights their significance in modern society, making them indispensable tools for communication and networking [1]. Despite their popularity and benefits, OSNs have also introduced substantial risks to user privacy and data security. The extensive sharing of personal information on these platforms makes them prime targets for malicious individuals looking to misuse sensitive data. Incidents of data breaches, identity theft, and unauthorized data access have become increasingly common, raising concerns among users and regulators alike [2]. These challenges necessitate the development of robust security mechanisms to safeguard user information and maintain trust in these platforms. Addressing privacy and security concerns in OSNs is of paramount importance. Users frequently disclose various types of personal information, such as names, contact details, social security numbers, financial records, and private communications. If this data is compromised, the consequences can be severe, including financial loss, reputational harm, and potential legal repercussions. A vital research focus is ensuring user data's confidentiality, integrity, and availability in OSNs [3]. Existing security measures in OSNs often fall short of providing comprehensive protection. Many platforms rely on basic encryption techniques and access control mechanisms that may not adequately address the complex security requirements of modern OSNs. The need for more sophisticated and effective security solutions has driven research efforts to explore advanced cryptographic methods and privacy-preserving technologies tailored to the unique challenges of OSNs [4].

1.2. Importance of Privacy and Security in OSNs

Privacy and security are paramount in the context of OSNs due to the sensitive nature of the data involved. Users expect their personal information to be protected against unauthorized access and misuse. Ensuring data privacy involves protecting the data from external threats and implementing controls that limit access to authorized individuals based on well-defined policies [5]. Effective security measures in OSNs help maintain user trust and engagement, which is crucial for these platforms' continued growth and success. The challenge of ensuring privacy in OSNs is compounded by these networks' dynamic and interconnected nature. Users frequently interact with a diverse group of individuals, including friends, acquaintances, and strangers. This creates a complex web of relationships and interactions that must be managed to prevent unauthorized data access. Traditional security models often struggle to handle the granularity required to manage these relationships effectively, necessitating the development of new approaches that work on absolute-grained control mechanisms [6].

Additionally, the cloud-based infrastructure frequently employed to support OSNs introduces further security concerns. Scalability, cost efficiency, and flexibility are among the numerous advantages of cloud computing. Nevertheless, it also poses obstacles regarding transparency, control, and data sovereignty. Users must ensure their data is secure and private to ensure cloud security [7]. This necessitates the implementation of trusted frameworks and robust encryption mechanisms.

Researchers have explored cryptographic methods to enhance data privacy and security in OSNs, with ABE being a key solution. ABE links encryption keys to attributes, allowing data owners to set access policies so only authorized users can decrypt the data. Integrating ABE into cloud-based OSNs boosts security and privacy without compromising usability [8]. Existing security models for cloud-based social networks lack the flexibility to handle dynamic user interactions and fine-grained access control. This study bridges these gaps by introducing a scalable Enhanced Attribute-Based Encryption (E-ABE) framework. The approach ensures secure, efficient data sharing through innovative flow graph analysis

2. Literature Review

2.1. Privacy-Preserving Data Publishing for Social Networks

One of the fundamental works in privacy-preserving data publishing is by Zhou, Pei, and Luk (2008), who addressed the challenges of anonymizing social network data. Their approach focuses on preserving the structural properties of social networks while anonymizing the data to protect user privacy. They proposed several anonymization techniques, such as k-anonymity, l-diversity, and t-closeness, to prevent re-identification attacks. By transforming the social network graph and ensuring that each user is indistinguishable from at least k-1 others, they effectively reduced the risk of privacy breaches. Their work highlights the balance between data utility and privacy, emphasizing that anonymized data should still be useful for analysis while protecting individual identities. This pioneering work laid the groundwork for subsequent research in privacy-preserving social network analysis, demonstrating that it is possible to share useful data without compromising user privacy [9]. In [10], the authors have proposed the concept of privacy wizards to help users manage their privacy settings on social networking sites. Privacy wizards are interactive tools that guide users through the process of configuring their privacy settings based on their preferences and social context. The authors conducted user studies to understand social network users' common privacy concerns and behaviors. They developed privacy wizards that simplify the configuration process, thereby facilitating the protection of users' privacy in accordance with their discoveries. This method resolves the usability obstacles linked to intricate privacy settings, enabling users to make well-informed decisions regarding their data.

The design of user-centric privacy tools has been significantly influenced by the concept of privacy wizards, which underscores the significance of user education and empowerment in protecting privacy. In [11], the experts explored privacy-preserving social network analysis techniques, focusing on the trade-offs between data utility and privacy. They proposed methods to anonymize social network graphs while preserving their structural properties, enabling meaningful analysis without compromising user privacy. Their work involved developing algorithms for graph anonymization and applying them to real-world social network data. The results demonstrated that performing social network analysis on anonymized data is possible, providing valuable insights while protecting individual identities. This research has significant implications for organizations that need to analyze social network data for various purposes, such as marketing, security, and social science research.

2.2 Secure Data Sharing in Cloud Computing

A recent model emphasizes the need for a new framework aimed at secure data sharing within cloud environments, focusing specifically on the challenges posed by outsourced data storage. The researchers introduced a Key-Policy Attribute-Based Encryption (KP-ABE) scheme as part of their proposed solution. This scheme enables data owners to encrypt data using a set of attributes and regulate access through policies defined over these attributes. This method ensures that only users with matching attributes can decrypt the data, providing fine-grained access control.

Their framework also includes mechanisms for key revocation, ensuring that users who no longer meet the access criteria are prevented from accessing the data. This work is significant as it addresses the dynamic nature of cloud environments, where users' access rights may change over time. By integrating KP-ABE with cloud storage, the authors demonstrated an effective way to secure sensitive data while maintaining flexibility and scalability in access control [12]. An ABE scheme is a fine-grained access control scheme for encrypted data. This approach enables data owners to encrypt data based on a predetermined set of attributes and establish access policies that specify the combinations of attributes necessary to decrypt the data. The ABE scheme supports complex access structures, such as boolean formulas, enabling more flexible and expressive access control. Their work was a breakthrough in cryptographic research, extending the traditional public key encryption model to support attribute-based access control. This approach is particularly well-suited for cloud environments and social networks, where data access needs to be carefully controlled based on user attributes. This scheme has influenced numerous subsequent works, furthering the development of privacy-preserving technologies in various domains [13].

