International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

A Survey on Asymmetric Key Aggregate Encryption in Data Sharing

Shilpashree P[1], Dr. K. N. Narasimha Murthy[2]

[1]PG Scholar, Department of Computer Science and Engineering,
Vemana IT, Visvesvaraya Technological University, Belagavi, Karnataka, India.

[2]Professor, Department of Computer Science and Engineering,
Vemana IT, Visvesvaraya Technological University, Belagavi, Karnataka, India.

Abstract: ***Data sharing is the ability to share the same data resource with multiple applications or users. The data can be shared with others in a secure, efficient and economic way. Internet and cloud storages are rapidly growing day by day. So sharing of sensitive data through internet and cloud storage is an issue because there is a possibility of third party can steal the sensitive and confidential data. To overcome this problem data should be encrypted before sent across internet. Encryption is a technique of converting plaintext to ciphertext. Asymmetric is a kind of encryption which helps to share a data in a public key encryption, here the encryption key and decryption key are different. In this paper, the concept of aggregator key is introduced, which collects all the keys generated by the data owner and made it as a compact key. This compact key has consists of number of keys and using this aggregate key the data consumer can access only the data who is authenticate.***

Keywords: Asymmetric, Cloud storage, Data Sharing, Encryption, Key Aggregate

1. Introduction

Data sharing is a kind of functionality in cloud storage and it is very essential. In the cloud storage the data can be shared with others with secured,efficient and economic way. Data sharing canbe used for scholarly research, other researchers and investigators. Security is the major issue in data sharing through internet. Without security the sensitive and confidential data may be captured by unauthorised users.It implies that the data can be stored in one or more servers. There is some mechanism for software locking that avoids the people to change the same set of data at the same time.

So, to share the data one must encrypt the data and then send to the others in the secured manner, so no one apart from the authorised consumer can get the data very easily. Encryption is a process of crypto graphing the information or the messages in such a way that only the parties who have authority can read it[9]. There are two types of encryption keys they are – symmetric key and asymmetric key or public key. In symmetric encryption, if the data owner wants his/her data to be originated from the third party then he/she has to give their secret key. And this is not always possible all the time because the scheme provides no guarantee on the correctness of the transformation done by the cloud server. In the cloud computing setting, cloud service providers may have strong financial incentives to return incorrect answers, if such answers require less work and are unlikely to be detected by users.

Thus communicating parties must have the same key for the secret communication. While in asymmetric the encryption key and decryption key are different in asymmetric or public key encryption. For example every employee in an enterprise can upload the encrypted data without having the knowledge of the master-secret key of the company.

Asymmetric key cryptography[10] and it is also known as public-key cryptography, is a kind of cryptographic algorithms and it requires two separate keys, one is for secret also called private and other is called public. Even though the both keys are different but it's key pair are mathematically linked. The public key is the one which encrypts plaintext or which is used for verifying digital signature. Whereas the private key is the one used for decryption of ciphertext or can be used to create digital signature.

The term Asymmetric originates from the use different keys to achieve the opposite functions like inverse of the other as reverse with symmetric cryptography which depends on the same key to perform both the functions.The public key may be issued without compromising the security, whereas the private key must not be published to anyone means not to perform digital signatures. Asymmetric key algorithms do not require a initial exchange of one or more secret keys for security between the parties.

Message authentication process involves a message with a key which is private to produce a digital signature. The verification of the signature can be done by anyone by processing the value of the signature with corresponding public key of signer's and comparing the result with the message. Since it was signed, it confirms the message is unmodified and confident the signer's private key to the signer has remained secret by that it shows the signer and no one else, performs the signature operation intentionally.

Public-key cryptography applications are found in IT security discipline information security. From unauthorised access the Information Security (IS) is the practice of contending information from unauthorised access, modification, inspection, disclosure, use, and recording. Against security threats, ISi.e., Information Security is concerned with all aspects of protecting electronic

Licensed Under Creative Commons Attribution CC BY

information. To satisfy the confidentiality, non-reputability and authenticity of electronic communications and data storage the public-key cryptosystem is used.

The encryption key is published for anyone to use and encrypt the messages in public-key encryption. The receiving party is the one to access the decryption key that enables the messages to be read. Before 1973 all encryption schemes were symmetric-key also called private key then later public key encryption was described.

