

# TRANSFORMATION OF ENCRYPTION WITH IDENTITY-BASED APPROACH FOR VERSATILE ENCRYPTED DATA SHARING IN PUBLIC CLOUD

1 MR. SNVASRK PRASAD, 2 B. TIMOTHY, 3 D. THARUN SAI

4 A. SHESHADRI, 5 G. NAGA SAI SATISH

*1 Assistant Professor, Department of CSE, Sri Indu College of Engineering and Technology-Hyderabad*

*2345 Under Graduate, Department of CSE, Sri Indu College of Engineering and Technology-Hyderabad*

## ABSTRACT

With the rapid development of cloud computing, an increasing number of individuals and organizations are sharing data in the public cloud. To protect the privacy of data stored in the cloud, a data owner usually encrypts his data in such a way that certain designated data users can decrypt the data. This raises a serious problem when the encrypted data needs to be shared to more people beyond those initially designated by the data owner. To address this problem, we introduce and formalize an identity-based encryption transformation (IBET) model by seamlessly integrating two well-established encryption mechanisms, namely identity-based encryption (IBE) and identity-based broadcast encryption (IBBE). In IBET, data users are identified and authorized for data access based on their recognizable identities, which avoids complicated certificate management in usual secure distributed systems. More importantly, IBET provides a transformation mechanism that converts an IBE ciphertext into an IBBE ciphertext so that a new group of users not specified during the IBE encryption can access the underlying data. We design a concrete IBET scheme based on bilinear groups and prove its security against powerful attacks. Thorough theoretical and experimental analyses demonstrate the high efficiency and practicability of the proposed scheme.

**Keywords: Cloud and Big Data, Data Security**

## INTRODUCTION

Cloud computing provides powerful and flexible storage services for individuals and organizations [1]. It brings about lots of benefits of sharing data with geographically dispersed data users, and significantly reduces local burden of storage management and maintenance. However, the concerns on data security and privacy are becoming one of the major obstacles impeding more widespread usage of cloud storage [2], since data owners lose physical control on their data after data are outsourced to cloud servers maintained by a cloud services provider (CSP). Data owners may worry about whether their sensitive data have been accessed by unauthorized users or malicious CSP. Cryptographic encryptions are widely suggested as standard approaches to protect the security and privacy of data outsourced to clouds [3]. With encryption mechanisms, data owners first encrypt their data and then outsource to cloud servers. Then the data in clouds are stored in ciphertext format and can only be accessed by the users having matching decryption keys. In a public cloud storage system, where different data owners may employ different encryption mechanisms according to their own data sharing requirements, it is often that a data owner wants to share his data with only one user and thus

encrypts the data to generate a particular ciphertext that can only be decrypted by the specific user. However, as data sharing requirement changes, the same data owner would like to share his data with more users, which, therefore, requires to transform the ciphertext format so that multiple users can decrypt. There are many scenarios in which the ciphertext transformation mentioned above is highly desirable. Consider a group of medical insurance agents draft a health insurance plan for a client. To do so, each agent needs to collect the client's personal information (e.g., electronic health records, occupations data, financial reports) from various data sources such as hospitals, employers, tax departments. The required data may be stored in remote cloud servers and especially, may be encrypted under different encryption mechanisms. To allow the agents to read and make use of the required data, a naive way is to let each agent acquire the corresponding decryption keys from the authorities who manage respective data. However, this would pose great concerns on data privacy. The authorities would ask a natural question: "If I give my decryption key to the agents, how to assure that all the agents would not leak the decryption key or use the decryption key to access other clients' stored data?" This paper attempts to solve such problem technically so that the authorities can transform the ciphertexts from one encryption system to another, without handing over their decryption keys. In particular, we consider an encryption transformation mechanism.

Data owner can securely outsource their data to a remote cloud server which is not fully trusted. The data are encrypted and stored in the server in IBE/IBBE ciphertext format so that only the users authorized by the data owners can access them. All users, including data owners and data consumers, are recognized with their unique identities, which avoids the usage of complicated public-key certificates.