Authors in [14] addressed the privacy issue in cloud computing by proposing a proxy re-encryption scheme. This scheme enables a proxy to transform ciphertexts from one encryption key to another without exposing the underlying data. This feature is especially beneficial in situations where data must be shared among multiple users with varying levels of access permissions. By using proxy re-encryption, Patsakis et al. provided a way to delegate decryption rights without compromising the security and privacy of the data. Their scheme also supports efficient key revocation, ensuring users who lose access rights can no longer decrypt the data. This work is significant for cloud-based social networks, where dynamic user interactions and changing access rights necessitate flexible and secure data-sharing mechanisms.

2.3. Secure Multi-Party Computation for Social Networks

The concept of Secure Multi-party Computation (SMC) enables collaborative computations on private data without revealing the inputs to the parties involved. In the context of social networks, SMC can be used to perform joint analysis of users' data while ensuring that each user's data remains private. The authors developed protocols that allow multiple parties to collaboratively compute a function based on their inputs while ensuring that the privacy of the individual inputs is preserved. This method is especially advantageous for applications in which data sharing is essential, but privacy concerns prohibit direct data exchange. SMC has been applied to various privacy-preserving applications, including social network analysis, collaborative filtering, and privacy-preserving data mining [15, 16].

The concept of differential privacy is a robust privacy-preserving mechanism that provides strong guarantees against re-identification attacks. Differential privacy ensures that the output of a computation does not reveal significant information about any individual in the dataset, regardless of the auxiliary information available to the attacker [17]. This is accomplished by introducing controlled noise to the data or query results, which complicates the process of inferring the presence or absence of any individual. In the context of social networks, differential privacy can be used to release aggregate statistics and perform data analysis without compromising user privacy. This work has profoundly impacted the privacy-preserving data analysis field, providing a mathematical foundation for developing privacy-preserving algorithms and systems [18].

2.4. Privacy-Preserving Data Mining Techniques

The field of privacy-preserving data mining introduces methods for analyzing data while safeguarding individual privacy. This approach involves transforming data in a manner that prevents the exposure of sensitive information yet still allows valuable patterns to be identified [19]. Techniques such as data perturbation, anonymization, and secure multi-party computation were proposed to achieve this goal. These methods have been widely applied across various fields, including social network analysis, healthcare, and finance, where protecting data privacy is paramount. The work of experts in this area has established a strong foundation for future research, proving both the importance and practicality of embedding privacy into data analysis processes [20]. Authors in [21] proposed a comprehensive framework for secure and privacy-aware data sharing in social networks.

Their framework integrates fine-grained access control policies with privacy-preserving techniques to ensure users can share data securely and privately. The authors proposed a rule-based method for defining access control policies, utilizing user relationships and attributes. They integrated cryptographic techniques to enforce these policies, ensuring only authorized users can access the shared data [22]. This work addresses the dual security and privacy challenges in social networks, providing a robust solution for controlled data sharing. Researchers' framework has been influential in the design of secure social network systems, highlighting the need for integrating access control and privacy-preserving mechanisms. Work in [23] presents a privacy-preserving technique for OSNs, focusing on sensitive attribute protection and minimizing structural changes to datasets. However, a notable drawback is its reliance on noisy node perturbation, which can lead to skewness and similarity attacks.

In contrast, the proposed E-ABE framework addresses this limitation by dynamically validating user relationships and applying fine-grained access control policies, eliminating the need for noisy nodes while enhancing privacy and preserving the utility of OSN data. The study in [24] addresses privacy risks posed by inference attacks in social network data sharing by optimizing the balance between user utility and privacy preservation. It introduces two methods: the Efficiency-based Privacy-Preserving Disclosure algorithm (EPPD) for high utility and a multi-dimensional Knapsack Problem (d-KP) approach for low computational complexity. While effective, these methods focus on inference attack defense without considering dynamic relationship validation, which is addressed in the proposed E-ABE framework by leveraging flow graph analysis for secure and efficient data sharing.

3. Proposed System

The primary goals of the proposed system are to safeguard the privacy of genuine data while using cloud-based OSN services and to enforce fine-grained access control within these systems. The threat model considers the CSP trustworthy and inquisitive, implying that although the CSP adheres to the system protocols, it still has a strong interest in examining user data and is still highly interested in examining and interpreting users' sensitive data. Additionally, it is assumed that the TAA and followees are reliable entities.

Followers, considered unauthorized entities, may occasionally attempt to interfere with the confidential information of followees for commercial or business purposes. It is assumed that standard security protocols, including SSL and SSH, are used to secure the communication channels between all system entities. Furthermore, it is presumed that the CSP and its followers do not have any mutual interests that would motivate them to engage in a conspiracy to violate the privacy of their followers.

- Cloud Service Provider (CSP): The CSP provides infrastructure support, including cloud storage and computational resources. It initializes the system by generating global system parameters and master keys, ensuring secure data storage and management. The CSP also facilitates encrypted data storage while ensuring no direct access to the data content.

- Trusted Attribute Authorities (TAAs): TAAs are designated entities responsible for generating and distributing cryptographic keys linked to user attributes. They verify user credentials, manage attribute definitions, and enforce access policies to ensure secure data sharing. TAAs play a critical role in attribute-based encryption and key revocation processes.

- End-Users (Followers and Followees): End-users interact within the system as followers (data receivers) and followees (data providers). Followers apply access policies to encrypted data, determining which followers can decrypt and access the content. Followers, on the other hand, can only access data if their attributes satisfy the defined policies.

In this model, the online social network is divided into two distinct categories of end users: followers and followees. These users are actively engaged in the exchange and reception of data within cloud-based OSNs. Figure 1 illustrates the framework of the proposed approach. The E-ABE model is illustrated through a directed flow graph that comprises vertices and edges. In this model, the vertices represent individuals (followers or followees), and the edges signify communication among users. This structure improves the system's utility while addressing the need for genuine data privacy in cloud-enabled OSNs.

