ISSN NO: 9726-001X
Volume 08 Issue2 April 2020

DECENTRALIZED ACCESS CONTROL ENABLES ANONYMOUS ENDORSEMENT IN THE CLOUD

M.KRISHNA MURTHY
Department of computer Engineering

[I] BSTRACT

The use of anonymous authentication in a decentralized access control strategy for cloud-based data storage. Before saving data, the cloud does a validity check on the series without revealing the user's identify. It also provides the additional benefit of access control, so that only authorized users may decode the data. This protects cloud-based information against replay assaults and enables its creation, variation, and consumption. Cancelling a user's access is a feature too. This is a crucial feature, since a user who has had their privileges revoked may no longer be able to save data in the cloud. In addition, our authentication and access control strategy is resilient and distributed, in contrast to existing access control techniques made specifically for clouds, which are centralized. Comparable to centralized methods in terms of communication, processing, and storage costs.

INTRODUCTION

In cloud computing, users rely on remote servers (or "clouds") for their computing and data storage needs. This alleviates the burden of storing and updating equipment for the user. Services like Amazon Elastic Compute Cloud (EC2), Eucalyptus, and Nimbus, as well as application development platforms like Amazon S3 and Microsoft Azure, are all examples of what may be found in the cloud.

Clouds host a variety of data, including medical information and social network profiles, that is considered very sensitive.As a result, cloud computing security and privacy are major concerns. User anonymity is also necessary.

in order to conceal one's identify from the cloud service or other users. The cloud may make the user responsible for whatever data it outsources, and it must also take responsibility for the quality of the services it offers. The data storage user's identity is also confirmed for accuracy. When it comes to protecting people's personal information and digital identities, technological solutions aren't enough.

Clouds also raise the issue of how to efficiently look through encrypted data. Clouds shouldn't be able to see the query, but they should still deliver results that match it. To accomplish this, searchable encryption [3, 4] is used. Cloud services receive encrypted keyword queries and provide results without decrypting them. The issue here is that the data records need to be tagged with keywords in order to be searchable. Only an accurate keyword search will get the desired results.

The cloud receives the data in ciphertext, conducts calculations on the ciphertext, and then delivers the encoded value of the result via homomorphic encryption. The user can decipher the outcome, but the cloud has no idea what information it has processed. There has to be a way for the user to check whether the cloud is providing accurate results in such a scenario.

Taking into account the following scenario: Alice, a law student, wishes to notify all of the professors at University X, the research chairs at universities around the nation, and the students in the Law departments at universities across the province of a series of reports detailing alleged misconduct on the part of University X administration. She prefers to keep her privacy while sharing everything

indications of wrongdoing. She files the data away on a remote server. In this scenario, access control is crucial to ensure that only approved individuals may access the information. Checking the information's credibility is also essential. The issues of authentication, access management, and privacy protection

concurrently resolved. In this work, we take up the whole scope of this issue.

Only authorized users should be able to access legitimate service, hence access control in clouds is gaining traction. Access control may be broken down into three major categories: user-based (UBAC), role-based (RBAC), and attribute-based (ABAC). UBAC's access control list specifies which users are permitted to see a certain set of records. There are too many people using the cloud for this to be practical. Users in RBAC are categorized according to the tasks they are authorized to do. Users with appropriate permissions may see the data. The system determines who does what. For instance, junior secretaries may not have access to information but faculty members and senior secretaries may. ABAC is more comprehensive since it assigns characteristics to users and associates an access policy with each data set. Only those users whose qualities match the requirements of the access policy will be granted access. Faculty members with more than 10 years of research experience and senior secretaries with more than 8 years of experience, for instance, may have access to particular documents under the aforementioned scenario.

[II] EXIXTING APPROACH

The Data are accessed in centralized way on the basis of single KDC,where KDC means Key Distribution Center(KDC) which is responsible for the distribution of keys and attributes to the users.A single Key distribution center does not support for authentication. A single failureof KDC can affect the maximum of data in cloud storage. Unfortunately, a single KDC is not only a single point of failure but difficult to maintain because of the large number of users that are supported in a cloud environment. The other drawback was that a user cancreate and store a file and other users can only read the file. Write access was not permitted to users other thanthe creator.

