


AN EFFICIENT CLOUD SECURITY SYSTEM USING DOUBLE SECRET KEY DECRYPTION TO PREVENT RANSOMWARE ATTACKS

*Nishanthi M.¹, Sanjay T.¹, Abhishek O.², Arthur Godson C.³, Bala Saravanan B.^{4**}*

1. Assistant Professor, Department of Artificial Intelligence and Data Science, Annai Vailankanni College of Engineering, Kanakumari-629401.
2. Student, Department of Artificial Intelligence and Data Science, Annai Vailankanni College of Engineering, Kanakumari-629401.
3. Student, Department of Artificial Intelligence and Data Science, Annai Vailankanni College of Engineering, Kanakumari-629401.
4.  Student, Department of Artificial Intelligence and Data Science, Annai Vailankanni College of Engineering, Kanakumari-629401.

ABSTRACT

Internet technology is advancing rapidly, enabling users to process, store, and share data with increasing efficiency. Cloud computing leverages shared infrastructure managed either internally or by third parties, where users store their data in encrypted formats. Attribute-Based Encryption (ABE) is a public-key encryption scheme that allows users to encrypt and decrypt data based on specific attributes associated with their identity. Access control for encrypted data in the cloud is enforced using access policies and attributes linked to private keys and ciphertexts. Existing ABE schemes face challenges due to costly decryption operations and complex access policies that scale with the number of attributes. This project addresses these issues by simplifying access policies into a single ciphertext. We use the security model of ABE with verifiable outsourced decryption, providing a verification key at the time of output decryption. Additionally, a user revocation scheme is implemented to address key leakage problems, and this approach is designed for real-time cloud environments. To mitigate the risk of ransomware attacks, we integrate anomaly detection algorithms within the cloud infrastructure. These algorithms continuously monitor user behavior, file access patterns, and system activities to identify suspicious or unauthorized activities indicative of a ransomware attack. Upon detection, the system isolates affected files or systems to prevent further encryption and notifies users and administrators for prompt action.

Keywords: Cloud Service, Attribute-Based Encryption, Ransomware, IoT, Verifiable Outsourced Decryption

1. INTRODUCTION

Cloud computing represents a paradigm shift in the way computing resources are provided and consumed. It involves a large pool of systems connected through private or public networks to deliver dynamically scalable infrastructure for application hosting, data storage, and file management. This technology significantly reduces the cost of computation, application hosting, content storage, and delivery, transforming traditional data centers from capital-intensive setups to variable pricing environments.

Cloud computing is grounded in the fundamental principle of "reusability of IT capabilities." It extends beyond traditional concepts such as grid computing, distributed computing, utility computing, and autonomic computing by broadening its scope across organizational boundaries. According to Forrester, cloud computing is defined as: "A pool of abstracted, highly scalable, and managed compute infrastructure capable of hosting end customer applications and billed by consumption."

This technology leverages the internet and central remote servers to manage data and applications, allowing consumers and businesses to use applications without the need for installation and to access their personal files from any computer with internet access. Cloud computing enhances efficiency by centralizing data storage, processing, and bandwidth. Common examples include email services like Yahoo Mail, Gmail, and Hotmail, which illustrate the practical application of cloud computing in everyday life.

Existing System:

Sensitive information shared and stored on third-party sites requires encryption to protect its confidentiality. Traditional encryption methods, however, often lead to coarse-grained access control, such as sharing personal keys with others, which can be impractical and insecure. To address this issue, Key-Policy Attribute-Based Encryption (KP-ABE) was proposed for fine-grained access control. In KP-ABE, attributes and personal keys are linked to access structures that manage which ciphertexts a user can decrypt. This method allows for more precise access control by hiding the attribute set under which the data is encrypted.

Ciphertext-Policy Attribute-Based Encryption (CP-ABE) extends this concept further. In CP-ABE, each secret key is associated with a set of attributes, while each ciphertext is linked to an access structure based on those attributes. Decryption is permitted only if the user's attributes satisfy the ciphertext's access structure. This approach provides fine-grained access management suitable for secure databases and multicast scenarios. CP-ABE improves efficiency with smaller ciphertexts and faster encryption/decryption operations by organizing attributes hierarchically, reducing the number of components required to represent attributes. This variant of ABE ensures security and allows for expressive access structures while maintaining constant ciphertext size.