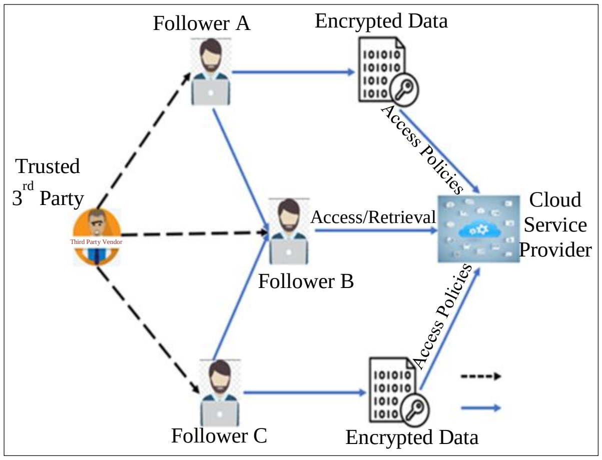

Fig. 1 Proposed model privacy preserving OSN system

- The Proposed Enhanced Attribute-Based Encryption (E-ABE) Algorithm:

The E-ABE algorithm is designed to address the limitations of existing security models in cloud-based social networks. The following sections detail the core components and mechanisms of the E-ABE framework, emphasizing its innovative approach to ensuring secure and efficient data sharing.

The OSN server offers read, write, and update services. In this system, user A can message user B if A follows B, confirmed by a directed edge. Factors like strength and certainty secure data transmission. After verifying their relationship, the followee encrypts the message using attribute-based encryption and a data access policy based on the follower's attributes. The cloud server stores the encrypted message, ensuring only followers matching the access policy can decrypt it. This approach enables secure data sharing between OSNs while simultaneously enhancing privacy and utility. Consider a flow graph H$=(V,E,\phi)$ where V represents the nodes, E the edges, and ϕ the flow function. The graph F is directed, finite, and acyclic. The set of edges $E \subseteq V \times V$, and the flow function $\phi: E \rightarrow R^+$ maps each edge to a non-negative real number. For any edge $(u,v) \in E(u, v)$, u is the input node, and v is the output node. $In(u)$ and $Out(u)$ are the inputs and outputs of a node u, respectively, for all $u \in V$. In graph H, the I/O values are represented by $In(H)=\{u \in V: In(u)=\phi\}$ and $Out(H)=\{u \in V: Out(u)=\phi\}$. Internal nodes lack I/O connections, whereas external nodes possess both. For any $(u,v) \in E$, the flow from u to v is represented by $\phi(u,v)$. Assuming $\phi(u,v) \neq 0$ for all $(u,v) \in E$, the inflow at any node u in the flow graph H is given by:

$$\emptyset_+(u) = \sum_{v \in In(u)} \emptyset(u,v) \tag{1}$$

And the outflow at any node u is given by:

$$\emptyset_-(u) = \sum_{v \in Out(u)} \emptyset(u,v) \tag{2}$$

Similarly, the complete graph H's inflow and outflow metrics are defined as follows:

$$\emptyset_{+}(H) = \sum_{v \in In(G)} \emptyset - (u) \tag{3}$$

$$\emptyset_{-}(H) = \sum_{v \in Out(G)} \emptyset - (u) \tag{4}$$

Now, assume that for any internal node u, $\emptyset_{+}(u) = \emptyset_{-}(u) = \emptyset(u)$, where $\emptyset(u)$ means the flow of data through node u. This leads us to the following flow conservation equation: $\emptyset_{+}$ (H)=∅- (H)=∅(H), where ∅(H) denotes the total through flow of the graph H.

3.1. Flow Graph Normalization

The flow graph that has been normalized is now defined as follows. The normalized flow graph *H (M, D, σ)* has a cycle, is directed, and is finite. Here, M stands for the collection of nodes, *D⊆M×M* represents both the set of directed branches and the normalized flow of variables *(u,v)* is denoted by *σ:D→<0,1>*. To determine the strength of *(u,v)*, use the following formula:

$$\sigma(u, v) = \frac{\emptyset(v,u)}{\emptyset(H)} \tag{5}$$

Where 0<σ(u,v)<1. The strength of (u,v) serves as an indicator of the proportion of the total flow that passes through the branches. In the computation process, the normalized flow graph is consistently taken into account. As follows, the normalized inflow and outflow measures are provided for each node u in the flow graph H:

$$\sigma_{+}(u) = \frac{\emptyset_{+}(u)}{\emptyset_{+}(H)} = \sum_{v \in In(u)} \sigma(u, v) \tag{6}$$

$$\sigma(u, v) = \frac{\emptyset(u,v)}{\emptyset(H)} = \sum_{v \in Out(H)} \sigma(u) \tag{7}$$

For every internal node u, $\sigma_{+}(u) = \sigma_{-}(u) = \sigma(u)$ where $\sigma(u)$ is a normalized through flow of u is calculated in Equation (8).

$$\sigma_{+}(H) = \frac{\emptyset_{+}(H)}{\emptyset(H)} = \sum_{u \in In(H)} \sigma_{-}(u)\sigma_{-}(H) = \frac{\emptyset_{-}(u)}{\emptyset_{-}(H)} = \sum_{v \in Out(H)} \sigma(u) \tag{8}$$

Flow graph analysis is a critical component of the E-ABE system, designed to validate user relationships and regulate data access securely. In this approach, nodes represent users, while directed edges signify communication or data flow between them.

The strength of these edges, calculated using normalized flow functions as shown in Equations (5)-(8), determines the proportion of data flow through the network. This analysis ensures that access control policies are enforced dynamically by verifying user connections and interaction validity. By integrating this mechanism, the E-ABE framework strengthens data security while maintaining efficient and precise access control in cloud-based social networks.