- Cross-domain encryption transformation. Our IBET scheme achieves a cross-domain encryption transformation which can be viewed as a bridge connecting IBE and IBBE. In particular, a data owner (or an authorized data consumer) can transform the data stored in IBE ciphertext format into the data in IBBE ciphertext format, so that a set of users specified by the data owner (or the authorized data consumer) can simultaneously access the data.
- Strong security guarantee. Our IBET scheme achieves a strong security in the sense that:
  - 1) it can deter any unauthorized access to the data stored in the cloud server
  - 2) it can prevent leakage of some private information (e.g., private key) about the one who authorizes to transform encrypted data
  - 3) the transformation would not reveal any useful information about the sensitive data.We also conduct a series of experiments on our IBET scheme and make comparisons with some related schemes. The results show that the IBET scheme achieves a high performance in transforming the encrypted data, without incurring any significant computation costs to cloud clients or cloud servers.

**Applications.** Our IBET scheme can be applied to many real-world data sharing applications. First of all, the example of health records sharing described previously is an appropriate area where our IBET can be applied. Cloud-based encrypted email forwarding is another possible application. Imagine that several companies deploy their email systems on cloud servers. IBET can be used as a gateway to transform an encrypted email destined to an employee in one company into an encrypted email what can be received and decrypted by multiple employees in different companies. Vehicular ad-hoc network is also a potential application for IBET. When a car receives an encrypted report about front car condition or accident ahead and would like

further to broadcast the situation to rear vehicles, IBET can be used to directly transform the encrypted report into a broadcast ciphertext that allows multiple receivers to decrypt. Last but not least, in a mobile office environment, IBET may be utilized as a mobile application to securely share business data with a company director via a public cloud, and then transform the encrypted business data (if requested) so that the whole management team can access.

## LITERATURE SURVEY

Key-exposure resistance has always been an important issue for in-depth cyber defence in many security applications. Recently, how to deal with the key exposure problem in the settings of cloud storage auditing has been proposed and studied. To address the challenge, existing solutions all require the client to update his secret keys in every time period, which may inevitably bring in new local burdens to the client, especially those with limited computation resources, such as mobile phones. In this paper, we focus on how to make the key updates as transparent as possible for the client and propose a new paradigm called cloud storage auditing with verifiable outsourcing of key updates. In this paradigm, key updates can be safely outsourced to some authorized party, and thus the key-update burden on the client will be kept minimal. In particular, we leverage the third-party auditor (TPA) in many existing public auditing designs, let it play the role of authorized party in our case, and make it in charge of both the storage auditing and the secure key updates for key-exposure resistance. In our design, TPA only needs to hold an encrypted version of the client's secret key while doing all these burdensome tasks on behalf of the client. The client only needs to download the encrypted secret key from the TPA when uploading new files to cloud. Besides, our design also equips the client with capability to further verify the validity of the encrypted secret keys provided by the TPA. All these salient features are carefully designed to make the whole auditing procedure with key exposure resistance as transparent as possible for the client. We formalize the definition and the security model of this paradigm. The security proof and the performance simulation show that our detailed design instantiations are secure and efficient.

Secure search techniques over encrypted cloud data allow an authorized user to query data files of interest by submitting encrypted query keywords to the cloud server in a privacy-preserving manner. However, in practice, the returned query results may be incorrect or incomplete in the dishonest cloud environment. For example, the cloud server may intentionally omit some qualified results to save computational resources and communication overhead. Thus, a well-functioning secure query system should provide a query results verification mechanism that allows the data user to verify results. In this paper, we design a secure, easily integrated, and fine-grained query results verification mechanism, by which, given an encrypted query results set, the query user not only can verify the correctness of each data file in the set but also can further check how many or which qualified data files are not returned if the set is incomplete before decryption. The verification scheme is loose-coupling to concrete secure search techniques and can be very easily integrated into any secure query scheme. We achieve the goal by constructing secure verification object for encrypted cloud data. Furthermore, a short signature technique with extremely small storage cost is proposed to guarantee the authenticity of verification object and a verification object request technique is presented to allow the query user to securely obtain the desired verification object. Performance evaluation shows that the proposed schemes are practical and efficient

