A Survey on Secure Block Storage and Access Control Using Big Data Environment

K. Jamuna Rani
Assistant Professor
Computer Science and Engineering
Auroras Technological and research Institute
jamunarani27@gmail.com

T. Soumya
Assistant Professor
Computer Science and Engineering
Auroras Technological and research Institute
soumyasritatikonda@gmail.com

V. Madhavi
Assistant Professor
Computer Science and Engineering
Auroras Technological and research Institute
Madhavi.vagu@gmail.com

***Abstract*:** Over past few years, the amount of data being collected continuous to grow more and more companies are building Big Data repositories to store , aggregate and extract meaning from this data and securing Big Data comes main challenge. This paper presents the comparison of different encryption based algorithms i.e. Key Management for Access Control , Attribute-Based Access Control, Attribute-Based Encryption (ABE), Key Policy Attribute-Based Encryption (KP-ABE), Cipher text -Policy Attribute-Based Encryption (CP-ABE) and cryptography for security and access control, its real time applications. This comparison results cannot provide flexibility and efficiency for data analysis. The future scope of this survey on big data can be discussed by using access control algorithm

Keywords: *Big data, Key Management for Access Control ABE, KP-ABE, CP-ABE, Data Privacy, data analysis.*

*****

I. INTRODUCTION

Big data is a collection of large datasets that cannot be processed using traditional computing techniques. Big data contains various tools, techniques and frame works. Big data contains the data produced by different devices and applications.

The devices are

- **Black Box Data:** It is a component of helicopter, airplanes, and jets, etc. It captures voices of the flight crew, recordings of microphones and earphones, and the performance information of the aircraft.
- **Social Media Data:** Social media such as Facebook and Twitter hold information and the views posted by millions of people across the globe.
- **Stock Exchange Data:** The stock exchange data holds information about the 'buy' and 'sell' decisions made on a share of different companies made by the customers.
- **Power Grid Data:** The power grid data holds information consumed by a particular node with respect to a base station.
- **Transport Data:** Transport data includes model, capacity, distance and availability of a vehicle.
- **Search Engine Data:** Search engines retrieve lots of data from different databases.

Big data includes of three types of Data i.e. structured data which has Relational data, semi structured data which has XML data, and unstructured data includes Word, PDF, Text and Media Logs.

The major challenges associated with big data are Capturing data, Curation, Storage, Searching Sharing, Transfer, Analysis, and Presentation

II. RELATED WORK

The primary goal of these techniques is to enforce access control to data stored in potentially un trusted repositories. That is, we want to give authorized parties access to the data they need while ensuring that unauthorized parties, either outsiders trying to gain access or malicious insiders in the organization managing the repository, cannot access sensitive data. In this section we focus on systems where data is stored in blocks that are stored and retrieved by a unique identifier, such as in a file system. In such systems, we want authorized parties to be able to retrieve data by its identifier, but do not need to enable complex search queries to retrieve subsets of the data.

Comparison of All Encryption Techniques

Key Management for Access Control

Key management includes generating and distributing cryptographic keys to system users in such a way that only authorized parties have the necessary keys to decrypt sensitive data. Most modern systems include some form of key management for controlling access to data in this way and there are many commercially available, standardized solutions for generating and managing keys. These typically use a trusted key management server to manage all keys in the system and to distribute the necessary keys to authorized parties. Here, we instead focus on a cryptographic technique called broadcast encryption or group keying which allows a data owner to encrypt data to a designated set of recipients without having to rely on a trusted key manager. This is particularly important in big data applications where the storage may be handled by an un trusted repository on which a trusted key manager may not be available [1].

Broadcast encryption or group keying is a cryptographic technique for establishing crypto-graphic keys shared among a designated set of parties, thus giving these authorized parties access to encrypted data. Specifically, broadcast encryption gives a way to establish a crypto-graphic key such that all authorized parties receive the key and all unauthorized parties have no information about the key. In particular, even if some number of unauthorized parties colludes, they should not be able to learn data that none of them is individually authorized to learn. An important goal of this primitive is to minimize the total size of encrypted data

Imagine that a data owner wishes to share data with some subset of the users of a system. One trivial solution is to have him share a different cryptographic key with each of the recipients and separately encrypt the data to each of them. However, this requires the data owner to store a large number of keys, and also the size of the encrypted data grows linearly in the number of recipients. Broadcast encryption gives techniques to achieve this functionality without incurring these costs in key and data storage.