3.1.1. Coverage and Certainty Factors

The hypothesis's degree of truth is quantified by the certainty factor, which is a numerical value. Equations (9) and (10) define the certainty and coverage factors for each branch (u,v) of the flow graph H.

$$cert(u, v) = \frac{\sigma(v,u)}{\sigma(v)} \tag{9}$$

$$con(u, v) = \frac{\sigma(v,u)}{\sigma(u)} \tag{10}$$

Where σ(u) and σ(v) are non-zero, the following equations provide certain properties of coverage factors and certainty:

$$\sum_{v \in Out(u)} cert(u, v) = 1 \tag{11}$$

$$\sum_{u \in In(v)} cov(u, v) = 1 \tag{12}$$

$$\sigma(u) = \sum_{v \in Out(u)} \sigma(u, v) \tag{13}$$

$$\sigma(v) = \sum_{u \in In(v)} \sigma(u, v) \tag{14}$$

$$cert(u, v) = cov(u, v) \frac{\sigma(v)}{\sigma(u)} \tag{15}$$

The properties mentioned earlier are based on Bayesian principles and can be interpreted deterministically to describe flow distribution across network branches. A directed path from node *u* to node *v* in the graph *H*, where *u≠v*, consists of a sequence of nodes $u_1, u_2, \dots, u_n$. Therefore, $u_1 = u_n$, $u_n = v$, and $(u_j, u_j+1) \in D$ for every *j*, $1 \leq j \leq n-1$. Therefore, the path from u to v is denoted as *[u⋯v]*. Currently, the three critical variables for the flow graph in question are calculated as follows: certainty, strength, and coverage.

$$cert[u_1 \dots u_n] = \Pi_{j=1}^{n-1} cert(u_j, u_{j+1}) \tag{16}$$

$$cov[u_1 \dots u_n] = \Pi_{j=1}^{n-1} cov(u_j, u_{j+1}) \tag{17}$$

$$\sigma(u, v) = \sigma(u) cert[u \cdots v] = \sigma(v) cov[u \cdots v] \tag{18}$$

Consequently, the connection between *u* and *v* in *H* is denoted by all paths from *u* to *v (u≠v)*. In other words, the nodes *u* and *v* are used to determine the subgraph *⟨u,v⟩* of *H*. Therefore, for every connection, the following are the definitions of the certainty, coverage, and strength factors: *⟨u,v⟩:*

$$cert\langle u, v \rangle = \sum_{[u \cdots v] \epsilon \langle u,v \rangle} cert[u \cdots v] \tag{19}$$

The coverage connection of $\langle u, v \rangle$ is given by:

$$cov\langle u, v \rangle = \sum_{[u \cdots v] \epsilon \langle u,v \rangle} cov[u \cdots v] \tag{20}$$

The connection quality is presented as follows:

$$\sigma\langle u, v \rangle = \sum_{[u \cdots v] \epsilon \langle u,v \rangle} \sigma[u \cdots v] = \sigma(u) cov\langle u, v \rangle \tag{21}$$

A path *[u⋯v]* is full when, for graph H, U is the input and V is the output. The complete connection from *u* to *v* in *H* is the collection of all complete paths from *u* to *v*. Access control methods are typically easier to implement when these complete paths indicate the connection between two nodes. A single branch (u,v) is derived for every complete connection ⟨u,v⟩ in graph H, such that σ(u,v)= σ⟨u,v⟩, cert(u,v)=cert⟨u,v⟩, and cov(u,v)=cov⟨u,v⟩. A new flow graph H′ is generated by these substitutions, with σ(H)=σ(H'). The combined flow graph is represented by the new flow graph H′, which shows how the values of input and output are related, including the valid connections between two nodes. To better understand these ideas, let's look at the social network analysis problem in Figure 1. In this case, it uses hypothetical information derived from one hundred social media users. Group B consists of social pages, and Group A consists of people who use social networks; studying their relationships is the main goal of this study. Three separate groups of characteristics make up the flow graph., each of which corresponds to a distinct age group of social network users: q_1 (older adults), q_2 (middle-aged adults), and q_3 (younger adults). These age groups are classified as p_1 (high), p_2 (middle), and p_3 (low), and they engage with social media pages in four categories: r_1 (business news), r_2 (cinema), r_3 (political updates), and r_4 (education).

Different categories of online social network users are defined by their social class, which is based on the level of access they have to various social networking sites. Initially, social media subscriptions are analyzed according to age groups. Among people in the q_1 age bracket, the data shows that 19% have subscribed to the social page r_1. Also, out of the age group q_2, 44% have subscribed; out of the age group q_3, 18% have done the same; and so on. To directly evaluate the subscriptions of social pages by a range of users from different socioeconomic classes, proposed work removes the age group attribute from the flow diagram. Notably, 22% of r_1's subscribers are from social class p_1, 78% are from social class p_2, and so on.

3.2. Improved Attribute-Based Encryption for Secure Proxy Re-Encryption and Efficient User Revocation

One enhanced version of the suggested attribute-based encryption system uses a single CSP in conjunction with numerous TAA. The organization and definition of user-defined attributes are entirely the responsibility of these attribute authorities. The following is the proposed operational framework of the attribute-based encryption system. CSPs are the ones who start the system. The CSP executes the following algorithms in this suggested model during the system initialization phase. Two main components make up this step: setting up the system and establishing the TAA. There is a mathematical representation for these steps in Equations (22) and 23, respectively.

$$\textit{System Setup ():}\ Sys_{set}(H) \rightarrow (PK_0, MK_0) \tag{22}$$

At this stage, the CSP receives a large security parameter, *S*, as input and generates the system parameters PK_0 and the master key MK_0. While the master key remains confidential, the system parameter PK_0 is publicly available.

$$\textit{Create TAA ():}\ Crt\ TAA(PK_0, MK_0, PK_i) \rightarrow (MK_i) \tag{23}$$

At this stage, the CSP creates a master key for every TAA linked to the system. During this phase, the master key and the public system *parameters are used to generate the master key (MK_i.) for every TAA.*

3.3. Secure Search in Social Networks

The data contents are encrypted by the follower and shared with the group of followers during this phase. Equation (24) mathematically illustrates the encryption process.

$$\textit{Encryption ():}\ \mathrm{Encrypt}(\mathrm{PK}_0, \mathrm{f}, \mathrm{AP}, \mathrm{PK}_0 | \mathrm{a} \epsilon \mathrm{AP}) \rightarrow \mathrm{CP}^{\sim} \tag{24}$$

After encrypting the data file with the *Encrypt ()* algorithm, the user proceeds to encrypt the keywords with *KwEnc()*. The ciphertext, $CP^{\sim}$, is the result of this procedure

$$\textit{KwdEnc ():}\ \left(kw_i, P_{uj}\right) \rightarrow \left(C_j, w_i\right) \tag{25}$$