In cryptosystems, applications and protocols the public-key algorithms are fundamental security ingredients. There are various internet standards such as Transport Layer Security (TLS), S/MIME, GPG and PGP. Some of the public key algorithms can produce secrecy and distribution of key. Example is Diffie–Hellman key exchange , some provide digital signatures example is Digital Signature Algorithm, and some produce both example like RSA.

A Digital Signature is a mathematical scheme for acknowledging the authenticity of a document or a digital message. A valid digital signature is the one which makes the recipient to believe that the message was created by a known sender, Such that the sender cannot excludes the sent message and it ensures that the message was not altered in the transit. Digital signatures can be used to detect tampering or forgery and in software distribution.

Encryption does not itself prevent interception, but it excludes the content of the message to the interceptor. In the encryption technique, the information or the message is referred to as plaintext and it is encrypted using an encryption algorithm. Generating ciphertext, it is a kind of scramble message, if it is decrypted then onlyit's able to read. The encryption scheme uses a pseudo-random encryption key usually and it is generated by the algorithm for some technical reasons.

It is possible to decrypt the message without acquiring the key but, large computational resources are required for well-designed encryption scheme. Unauthorised interceptors cannot decrypt the message but, with the key an authorized recipient can easily decrypt the message.

An aggregate key is a kind of key that consists of two or more attributes that identify an entity occurrence uniquely. If a key has only one attribute then it is called simple attribute. Attribute keys may be confident of other simple keys which are unique and non-key attributes, but, another compound key may not included.

A composite key is the one which consists of at least one attribute key and one compound key. Composite keys may also include non-key attributes and simple keys. The entity is the one which has a module code and student ID as its primary key. An example may each student is attending at university be an entity that represents the modules. Each of the attributes is used to make up the simple keys and primary keys because each key represents a reference that is unique when identifying a student in one kind of instance and a module in the other.

So the paper is organised as follows:In the section 2, related work and their contributions are discussed.In the section 3, compare of the results are discussed.

2. Related Work

J. Benaloh et al. [3] has conducted the experiments to securely share the data example like patient controlled encryption. In an electronic system health record, medical devices and healthcare providers can upload the patient health records and when it is necessary they can retrieve and view them at later time. Furthermore, patients may have the chance to delegate the rights to access and allow friends, family and designated healthcare providers to edit or view of their record. Patients and their delegates may perform searches efficiently in an skilful manner over part or all of the record.

They consider the problems that arise in a naive attempt to add security to such a system. They argue above that they want to allow the patient to produce their own decryption key. But here in this case, how the patient can allow others to access their record is a question. So, clearly their does not want to give their entire key, because if other recipient who got their entire key can modify or read all the parts of her record.

The patient can grant access to a category easily and even without knowing what types of files are already exists that might ultimately be included in it. But, the hierarchy is fixed in that there is only one way in which they can partition the record. If they want to give out the access rights based on something else like example based on document type or sensitivity of data, they have to take care of all the low level categories involved, and they has to provide a separate decryption key for each. Example like giving a access to a lab report to all X-rays would require giving separate keys for Cardiologic X-rays, Dental X-rays and Mental Health X-rays.

A.Sahai et al. [5] conducted the experiments to securely share the data example like Attribute-Based Encryption for Fine-Grained Access Control of Encrypted data. Encryption of the data is needed whenever the more sensitive data is shared and stored by third-party on the internet. The drawback of encrypting data is that it can be share to another party then the data owner has to give the private key, this level is called coarse grained level. They develop a unique cryptosystem for fine grained sharing of data that is encrypted and that they call it as Key-Policy Attribute-Based Encryption (KP-ABE). In their cryptosystem, the message or the ciphertexts are labelled with sets of attributes and the private key are with access structures that restrict which ciphertext a user is able to decrypt.

They demonstrate the applicability of their construction to sharing of audit-log information and broadcast encryption. There construction supports the delegation of private keys which classify Hierarchical Identity-Based Encryption (HIBE).They develop a much richer type of attribute-based encryption cryptosystem and demonstrate its applications. In their system each ciphertext is labelled by the encryptor with a set of descriptive attributes. Each private key is combined

Licensed Under Creative Commons Attribution CC BY

with an access structure that specifies which type of ciphertexts the key can decrypt. They call such a scheme a Key-Policy Attribute-Based Encryption (KP-ABE), since the access structure is specified in the private key, while the ciphertexts are simply labelled with a set of descriptive attributes.