Single Sender Broadcast Encryption: Broadcast encryption was first considered in a setting where there is one data owner who wants to share data with a set of authorized recipients [2]. This construction was able to achieve much shorter keys and cipher texts when the number of unauthorized users is small. This was further improved by a protocol be able to handle an arbitrary number of unauthorized users while only incurring a logarithmic (in the number of parties) overhead both in key and cipher text size when the number of adversaries isn't too large. Roughly, both of these schemes work as follows.

First, they generate cryptographic keys and then distribute these keys among all the possible users. Then to encrypt data to an authorized set of users, they encrypt the data under an appropriately chosen subset of the keys. The partitions of the keys and the subset used to encrypt are chosen to guarantee that all authorized user will know at least one key enabling it to decrypt the data, while all unauthorized users will not know any of these keys. One critical limitation of both of these schemes is that they only allow for one data owner who can share data, but in most big data scenarios there are multiple data providers.

Public-Key Broadcast Encryption: In order to overcome this single sender limitation, the literature turned to public-key broadcast encryption. Such schemes allow anybody to share data, but rely on much stronger computational assumptions. To achieve parameters similar scheme in the public-key setting. An alternative construction is to reduces the size of the secret keys needed by the data recipients down to a constant independent of the number of users, at the cost of (somewhat) increasing the size of the public key and encrypted data[3,4]. One major drawback of these schemes is that they rely on a relatively novel, powerful, and non-standardized cryptographic building block called bilinear groups.

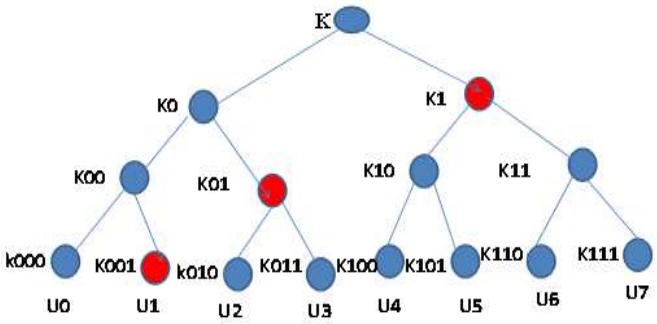
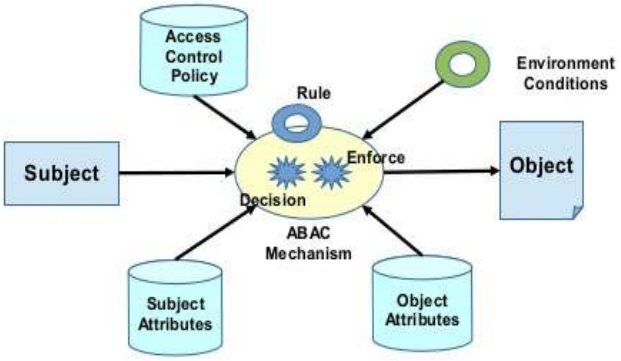


Figure: The NNL tree for broadcast encryption for 8 users. Each K represents an independent cryptographic key. Each user Uj receives all the keys on the path from the root to their leaf node. To encrypt the data, encrypt a copy of the data under the keys at the roots of sub trees containing only authorized users. For example, to encrypt to the set of all users except U0, one would use the keys corresponding to the nodes colored in red.

Attribute-Based Access Control

Key management based solutions have limitation. To share data with a set of users, it is necessary to know the identities (and keys) of all the authorized users. This is problematic in large systems or in systems with several organizational structures (as is very common in big data architectures where the data is collected, stored, and used in different environments) as the data owner is unlikely to know the identities of all the authorized users. An alternative approach

to access control in such settings is a technique called attribute-based access control (ABAC). In ABAC, data is encrypted together with a policy describing the attributes of users authorized to access the data. The users receive keys for the attributes they possess and are able to access the data if and only if those attributes are authorized. This allows for enforcing access to data without knowing the full set of users with the authorized attributes.