The encrypted keyword, C_j, w_i is generated as an output by this procedure in Equation (25), which takes the user's public key and the keyword as inputs.

$$\textit{Encrypt ():}\ (PK_0, f, AP, PK_0 | a \epsilon AP) \rightarrow (C_f) \tag{26}$$

In this phase, the input file *f*, the data access policy (AP), and the public keys for all attributes are used as inputs as per Equation (26), which then produces the encrypted file C_f as an output.

$$\textit{RKeyGen ():}(PK_0, Pr_{ui}, P_{ui}) \rightarrow (RKey_{i \rightarrow j}) \tag{27}$$

In Equation (27), this phase generates the re-encryption key $(PK_{wi}, P_{uj}) \rightarrow j$ as an output by utilizing the public parameters and the private key pair.

$$\textit{TrapGen ():}(w_i) \rightarrow (T_{wi}) \tag{28}$$

This phase generates the corresponding trapdoor function by accepting the keyword as input in Equation (28). With the help of the trapdoor function, you can update your access policies effectively without downloading any data.

$$\textit{FDown ():}(T_{wi}, C^{\sim}) \rightarrow (F_{wi}) \tag{29}$$

3.4. Permitted Access to Data in Social Media Portals

The following procedures enable the followers and followees in the proposed method to perform authorized data access for end users.

DecKeyGen ():

$$(PK_0, MK_i, PK_a) \rightarrow (SK_{i,u}, SK_{i,u,a}) \tag{30}$$

In this phase, the TAA confirms that the user's attribute is part of the set of attributes under its management, as outlined in Equation (30), and generates the decryption key for the users.

Decrypt ():

$$(PK_0, CP, SK_{i,u}, SK_{i,u,a|a\epsilon p}) \rightarrow (f) \tag{31}$$

At this point, the data user is able to access the content in Equation (31) thanks to the TAA's management of the data decryption process. Only attributes that are in accordance with the access policies set by the data owner will allow the user to decrypt the data. The proposed attribute-based encryption system introduces an enhanced model featuring a single CSP alongside multiple TAAs. This model emphasizes efficient management of user attributes, where TAAs are tasked with defining and structuring these attributes. During system initialization, the CSP executes two primary algorithms: system setup and TAA creation. The system setup phase generates public system parameters and a confidential master key, while the TAA creation phase involves producing individual master keys for each TAA using these system parameters. This ensures the secure and efficient distribution and management of encryption keys within the network. Furthermore, the proposed system supports secure data sharing and search functionalities in social networks. Followers encrypt their data and share it with followers, utilizing algorithms such as *Encrypt (), KwEnc(),* and *TrapGen()* to ensure data confidentiality and access control. The encryption process is designed to protect data integrity while enabling keyword-based searches through trapdoor functions, which facilitate policy updates without accessing the actual data. Additionally, authorized data access is streamlined through a decryption key generation process managed by TAAs, allowing users to decrypt data only if they meet specific access policies. This comprehensive approach ensures robust security and efficient data management within social networks. The E-ABE framework demonstrates practical relevance in scenarios like a health-focused social network, “Health Connect”. Here, users share sensitive medical information with authorized individuals, such as doctors or support groups. The CSP securely stores encrypted data, while TAAs manage cryptographic keys based on user roles. By enforcing access policies and validating relationships through flow graph analysis, the framework ensures that only authorized individuals can access data, enhancing privacy, regulatory compliance, and user control.

4. Results and Discussions

This section analyses the developed mechanism to determine how well the suggested method works in cloud-based OSNs. The suggested approach is implemented using the Charm-Crypto library and the CryptoCC5.6.3 library, providing an extensible cryptographic technique framework. The hardware and software components of the experimental setup include an Intel Xeon processor, 8 GB of RAM, Ubuntu, and 1 TB of disk space.

4.1. Performance Metrics

Utilizing a variety of performance parameters like encryption time, time of generating the key, and decryption time, the experimental outcomes of the E-ABE approach for cloud-based privacy-preserving OSNs are assessed.

Key Generation Time (KGT): This metric quantifies the time required for the MHKCS algorithm to generate a key for a particular dataset. The formula is employed to determine it:

$$Key_{time} = file_{trns\_time} - file_{exe\_time} \tag{32}$$

Data Encryption Time (ET): This is the amount of time required for the data owner to encrypt the original data. The following is its source:

$$Data_{Encrypt\_time} = process_{time} - receive_{time} \tag{33}$$

Data Decryption Time (DT): This metric quantifies the time it takes the data owner to decrypt the encrypted data, as measured in milliseconds (ms). It is determined by employing the following formula:

$$Data_{Decrypt\_time} = process_{time} - receive_{time} \tag{34}$$

Communication Cost: This metric defines the cost associated with communication between the user and the Trusted Third Party (TPA) and between the cloud and the TPA. It is given by:

$$Commuination_{cost} = c(|s| + |p|) \tag{35}$$

As long as the size of the file set (s1, s2,...) is represented by |s| and c is the number of blocks that have been selected.sc, where |p| is the size of a file set element. The experiment was conducted to measure the computation time required for generating a secret key using the proposed model. The results indicate that the proposed model achieves a shorter computation time for secret key generation than other models, especially as the number of features increases. The performance of secret key generation computation time is presented in Table 1. The secret key generation time, measured in seconds, takes 19 seconds on average for 20 attributes, 24 seconds for 40 attributes, and 80 seconds for 100 attributes. Table 1 demonstrates that the computation time for secret key generation increases gradually with the number of user attributes. This increase is attributed to the presence of superfluous and redundant attributes. Despite a rise in the volume of data access requests, the suggested method demonstrates improved authorization accuracy, as illustrated in Figure 2. It maintains an accuracy of 96.5 percent regardless of the quantity of data access requests. The E-ABE method was evaluated using a varying number of users, and its accuracy values were measured.