[III] PROPOSED APPROACH

An area where access control is widely beingused is health care. Clouds are being used to store sensitive information about patients to enable access tomedical professionals, hospital staff, researchers, and policy makers. To Maintain the large number of datas in cloud, the decentralized access control approaches is proposed. It involves many KDC's for the distribution of secret keys and attributes of all users. It is important to control the access of data so that only authorized users can access the data. Using ABE, the records are encrypted under some access policy and stored intheclo ud . Users are give n sets o f attrib utes and corresponding keys. Only when the users have matching set of attributes, can they decrypt the information stored in the cloud.By the use of ABS the authenticity and the privacy can be achieved. This decentralized scheme also allows writing multiple times which was not possible in the existing approach.

4.1 ASSUMPTIONS

We make the following assumptions in our work:

1. The cloud is honest-but-curious, which meansthat the cloud administrators can be interested in view-ing user's content, but cannot modify it.This is a valid assumption that has been made in [12] and [13]. Honest-but-curious model of adversaries do not tamper with data so that they can keep the system functioning normally and remain undetected.

2. Users can have either read or write or both accesses to a file stored in the cloud.
3. All communications between users/clouds are se-cured by secure shell protocol, SSH.

4.2 CONTRIBUTIONS

The main contributions are:

1. Distributed access control of data stored in cloudso that only authorized users with valid attributescan access them.
2. Authentication of users who store and modify their data on the cloud.
3. The identity of the user is protected from the cloud during authentication.
4. The architecture is decentralized, meaning that there can be several KDCs for key management.

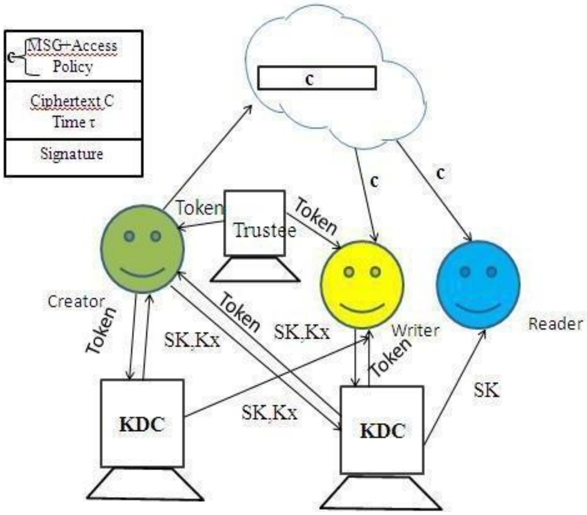
5. The access control and authentication are both collusion resistant, meaning that no two userscan collude and access data or authenticate themselves, if they are individually not

authorized.

6. Revoked users cannot access data after they have been revoked.
7. The proposed scheme is resilient to replay attacks.A writer whose attributes and keys have been revoked cannot write back stale information.

8. The protocol supports multiple read and write on the data stored in the cloud.
9. The costs are comparable to the existing centralized approaches, and the expensive operations are mostly done by the cloud.

4.3 SYSTEM MODEL



Here is the privacy p reservin g authenticated access control scheme. According tothe scheme a user cancreate a file and store it securely in the cloud. This scheme consists of use of the two protocols ABE and ABS. There are three users, a creator, a reader, and writer.Creator receives a token from the trustee, who is assumed to be honest. A trustee can be someone like the federal government who manages social insurance numbers etc. On presenting her id (like health/social insurance number), the trustee gives her a token.There are multiple KDCs (here 2), which can be scattered. For example,these can be servers in different parts of the world. A creator on presenting the token to one or more KDCs receives keys for encryption/decryption and signing. In the Fig. 1, *S Ks* are secret keys given for decryption, K_x are keys for signing. The message *MSG* is encrypted under the access policy X. The access policy decides who can access the data stored in the cloud. The creator