M. J. Atallahet al.[2] has conducted the experiments to securely share the data, example like Dynamic and Efficient Key Management for Access Hierarchies. Whenever the user population can be modelled as a set of ordered classes which are partial represented like directed graph at this time the hierarchies arise in the context of access control. If a user with access privileges for a class obtains access rights to objects stored at that class and also all descendant classes of hierarchies.The key management problem for such hierarchies later consists of assigning a key in the hierarchy for such classes so that keys for descendant classes can be obtained through efficient key derivation. They propose a solution to this problem with the following properties: (A) with a number of symmetric-key operationsbounded by the length of the path between the nodes,each node can derive the key of any of its descendant; (B) Against collusion the scheme is provably secure; (C) Locally the updates are handled in the hierarchy.

F. Guo et al. [4] conducted the experiments to securely share the data example like Multi-Identity Single-Key Decryption without Random Oracles. Multi-Identity Single-Key Decryption (MISKD) is an Identity-Based Encryption (IBE) system where a private decryption key can map multiple public keys (identities).In MISKD more exactly a single private key can be used to decrypt n number of ciphertext encrypted with different public keys associated with the private key. of MISKD was recently introduced by Guo, Mu and Chen and they proposed a concrete MISKD scheme and proved its security based on the Bilinear Strong Diffie-Hellman problem (q-BSDH) in random oracle model. They present a novel MISKD scheme that is provably secure in the selective-ID model and it is based on the Decisional Bilinear Diffie-Hellman (DBDH) assumption. There scheme gives more efficient in decryption.

Q. Zhang et al.[6] conducted the experiments to securely share the data, example like"A Centralized Key Management Schemefor Hierarchical Access Control". Users often have different access rights to multiple data streams in group communication. Users can form partially ordered relations, and data streams can formpartially ordered relations and it is based on the access relation of users. They proposed a key management scheme for hierarchical access control, which considers both partially, ordered data stream relations and partially ordered user relations. They also proposed an algorithm for logical key graph construction and this is suitable for users and data streams which have complex relations. Their schemes can improve the efficiency significantly of key management as shown by simulation results.

D. Boneh et al. [7] conducted the experiments to securely share the data, example like Chosen Ciphertext Security from Identity-Based Encryption.They proposed a simple as well as efficient construction of chosen-ciphertext attacks (CCA) for secure public-key encryption scheme from any chosen-plaintext attacks(CPA) secure identity-based encryption (IBE) scheme.Their construction requires the IBE scheme to satisfy a weak/relative notion of security and it is known to be achievable without random oracles. Thus in their standard model it results to provide a new approach for constructing CCA-secure encryption schemes. Their scheme is quite different because it avoids non-interactive proofs of well-forkedness. Moreover, applying their conversion to some proposed IBE scheme recently results in CCA secure schemes whose makes them quite practical recently. Their construction stretch to give a efficient and simple method for securing any binary tree encryption (BTE) scheme and it is against adaptive chosen-ciphertext attacks. Thus it returns efficient CCA secure hierarchical identity based and forward-secure encryption schemes.

X. Boyen et al. [8] conducted the experiments to securely share the data, example like Hierarchical Identity BasedEncryption with Constant Size Ciphertext. They present a Hierarchical Identity Based Encryption (HIBE) system where the ciphertext consists of three element group and decryption requires two bilinear map computations only. And it is disregarding of the hierarchy depth.Encryption is as efficient as in other HIBE systems.In the standard model they have proved that their scheme is selective ID secure and in the random oracle model it is fully secured. Their system has more applications and gives efficient forward secure public key and identity based cryptosystems with short ciphertexts and it converts the NNLbroadcast encryption system into an efficient public key broadcast system. And it provides mechanism for encryption to the future efficiently. Their system also supports limited delegation where users can be given restricted private keys that allow only delegation to bounded depth. The HIBE system can be modified to support sub linear size private keys at the cost of some ciphertext expansion.

3. Comparision Analysis of Previous Work

In this section comparison analysis is done on decryption key size, ciphertext size and encryption type.