Table 1. Computation time with respect to number of attributes

Number of Users Attributes	Computation Time (sec)
20	19
40	24
60	42
80	63
100	80

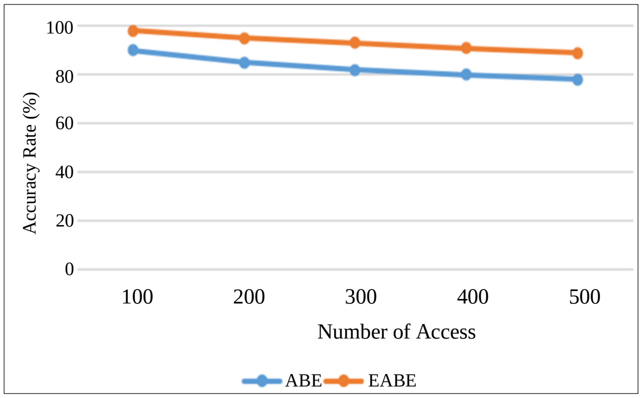

Fig. 2 Authorization accuracy comparison

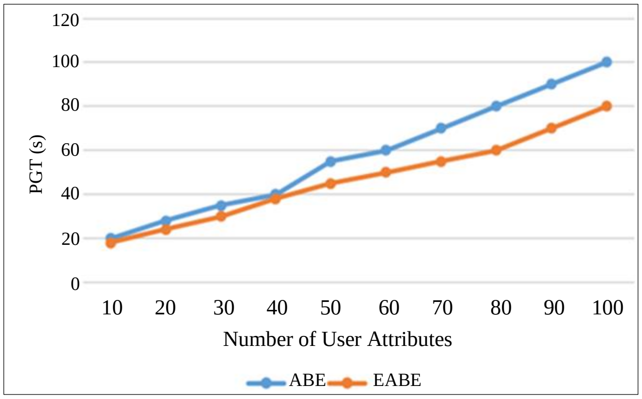

Fig. 3 Computational time of ABE and EABE

This method was compared with the existing ABE technique across different user counts. The proposed E-ABE technique demonstrated higher accuracy compared to the traditional ABE method. This improvement is attributed to the removal of unnecessary and redundant attributes. Applying the proposed method to a varying number of users consistently showed increased accuracy.

The E-ABE method achieved significantly higher accuracy rates than the existing ABE technique. Specifically, for 100 accesses, the E-ABE method achieved an accuracy of 96.5%, whereas the traditional ABE method only reached 90%. This enhancement in accuracy is due to the efficient elimination of redundant and unnecessary attributes within the E-ABE framework.

The private key generation time for the E-ABE is calculated for the various numbers of user attributes. This shows that the increasing number of user attributes increases the computation time of the private key generation. The computation time of the existing method for the 20 user attributes is around 25 seconds. Meanwhile, the computation time of the E-ABE method for the same is 15 seconds. The proposed method took less computation time for the private key generation for more number of user attributes. Figure 3 indicates the performance of the computation time of the secret key for different numbers of user attributes. The figure clearly shows that the increase in the computation time follows the increase in user attribute number. This is due to the consumption time taken for encrypting the plain text using the ABE algorithm and constructing indexes for large data blocks. Similarly, data file decryption is the main reason for the communication cost among members.

4.2. Evaluate Time Complexities of Data Contribution Time and Data Retrieval Time

The two main metrics used for comparison are the data access policy's number of user attributes and the amount of time needed for data contribution and retrieval actions. The simulation results suggest that the current ABE method contributes data in an average of 30 seconds when the data access policy includes 15 attributes. Conversely, the E-ABE method that has been proposed necessitates only 20 seconds to process the same number of attributes. The same applies to data retrieval with an access policy that includes 20 attributes; the existing ABE method averages 20 seconds, while the proposed E-ABE method averages only 15 seconds. This demonstrates that the E-ABE method significantly reduces the time required for both data contribution and retrieval compared to the existing systems, thereby enhancing efficiency and secure communication.

The proposed E-ABE method calculates the dependencies between decision attributes and conditional attributes. Attributes that lack dependencies are excluded from the system, ensuring that only highly relevant features are included for decision-making. Tests are made to suggest the E-ABE method with varying numbers of user attributes to see how long it takes to contribute data or encrypt data. For example, it takes 38 seconds to encrypt data for 50 user attributes. The time required to encrypt data grows directly proportional to the number of user attributes. In particular, 77 seconds is the time required to encrypt data for 80 user attributes.

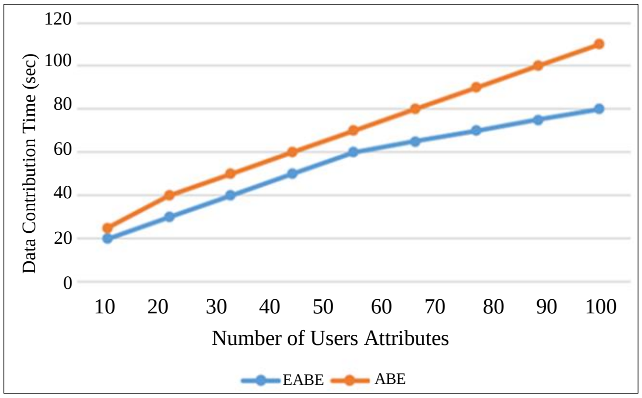

Fig. 4 Data contribution time comparison

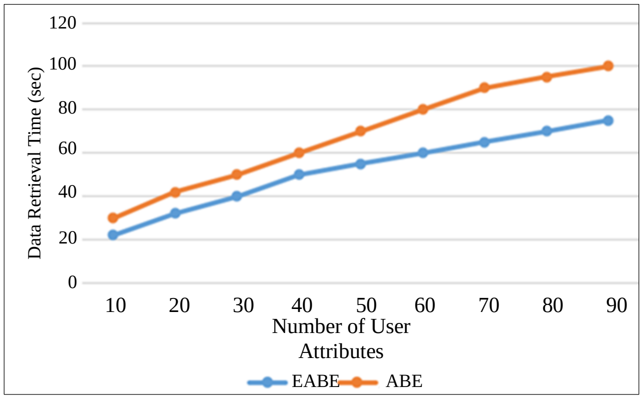

Fig. 5 Data retrieval time comparison

The data retrieval time, or data decryption time, for the proposed E-ABE method is also evaluated for different numbers of user attributes and is illustrated in Figure 5. The results indicate that the proposed E-ABE method has a reasonable data retrieval time across various numbers of user attributes. For example, the data retrieval time for 40 user attributes is 22 seconds.