Proposed System:

The proposed system enhances data security in cloud environments by integrating advanced Attribute-Based Encryption (ABE) with verifiable outsourced decryption and recoverability mechanisms. Building on ABE's flexible access control based on user attributes, the system introduces verifiability of decryption transformations, ensuring the correctness of data retrieval from the cloud. This enhancement allows for verification of decryption operations to maintain data integrity and security.

In addition to ABE, the system incorporates ransomware-specific security measures to mitigate the risk of ransomware attacks. This includes a dedicated ransomware detection module that utilizes machine learning algorithms and behavioral analysis techniques to identify and respond to ransomware threats in real time. File integrity monitoring, backup and recovery mechanisms, and isolation and containment measures are also implemented to protect against ransomware infections and minimize potential damage.

Furthermore, the system emphasizes user awareness and training to help users recognize and respond effectively to ransomware threats. By combining these ransomware-specific security measures with ABE and verifiable outsourced decryption, the proposed system provides a comprehensive approach to securing cloud-based data. It ensures data protection against unauthorized access, tampering, and ransomware attacks, thereby strengthening the resilience of cloud-based applications and services.

Module:

Cloud Entities:

Cloud computing involves using a network of remote servers to centralize data storage and provide online access to computer services or resources. Clouds can be classified into public, private, or hybrid types. The essence of cloud computing is to maximize the effectiveness of shared resources, which are dynamically reallocated based on demand. The system model in cloud computing typically consists of three main entities:

1. Cloud Server:

- **Responsibilities:** The cloud server is tasked with storing data in the cloud. It encompasses two sub-servers:
 - **Ciphertext Transformation Server (CTS):** This server handles the transformation and encryption/decryption of data.
 - **Cloud Storage Server (CSS):** This server manages the actual storage of encrypted data and handles data retrieval requests.

2. Data Owner:

- **Role:** The data owner is responsible for storing data in the cloud and can also download the data from the cloud. Data owners must have the appropriate authorization to manage their data effectively.

3. Cloud Users:

- **Function:** Cloud users access data stored in the cloud using attributes and adhere to the access control mechanisms in place. They are granted access based on the policies and permissions assigned to them.

Access Control Mechanism:

Access control is a fundamental policy or procedure that regulates who can access a system, what they can do with it, and how their actions are monitored. It is crucial for protecting computer security and can involve:

- **Mandatory Access Control (MAC):** Access rights are regulated by the system based on predefined policies, and users cannot change these rights.
- **Discretionary Access Control (DAC):** The owner of the data or resource has the discretion to grant or deny access to other users.
- **Role-Based Access Control (RBAC):** Access rights are assigned based on the roles of users within an organization, simplifying management and enforcing organizational policies.

These access control models are identity-based, focusing on the identification and authentication of users to determine access levels. Each model serves different needs and scenarios, providing a flexible framework to protect sensitive information and ensure that only authorized individuals can access specific resources.

Control Models:

In access control systems, both users (subjects) and resources (objects) are identified by unique names. Identification can be direct or through roles assigned to subjects. These access control methods are particularly effective in static distributed systems, where the set of users and their corresponding services are predefined and known.

Cloud Server Responsibilities:

The cloud server is tasked with distributing the global secret key and global public key to each authorized user within the system. The cloud storage server is split into two components:

- **Cloud Storage Server (CSS):** Responsible for storing encrypted data.
- **Ciphertext Transformation Server (CTS):** Handles the transformation and encryption/decryption processes.

The cloud server itself does not manage attributes or create secret keys associated with these attributes. The CTS is responsible for dividing the secret key into two parts:

- **Transformation Key (tk):** Maintained by the CTS and transferred between the user and the CTS server.
- **ElGamal-Type Secret Key (DK):** Kept secret on the user's side.

Security Model:

The cloud server stores data owned by users and provides access services. It generates decryption tokens for ciphertexts based on the secret keys issued by the CTS.

User Revocation:

User revocation is addressed by updating the system's master key components to exclude the attributes associated with the revoked user. This process involves:

1. Identifying a minimal set of attributes necessary for the user's access structure.
2. Updating these attributes to prevent the revoked user's access.

Challenges include significant computational overhead for data owners and the necessity for them to remain online to provide secret key updates. To mitigate these issues, a verifiable outsourced decryption approach is employed. This approach improves security by ensuring correct data decryption without requiring constant user interaction.