## SYSTEM ANALYSIS

### EXISTING SYSTEM

There are many scenarios in which the ciphertext transformation mentioned above is highly desirable. Consider a group of medical insurance agents draft a health insurance plan for a client. To do so, each agent needs to collect the client's personal information (e.g., electronic health records, occupations data, financial reports) from various data sources such as hospitals, employers, tax departments. The required data may be stored in remote cloud servers and especially, may be encrypted under different encryption mechanisms. To allow the agents to read and make use of the required data, a naive way is to let each agent acquire the corresponding decryption keys from the authorities who manage respective data. However, this would pose great concerns on data privacy.

### DISADVANTAGES

- The data owner has to rebuild the search index tree, which is time-consuming.
- Traditional solutions have to suffer high computational costs.

### PROPOSED SYSTEM

we try to answer the above question by studying encryption transformation between two different encryption systems. For the first time, we propose a novel notion called identity-based encryption transformation (IBET). We also define the notion (including algorithm definition and security model) of IBET. Then we design a concrete IBET scheme in bilinear groups, which provides the following attractive features.

**Identity-based data storage.** Data owner can securely outsource their data to a remote cloud server which is not fully trusted. The data are encrypted and stored in the server in IBE/IBBE ciphertext format so that only the users authorized by the data owners can access them. All users, including data owners and data consumers, are recognized with their unique identities, which avoids the usage of complicated public-key certificates.

**Cross-domain encryption transformation.** Our IBET scheme achieves a cross-domain encryption transformation which can be viewed as a bridge connecting IBE and IBBE. In particular, a data owner (or an authorized data consumer) can transform the data stored in IBE ciphertext format into the data in IBBE ciphertext format, so that a set of users specified by the data owner (or the authorized data consumer) can simultaneously access the data.

**Strong security guarantee.** Our IBET scheme achieves a strong security in the sense that:

- 1) it can deter any unauthorized access to the data stored in the cloud server;
- 2) it can prevent leakage of some private information (e.g., private key) about the one who authorizes to transform encrypted data;
- 3) the transformation would not reveal any useful information about the sensitive data.

### ADVANTAGES

**Data security protection:** If data have been encrypted before outsourced, then only the clients holding correct decryption keys can access (these clients are also called authorized clients).

The encrypted data are unreadable to CSP or unauthorized clients (those having no correct decryption keys).

Controllable transformation: Only the files specified by the data owner in the authorization token can be transformed by CSP. CSP and other clients cannot cooperatively deduce a valid authorization token in order to transform unspecified files, nor detect sensitive information about the data encrypted in unspecified files.

## IMPLEMENTATION

### MODULE DESCRIPTION

#### 1. DATA OWNER

Is an entity who encrypts its documents under an arbitrary access control policy and outsources them to the cloud. He/She considers the time of encrypting in generating the cipher texts. We should highlight that the data owner also encrypts his/her documents under his/her arbitrary access control policy. However, in this paper we concentrate on the encryption of the extracted keywords from documents.

#### 2. DATA USER

Is an entity who is looking for documents which contains an intended keyword, and are encrypted in a determined time interval. The time interval is arbitrarily selected by the data user.