4.3. Throughput Measures for Read and Write Operation

The performance of the E-ABE algorithm is evaluated by measuring its throughput. This model evaluates the system's performance based on the amount of requests it handles in a specific time period. Users simultaneously send their requests to the cloud servers, which consist of both read and write operations.

Initially, observations are conducted using read operations only. For read access, the ABE method can handle 100 users with an average throughput time of 30 seconds, whereas the E-ABE method completes the same task in just 25 seconds for every 100 users. The E-ABE method measures the throughput for write-only operations across varying numbers of users, as illustrated in Figure 7. For 100 users, the E-ABE method achieves a write operation throughput of 40 seconds, while the existing ABE method takes approximately 50 seconds for the same number of users.

It is evident that the throughput for write operations increases as the number of users rises. The combination of requests from read and write operations demonstrates that the existing ABE method yields an average throughput time ranging from 35 to 50 seconds per every 100 users. Conversely, the E-ABE method that has been proposed achieves 30–45 seconds for the same amount of users, which is the average throughput time. Despite an increase in the volume of user requests, the E-ABE algorithm maintains consistent performance across various operations. In contrast, the performance of current systems begins to deteriorate upon reaching a specific threshold. This indicates that the E-ABE method offers enhanced performance metrics, even in complex scenarios.

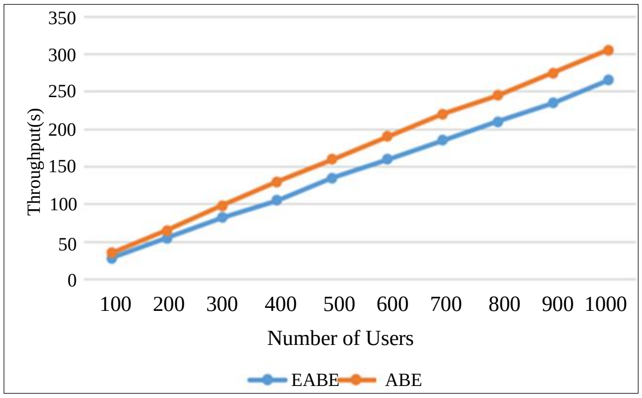

Fig. 6 Performance evaluation of read-only operations

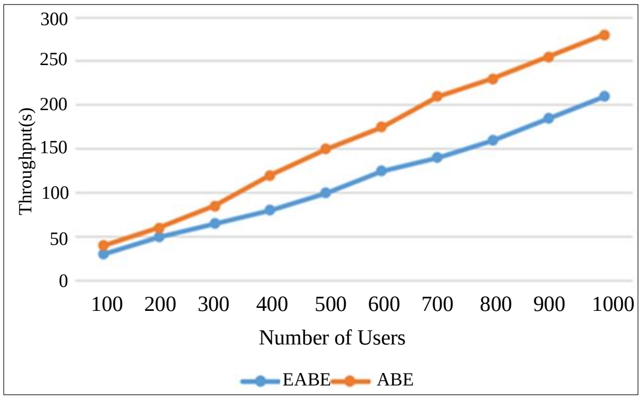

Fig. 7 A comparison of throughput for write-only operations

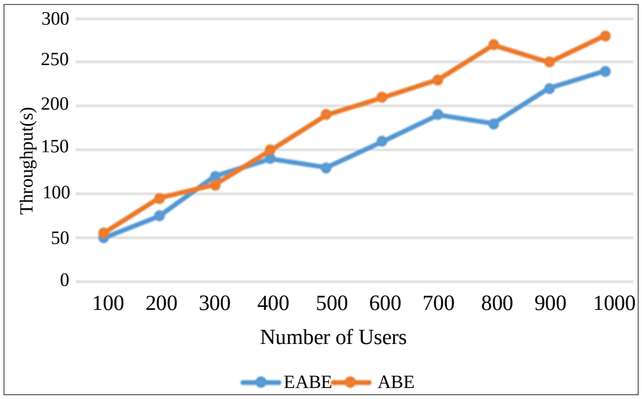

Fig. 8 Compare the throughput of the two read-only operations.

Experimental results clearly show that the E-ABE method outperforms the existing ABE method, providing 6% higher accuracy, 5 seconds less in key generation and data contribution times, and better performance in data retrieval and throughput times by less than 10 seconds.

Further improving its functionality compared to existing methods, the suggested E-ABE approach also allows for expressive hidden access policies, whether or not data users are specified.

Table 2. Comparative analysis of performance metrics between ABE and E-ABE frameworks

Metric	ABE	E-ABE	Improvement
Key Generation Time (s)	25	15	40% reduction
Encryption Time (s)	35	22	35% reduction
Decryption Time (s)	20	15	25% reduction
Authorization Accuracy (%)	90	96.5	6.5% increase
Throughput (req/s)	30	40	33% increase

The results of this study highlight the advantages of the proposed E-ABE framework over the existing ABE method, which are presented in Table 2. With significant reductions in computational time and higher authorization accuracy, the E-ABE framework addresses the critical challenges of efficiency and security in cloud-based social networks. These findings underscore its potential for broader applications in privacy-preserving systems.

5. Conclusion and Future Scope

In order to keep users' data safe in online social networking systems, this work provides a secure privacy-preserving method. System utility is the intended target of the proposed method. While enhancing data privacy, despite the fact that existing research has addressed various privacy concerns within OSNs. The OSN is modeled by the proposed method using a directed flow graph, which permits users to share sensitive information only if a directed edge exists between them. This innovative flow graph-based approach guarantees the validity of relationships between users. The main contributions are a flow graph-based method to verify user connections and a cloud-based OSN system using enhanced attribute-based encryption with multiple authorities for secure and efficient data processing. At last, attribute-based search encryption is employed to lessen the need for proxy re-encryption, which includes a trapdoor function. This saves time and effort by letting users change data access policies without constantly downloading the data. Users can streamline the process and enhance efficiency by modifying access policies associated with their data without downloading the actual content, as the proposed method allows for this.