Verifiable Outsourced Decryption:

In the verifiable outsourced decryption approach, the Attribute-Based Encryption (ABE) ciphertext hides a symmetric session key. The attribute-based Key Encapsulation Mechanism (KEM) with outsourced decryption is defined similarly to ABE, but the encapsulation algorithm replaces the encryption algorithm and does not require a message input.

Data Encryption:

Data encryption involves:

- Dividing data into several components $m = m_1, \dots, m_n$ and generating the corresponding ciphertext.
- The server can then produce the correct decryption token based on these components.

In this model:

- Secret keys and global public keys are stored on the server, minimizing the need for data owners to submit secret keys for decryption token generation.

- Users with eligible attributes can decrypt the entire data stored in the cloud server. However, the system does not restrict data access to only authorized users, which means it cannot enforce strict access control beyond the attribute-based encryption.

Data Sharing

In the proposed system, the decryption algorithm utilizes public parameters, transformed ciphertext, and ciphertext for verification. The public parameters are defined as follows:

- $PK=(G,GT,e,g,u,v,d,ga,e(g,g)\alpha,T_i=gs_i=g\forall i,H)\text{PK} = (G, G_T, e, g, u, v, d, g^a, e(g, g)^\alpha, T_i = g^{s_i} = g \forall i, H)$
- $Ciphertext\ CT=(A,\rho,c^{\wedge},C^{\wedge}1,C',C1,i,D1,i,C^{\wedge}2,C'',C2,i,D2,i,i)\text{CT} = (A, \rho, \hat{c}, \hat{C}_1, C', C_{1,i}, D_{1,i}, \hat{C}_2, C'', C_{2,i}, D_{2,i}, i)$

Encryption Process:

1. **Data Encryption:** Data components are encrypted using different content keys $k=\{k_1,\dots,k_n\}k = \{k_1, \dots, k_n\}$ through symmetric encryption methods.
2. **Access Structure Definition:** An access structure M_iM_i is defined for each content key k_i and encrypted using the encryption algorithm Encrypt.

Secure Data Sharing

In secure data sharing:

- **User Assignment:** Each user is assigned a Ciphertext Transformation Server (CTS). Users can securely retrieve ciphertexts from the server.
- **Decryption Process:** To decrypt a ciphertext, users submit their secret key $TKTKTK$ issued by the CTS along with their stored key $DKDKDK$ at the time of requesting a decryption token. Upon receiving the decryption token, the user can decrypt the ciphertext using $DKDKDK$. Decryption is permitted only if the user's attributes satisfy the access policy defined in the ciphertext. The transformed ciphertext is represented as:

$$CT'=(T=C,T1=C1,T'=C'',T2'=C2)\text{CT}' = (T = C, T_1 = C_1, T' = C'', T_2' = C_2)$$

Key Revocation

In the key revocation module:

- **Revocation Handling:** When a user is revoked from a group, the system addresses the key revocation issue by updating the group key associated with each user. If a user changes the data owner, the group key is automatically updated.

- **Notification:** Key update reports are sent to all existing users in the group, ensuring that key leakage problems are mitigated.

Evaluation Criteria

The performance of the system is evaluated using the following metrics:

- **Storage Overhead:** This includes the storage of attributes, public keys, and secret keys for each user within the CTS. The storage overhead in our scheme scales linearly with the number of users in the system.
- **Communication Cost:** The communication cost for normal access control is consistent. For attribute revocation, the cost is linear to the number of ciphertexts containing the revoked attribute.
- **Computation Efficiency:** Computation efficiency is assessed based on the number of authorities and the number of attributes per authority, comparing both encryption and decryption processes.

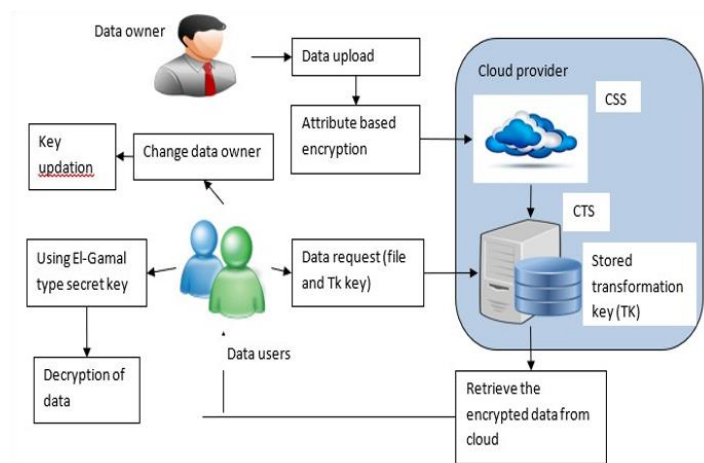
By examining these criteria, the efficiency and effectiveness of the proposed system in managing and securing data in cloud environments can be evaluated.