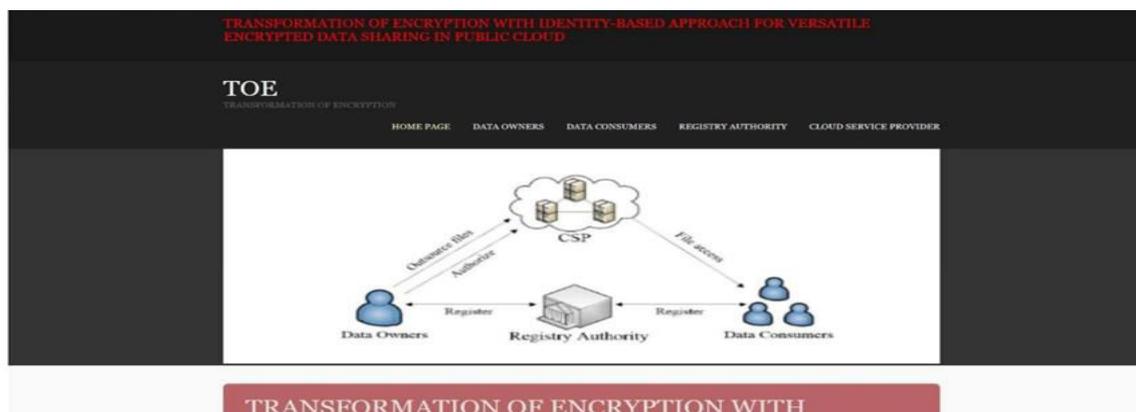
#### 3. CLOUD SERVICE PROVIDER

Is an entity with powerful computation and storage resources. CS stores a massive amount of encrypted data, and receives the search tokens to look for the required documents on behalf of the data user. The cloud finds the relevant documents, and sends them back to the data user.

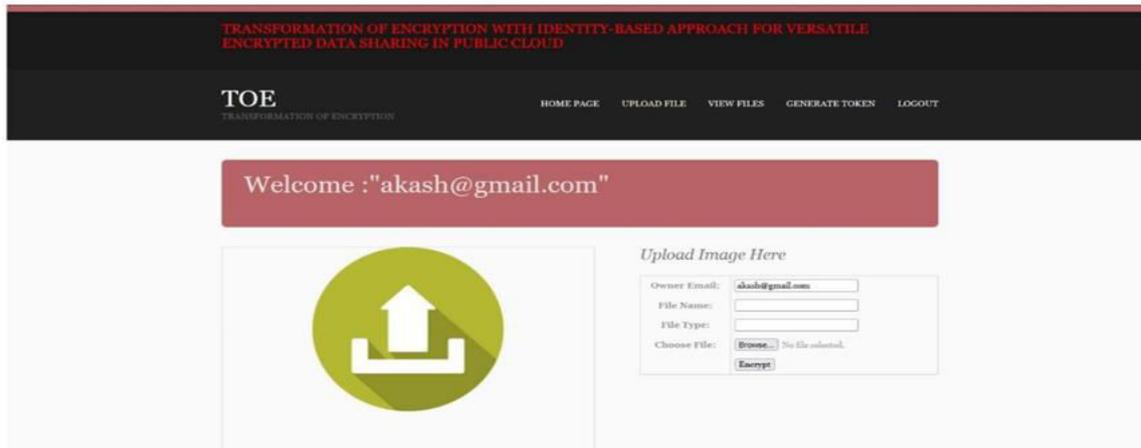
#### 4. REGISTRY AUTHORITY

Is a fully trusted entity who receives each user's access tree, and generates their secret keys corresponding to his/her attributes set presented in his/her access tree. Then, the TTP sends back the users' credentials through a secure and authenticated channel.