The proposed framework can be further enhanced by integrating adaptive access control using machine learning to tailor policies based on user behavior. Exploring its application in decentralized cloud systems could improve scalability and eliminate single points of failure. Additionally, incorporating quantum-resistant encryption would strengthen security against future threats, while implementing the framework across domains like healthcare and finance would validate its real-world applicability.

References

[1] Srivastava Sonam, Mahesh Kumar Singh, and Yogendra Narain Singh, “Social Media Analytics: Current Trends and Future Prospects,” *Communication and Intelligent Systems, Lecture Notes in Networks and Systems*, Singapore, pp. 1005-1016, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[2] Jingwei Li et al., “Enabling Efficient and Secure Data Sharing in Cloud Computing,” *Concurrency and Computation: Practice and Experience*, vol. 26, no. 5, pp. 1052-1066, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[3] Asma Siddiqua et al., “A Review and Techniques in Smart Grid for Authentication of Messages,” *International Journal of Latest Engineering and Management Research*, vol. 3, no. 3, pp. 91-96, 2018. [Google Scholar] [Publisher Link]

[4] Zhiguang Qin et al., “A Survey of Proxy Re-Encryption for Secure Data Sharing in Cloud Computing,” *IEEE Transactions on Services Computing*, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[5] Ahsan Saud Qadri Syed et al., “A Chaotic Map-based Approach to Reduce Black Hole Attacks and Authentication Computational Time in MANETs,” *Engineering, Technology & Applied Science Research*, vol. 14, no. 3, pp. 13909-13915, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[6] Syed Shakeel Hashmi et al., “Data Hiding in the Multi-Cloud Environment by Product Cipher-Based Distributed Steganography,” *Research Square*, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[7] Michael Gertz, and Sushil Jajodia, *Handbook of Database Security: Applications and Trends*, Springer Science and Business Media, 1st ed., New York, 2007. [CrossRef] [Google Scholar] [Publisher Link]

[8] Syed Shakeel Hashmi et al., “Enhancing Data Security in Multi-Cloud Environments: A Product Cipher-Based Distributed Steganography Approach,” *International Journal of Safety and Security Engineering*, vol. 14, no. 1, pp. 47-61, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[9] Subashini Subashini, and Veeraruna Kavitha, “A Survey on Security Issues in Service Delivery Models of Cloud Computing,” *Journal of Network and Computer Applications*, vol. 34, no. 1, pp. 1-11, 2011. [CrossRef] [Google Scholar] [Publisher Link]

[10] Vipul Goyal et al., “Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data,” *Proceedings of the 13th ACM Conference on Computer and Communications Security*, New York, United States, pp. 89-98, 2006. [CrossRef] [Google Scholar] [Publisher Link]

[11] Bin Zhou, Jian Pei, and WoShun Luk, “A Brief Survey on Anonymization Techniques for Privacy-Preserving Publishing of Social Network Data,” *ACM Sigkdd Explorations Newsletter*, vol. 10, no. 2, pp. 12-22, 2008. [CrossRef] [Google Scholar] [Publisher Link]

[12] C. Atheeq et al. “Securing UAV Networks: A Lightweight Chaotic-Frequency Hopping Approach to Counter Jamming Attacks,” *IEEE Access*, pp. 38685-38699, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[13] Jin Li et al., "Identity-Based Encryption with Outsourced Revocation in Cloud Computing," *IEEE Transactions on Computers*, vol. 64, no. 2, pp. 425-437, 2013. [CrossRef] [Google Scholar] [Publisher Link]

[14] Arthur Sandor Voundi Koe, and Yaping Lin, "Offline Privacy Preserving Proxy Re-Encryption in Mobile Cloud Computing," *Pervasive and Mobile Computing*, vol. 59, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[15] Lujun Fang, and Kristen LeFevre, "Privacy Wizards for Social Networking Sites," *Proceedings of the 19th International Conference on World Wide Web*, pp. 351-360, 2010. [CrossRef] [Google Scholar] [Publisher Link]

[16] Elena Zheleva, and Lise Getoor, "To Join or Not to Join: The Illusion of Privacy in Social Networks with Mixed Public and Private User Profiles," *Proceedings of the 18th International Conference on World Wide Web*, pp. 531-540, 2009. [CrossRef] [Google Scholar] [Publisher Link]

[17] C. Atheeq et al., "Advancing IoT Cybersecurity: Adaptive Threat Identification with Deep Learning in Cyber-Physical Systems," *Engineering, Technology & Applied Science Research*, vol. 14, no. 2, pp. 13559-13566, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[18] Martin Hirt, and Kazue Sako, "Efficient Receipt-Free Voting Based on Homomorphic Encryption," *International Conference on the Theory and Applications of Cryptographic Techniques*, Springer Berlin Heidelberg, pp. 539-556, 2000. [CrossRef] [Google Scholar] [Publisher Link]

[19] Cynthia Dwork, "Differential Privacy," *International Colloquium on Automata, Languages, and Programming*, Springer Berlin Heidelberg, vol. 4052, pp. 1-12, 2006. [CrossRef] [Google Scholar] [Publisher Link]

[20] Rakesh Agrawal, and Ramakrishnan Srikant, "Privacy-Preserving Data Mining," *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, pp. 439-450, 2000. [CrossRef] [Google Scholar] [Publisher Link]

[21] C. Atheeq, and M. Munir Ahmed Rabbani, "Mutually Authenticated Key Agreement Protocol Based on Chaos Theory in Integration of Internet and MANET," *International Journal of Computer Applications in Technology*, vol. 56, no. 4, pp. 309-318, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[22] Barbara Carminati, Elena Ferrari, and Andrea Perego, "Enforcing Access Control in Web-Based Social Networks," *ACM Transactions on Information and System Security (TISSEC)*, vol. 13, no. 1, pp. 1-38, 2009. [CrossRef] [Google Scholar] [Publisher Link]

[23] Amardeep Singh, and Monika Singh, "Social Networks Privacy Preservation: A Novel Framework," *Cybernetics and Systems*, vol. 55, no. 8, pp. 2356-2387, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[24] Jiayi Chen et al., "Disclose More and Risk Less: Privacy Preserving Online Social Network Data Sharing," *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 6, pp. 1173-1187, 2018. [CrossRef] [Google Scholar] [Publisher Link]