Advantages

1. **Enhanced Security:** The cloud security system using double secret keys ensures a robust protection mechanism against unauthorized access and tampering. This approach leverages two levels of encryption to secure data, improving overall security.
2. **Reduced Computation Time:** The proposed scheme significantly reduces the computation time required for resource-limited devices to recover plaintexts. By optimizing decryption processes and leveraging efficient algorithms, the system improves performance and resource utilization for devices with limited processing capabilities.

Architecture Diagram

Here's a simplified architecture diagram illustrating the components and interactions within the proposed system:



Conclusion

This project introduces a novel framework for fine-grained access control in sharing personal data, addressing the unique challenges posed by partially trustworthy cloud servers. By empowering users with complete control over their privacy through advanced encryption techniques, the framework significantly reduces the complexity of key management while enhancing privacy guarantees. Utilizing Attribute-Based Encryption (ABE) with verifiable outsourced decryption, this approach ensures that users can manage access not only for personal users but also for a variety of public users with diverse roles and qualifications.

The proposed system eliminates decryption overhead for users by leveraging attributes and guarantees secure data transformation and storage in the cloud. This robust attribute-based cryptographic technique offers a flexible and secure solution for encrypted data shared in cloud environments, ensuring both security and usability.

Future Enhancement

Future work will focus on extending the framework by integrating various advanced algorithms to further enhance security within cloud environments. This includes:

- **Behavior-Based Detection Algorithms:** To identify unusual patterns indicative of ransomware activity.
- **File Anomaly Detection Systems:** To detect unauthorized encryption attempts.
- **Decoy Files:** To act as traps and trigger alerts when accessed by ransomware.
- **Machine Learning:** For real-time threat analysis and faster response to emerging risks.
- **Automated Incident Response:** To isolate infected systems and initiate swift recovery processes.
- **Blockchain Technology:** To ensure the integrity of file records.
- **Zero-Trust Architecture:** To minimize lateral movement and restrict access.
- **Behavioral Analytics and Threat Hunting Tools:** To proactively seek out indicators of compromise.
- **Regular Data Backup Verification and User Awareness Training:** To fortify defenses.
- **Dynamic Access Controls:** To adjust permissions based on risk factors and limit ransomware impact.

By incorporating these enhancements, the system aims to further improve resilience against ransomware attacks and effectively safeguard digital assets in increasingly complex cloud environments.

References

- [1]. A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Advances in Cryptology* (Lecture Notes in Computer Science), vol. 3494, R. Cramer, Ed. Berlin, Germany: Springer-Verlag, 2005, pp. 457–473.
- [2]. V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proc. ACM Conf. Comput. Commun. Secur.*, 2006, pp. 89–98.

- [3]. J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. IEEE Symp. Secur. Privacy*, May 2007, pp. 321–334.
- [4]. B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in *Public Key Cryptography (Lecture Notes in Computer Science)*, vol. 6571, D. Catalano, N. Fazio, R. Gennaro, and A. Nicolosi, Eds. Berlin, Germany: Springer-Verlag, 2011, pp. 53–70.
- [5]. Y. Rouselakis and B. Waters, "Practical constructions and new proof methods for large universe attribute-based encryption," in *Proc. ACM Conf. Comput. Commun. Secur.*, 2013, pp. 463–474.
- [6]. L. Cheung and C. Newport, "Provably secure ciphertext policy ABE," in *Proc. ACM Conf. Comput. Commun. Secur.*, 2007, pp. 456–465.
- [7]. N. Attrapadung, J. Herranz, F. Laguillaumie, B. Libert, E. de Panafieu, and C. Ràfols, "Attribute-based encryption schemes with constant-size ciphertexts," *Theoretical Computer Science*, vol. 422, pp. 15–38, Mar. 2012.
- [8]. M. Green, S. Hohenberger, and B. Waters, "Outsourcing the decryption of ciphertexts," in *Proc. ACM Conf. Comput. Commun. Secur.*, 2011, pp. 343–352.