One approach to ABAC that has a trusted server evaluates the access policy over a user's attributes and grant or restricts access to data accordingly. However, this requires trusting the server to correctly administer these permissions and is problematic in scenarios where there is no such trusted entity, such as in outsourced storage. We instead focus on solutions for ABAC that do not require a trusted server to evaluate the access policies. A powerful cryptographic technique known as attribute-based encryption (ABE) that can be used to solve this problem cryptographically.

Attribute-Based Encryption:

The first ABE scheme to satisfy full collusion-resistance was supporting a limited set of policies known as threshold policies, where an authorized user's key has to have large overlap with the set of keys specified in the policy. An implementation of this scheme describing how this can be used for access control the class of supported policies was extended to arbitrary Boolean formulas.

All of these schemes to prevent collusion attacks. As in the no-collusion scheme above, they generate cryptographic keys corresponding to the possible attributes. However, each generated key is also personalized to the specific user. So, the key for attribute A given to user 1 would be different from the key for attribute A given to user2.

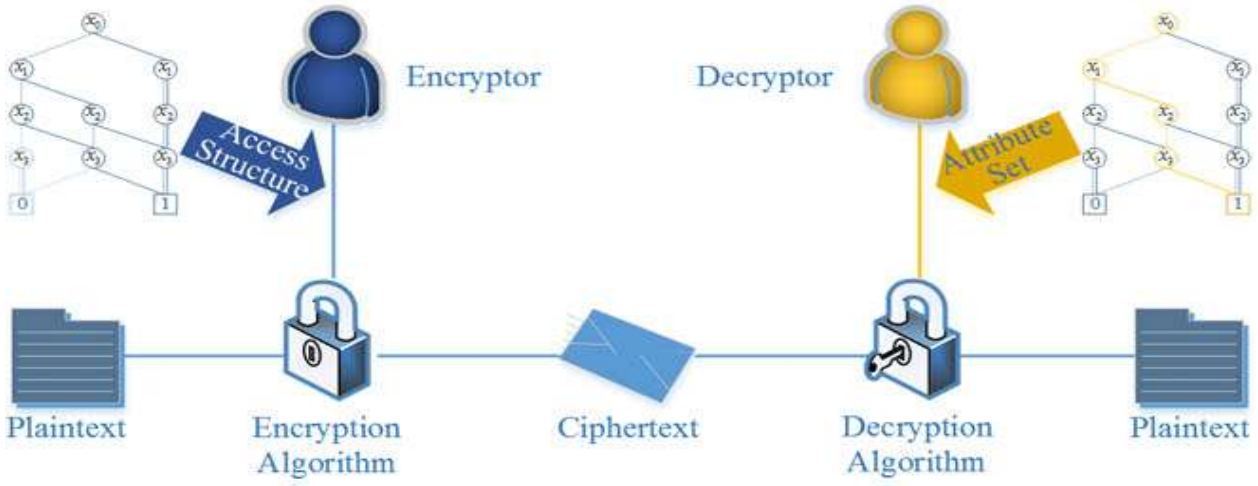
This prevents keys from different users from decrypting the data since the personalized components will not match. However, this personalization approach requires the use of a non-standardized cryptographic building block called bilinear groups and it is unlikely that (collusion resistant) ABE can be based on standard encryption.

Since these works there have been a number of improvements optimizing performance, achieving richer classes of access policies allowing for delegation of access permissions, and giving constructions relying on alternative security assumptions. There are now a number of available implementations of ABE is pairing-Based Crypto library. On the functionality side, we now have ABE schemes capable of enforcing very rich classes of policies such as policies specified by arbitrary (polynomial-size) Boolean circuits over the attributes.