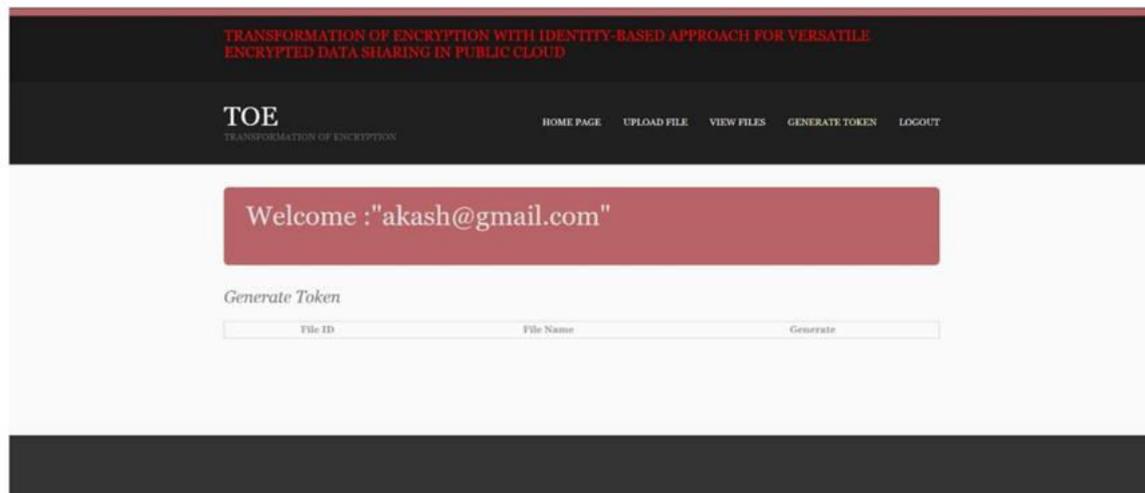
## .RESULTS



### HOME SCREEN



### FILE UPLOADING PAGE



### GENERATE TOKEN PAGE

## CONCLUSION

In this paper we studied how to securely and efficiently transform encrypted data in clouds. To address this issue, we proposed an identity-based encryption transformation (IBET) model, which connects the well-studied IBE and IBBE systems. IBET allows data owners to secure outsourced data with identity-based access control, which eliminates complicated cryptographic certificates for all users. Moreover, IBET provides a transformation mechanism for data owners to authorize cloud service provider (CSP) to transform a file in IBE-ciphertext format into a file in IBBE-ciphertext format, so that a set of authorized users can access the underlying data. We proposed a concrete IBET scheme that is secure against powerful attacks. Thorough experimental analyses demonstrate the efficiency and practicability of the scheme.

## REFERENCES

- [1] D. Song, E. Shi, I. Fischer, and U. Shankar, "Cloud data protection for the masses," *Computer*, vol. 45, no. 1, pp. 39–45, 2012.

- [2] J. Yu, K. Ren, and C. Wang, "Enabling cloud storage auditing with verifiable outsourcing of key updates," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 6, pp. 1362–1375, 2016.
- [3] H. Yin, Z. Qin, J. Zhang, L. Ou, and K. Li, "Achieving secure, universal, and fine-grained query results verification for secure search scheme over encrypted cloud data," *IEEE Transactions on Cloud Computing*, 2017.
- [4] K. Li, W. Zhang, C. Yang, and N. Yu, "Security analysis on one-to-many order preserving encryption-based cloud data search," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 9, pp. 1918–1926, 2015.
- [5] R. Zhang, R. Xue, and L. Liu, "Searchable encryption for healthcare clouds: a survey," *IEEE Transactions on Services Computing*, vol. 11, no. 6, pp. 978–996, 2018.
- [6] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," *SIAM Journal on Computing*, vol. 32, no. 3, pp. 586–615, 2003.
- [7] J. Wei, W. Liu, and X. Hu, "Secure data sharing in cloud computing using revocable-storage identity-based encryption," *IEEE Transactions on Cloud Computing*, 2016.
- [8] D. He, N. Kumar, H. Wang, L. Wang, K.-K. R. Choo, and A. Vinel, "A provably-secure cross-domain handshake scheme with symptoms-matching for mobile healthcare social network," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 633–645, 2018.
- [9] C. Delerablé, "Identity-based broadcast encryption with constant size ciphertexts and private keys," in *International Conference on the Theory and Application of Cryptology and Information Security*. Springer, 2007, pp. 200–215.
- [10] H. Deng, Q. Wu, B. Qin, W. Susilo, J. Liu, and W. Shi, "Asymmetric cross-cryptosystem re-encryption applicable to efficient and secure mobile access to outsourced data," in *Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security*. ACM, 2015, pp. 393–404.