M. J. Atallah et al, selected decryption key size as most likely non-constant(it depends on the hierarchy). He also took ciphertext size as constant and encryption type as symmetric or public key. J. Benaloh et al, selected decryption key size as constant. He took that ciphertext size as constant and the encryption type as symmetric key.Z. Chenet al, selected decryption key size as constant. He took that ciphertext size as non-constant and the encryption type as public key.B. Waters al, selected decryption key size as non-constant. He took that ciphertext size as constant and the encryption type as public key.

There exist several expressive ABE schemes where the decryption algorithm only requires a constant number of pairing computations. Recently, Green et al. proposed a remedy to this problem by introducing the notion of ABE with outsourced decryption, which largely eliminates the decryption overhead for users. Based on the existing ABE schemes, Green et al. also presented concrete ABE schemes with outsourced decryption.

Licensed Under Creative Commons Attribution CC BY

In KAC, users encrypt a message not only under a public-key, but also under an identifier of cipher text called class. That means the cipher texts are further categorized into different classes. The key owner holds a master-secret called master-secret key, which can be used to extract secret keys for different classes. More importantly, the extracted key have can be an aggregate key which is as compact as a secret key for a single class, but aggregates the power of many such keys, i.e., the decryption power for any subset of cipher text classes.

4. Conclusion

The conclusion of this paper is as follows. To share the data in a secure way, the required tool is encryption. Asymmetric key encryption is a more secure way to share the data than symmetric key encryption because it uses two keys, public key and private key. Among these two keys, either of the keys is used for encryption and decryption.

References

[1] L. Hardesty, "Secure computers aren't so secure," MIT press, 2009, http://www.physorg.com/news176107396.html.
[2] M. J. Atallah, M. Blanton, N. Fazio, and K. B. Frikken, "Dynamic and Efficient Key Management for Access Hierarchies," ACM Transactions on Information and System Security (TISSEC), vol. 12, no. 3, 2009.
[3] J. Benaloh, M. Chase, E. Horvitz, and K. Lauter, "Patient Controlled Encryption: Ensuring Privacy of Electronic Medical Records," in Proceedings of ACM Workshop on Cloud Computing Security (CCSW '09). ACM, 2009, pp. 103–114.
[4] F. Guo, Y. Mu, Z. Chen, and L. Xu, "Multi-Identity Single-Key Decryption without Random Oracles," in Proceedings of Information Security and Cryptology (Inscrypt '07), ser. LNCS, vol. 4990. Springer, 2007, pp. 384–398.
[5] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted data," in Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS '06). ACM, 2006, pp. 89–98.
[6] Q. Zhang and Y. Wang, "A Centralized Key Management Scheme for Hierarchical Access Control," in Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '04). IEEE, 2004, pp. 2067–2071.
[7] D. Boneh, R. Canetti, S. Halevi, and J. Katz, "Chosen-Ciphertext Security from Identity-Based Encryption," SIAM Journal on Computing (SIAMCOMP), vol. 36, no. 5, pp. 1301–1328, 2007.
[8] D. Boneh, X. Boyen, and E.-J. Goh, "Hierarchical Identity Based Encryption with Constant Size Ciphertext," in Proceedings of Advances in Cryptology - EUROCRYPT '05, ser. LNCS, vol. 3494. Springer, 2005, pp. 440–456.
[9] "Encryption Basics | EFF Surveillance Self-Defense Project." Encryption Basics | EFF Surveillance Self-Defense Project. Surveillance Self-Defense Project, n.d. Web. 06 Nov. 2013. <https://ssd.eff.org/tech/encryption>.
[10] "Public-Key Encryption - how GCHQ got there first!". gchq.gov.uk. Archived from the original on May 19, 2010.

Author Profile



Dr K N Narasimha Murthy is presently working as a professor in the Department of Vemana Institute of Technology Banglore. He has published more than 20 research papers and guiding three PhD students in the area of Computer Science and Engineering. His area of interests is Cloud computing, Image processing and Networking.



Shilpashree P is a presently a PG scholar at Vemana Institute of Technology. Her fields of interest is Cloud computing, Computer Networking and Operating Systems.

Licensed Under Creative Commons Attribution CC BY