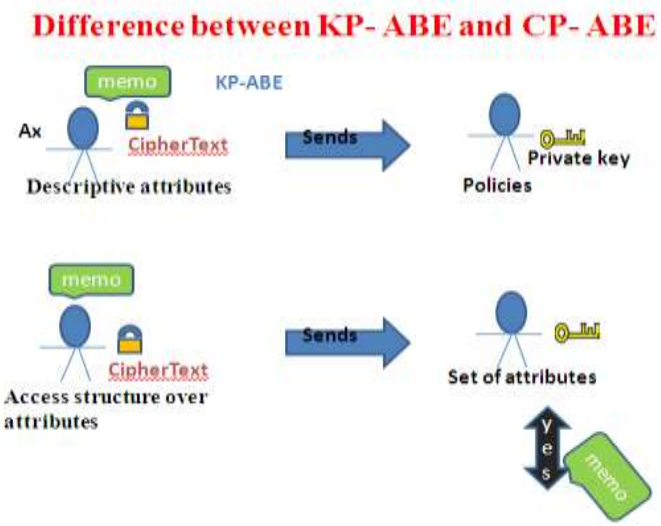
Key policy attribute based encryption providing security and access control with constant size cipher text. It is the modified form of ABE. In this approach users are assigned with an access tree structure. The attributes are associated by leaf nodes. Threshold gates are the nodes of the access tree. The attributes are associated by leaf nodes. The secret key of the user defines the access tree structure. Cipher texts are labeled with sets of attributes and private keys are associated with monotonic access structure that defines and controls which cipher text a user is able to decrypt. KP-ABE defines algorithms for setup, encryption, and key generation decryption.KP-ABE scheme can achieve fine-grained access control and more flexibility to control users than ABE scheme.

The problem with KP -ABE scheme is the data owner cannot decide who can decrypt the encrypted data. It can only choose descriptive attributes for the data and not suitable in some application because a data owner has to trust the third party for key issuing

Cipher text-policy attribute-based encryption is a form of encryption in which keys are associated with attributes and data is encrypted with a policy specifying which attributes are needed to decrypt the cipher text. The security of these primitive guarantees that a key will succeed in decrypting a cipher text if and only if the attributes in the decryption key satisfy the access policy specified in the cipher text. More formally, the cipher text contains a Boolean formula over attributes and a key will successfully decrypt the cipher text if it evaluates to 1 on the attributes contained in the key.



There are two properties that must be addressed in any ABE scheme [6]. The first is the expressivity of the supported policy formulas. First property is Schemes achieving more expressive policies can be used to enforce access control in more diverse settings. The second property is collusion resistance. That is, unauthorized users should not be able to combine their keys in order to decrypt data for which neither of their keys individually satisfies the access formula. This is typically the hardest feature of ABE to achieve and requires stronger cryptographic assumptions.



CP-ABE overcomes the drawbacks of KP-ABE that the encrypted data cannot choose who can decrypt. It can support access control in the real environment and the users private key in this scheme is a combination of a set of attributes, so that a user can use only this set of attributes to satisfy the access policy in the encrypted data. Drawback of most existing CP-ABE scheme is lack of flexibility and efficiency, for this reason most of the enterprises are unable to use it. CP-ABE decryption key supports attributes that are organized logically as a single set, so the users can only use all combinations of attributes in a single set issued in their keys to satisfy access policies.

III. CONCLUSION

In this paper we conclude that the comparison of all encryption schema algorithms and cryptography on security results less accuracy and efficiency on data, this survey can guide various challenges regarding Key Management ,ABAC, ABE, KP-ABE, CP-ABE.

REFERENCES & BIBLIOGRAPHY

[1] "Cryptography for Big Data Security Book Chapter for Big Data: Storage, Sharing, and Security" by Ariel Hamlin, Nabil Schear, Emily Shen, Mayank Varia, Sophia Yakoubov, 2015

[2] Amos Fiat and Moni Naor. Broadcast encryption. "In Advances in Cryptology - CRYPTO '93, 13th Annual International Cryptology Conference, Santa Barbara, Cal-ifornia, USA, August 22-26, 1993, Proceedings, pages 480{491, 1993.

[3] Dan Boneh and BrentWaters, "Constrained pseudorandom functions and their applications. In Advances in Cryptology –" ASIACRYPT 2013 - 19th International Conference on the Theory and Application of Cryptology and Information Security, Bengaluru, India, December 1-5, 2013, Proceedings, Part II, pages 280{300, 2013}.

[4] Dan Boneh, Brent Waters, and Mark Zhandry," Low overhead broadcast encryption from multilinear maps. In Advances in Cryptology – "CRYPTO 2014 - 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part I, pages 206{223, 2014.

[5] Oracle labels security with oracle database http://www.oracle.com/technetwork/database/options/label-security/label-security-wp-12c-1896140.pdf?ssSourceSiteId=ocomen, June 2013

[6] Time domain attribute based encryption for big data access control in cloud environment ACCENTS Transactions on Information Security, Vol 2(7).