decides on a claim policy Y, to prove her authenticity and signs the message under this claim. The ciphertext *C* with signature is *c,* and is sent to the cloud. The cloud verifies the signature and storesthe ciphertext *C.* When a reader wants to read, the cloud sends *C.* If the user has attributes matching with access policy, it can decrypt andget back original message.Write proceeds in the same way as file creation. By designating the verification process to the cloud, it relieves the individual users from time consuming verifications. When a reader wants to read some data stored in the cloud, it tries to decrypt it using the secret keys it receives from the KDCs. If it has enough attributes matching with the access policy, then itdecrypts the information stored in the cloud.

4.4 ATTRIBUE BASED ENCRYPTION

Attribute-based encryption (ABE) is a relatively recent approach that reconsiders the concept of public- key cryptography. In traditional public-key cryptography, a message is encrypted for a specific receiver using the receiver’s public-key. Identity-based cryptography and in particular identity-based encryption (IBE) changed the traditional understanding of public- key cryptography by allowing the public-key to be an arbitrary string, e.g., the email address of the receiver. ABE goes one step further and defines the identity not atomic but as a set of attributes, e.g., roles, and messages can be encrypted with respect to subsets of attributes (key-policy ABE - KP-ABE) or policies defined over a set of attributes (ciphertext-policy ABE - CP-ABE). The key issue is, that someone should only be able to decrypt a ciphertext if the person holds a key for "matching attributes" (more below) where user keys are always issued by some trusted party.

4.4.1 Ciphertext-Policy ABE

In ciphertext-policy attribute-based encryption (CP-ABE) a user's private-key is associated with a set of attributes and a ciphertext specifies an access policy overa defined universe of attributes within the system. A userwill be able to decrypt a ciphertext, if and only if his attributes satisfy the policy of the respective ciphertext. arbitrary circuits).

CP-ABE thus allows to realize implicitauthorization, i.e., authorization is included into the encrypted data and only people who satisfy the associated policy can decrypt data. Another nice features is, that users can obtain their private keys after data has been encrypted with respect to policies. So data can be encrypted without knowledge of the actual set of users that will be able to decrypt, but only specifying the policy which allows to decrypt. Any future users that will be given a key with respect to attributes such that the policy can be satisfied will then be able to decrypt the data.

4.4.2 Key-Policy ABE

KP-ABE is the dual to CP-ABE in the sense that an access policy is encoded into the users secret key, e.g., (AAC)∨D, and a ciphertext is computed with respect to a set of attributes, e.g., {A,B}. In this example the user would not be able to decrypt the ciphertext but would for instance be able to decrypt a ciphertext with respect to {A,C}.

An important property which has to be achieved by both, CP- and KP-ABE is called collusion resistance. This basically means that it should not be possible for distinct users to "pool" their secret keys such that they could together decrypt a ciphertext that neither of them could decrypt on their own

4.5 ATTRIBUTE BASED SIGNATURE

Keeping data in the cloud safely isn't always enough; sometimes it's also important to protect the user's privacy. A user could, for instance, choose to anonymize himself while storing sensitive data. The user could wish to leave a remark on the article, but remain anonymous. A user may keep information anonymously, but must be able to show to other users that they are the legitimate owner of the data. For this purpose, we have cryptographic techniques such as ring signatures [20], mesh signatures [21], and group signatures [22]. For highly trafficked clouds, ring signature is not a viable alternative. For group signatures to work, a group must already exist, which might be impossible in clouds. Mesh signatures cannot verify whether a message originated from a single user or a group of users working in concert. A new protocol called attribute-based signature (ABS) has been implemented for this same purpose. A claim predicate is attached to a message in ABS. The claim predicate aids in authenticating the user as a legitimate one without disclosing the user's true identity. The cloud or other users may confirm the identity of the user and the authenticity of the communication.

[IV] CONCLUSION

It presented a decentralized access control technique with anonymous authentication, which provides removes the user and stops repeat assaults. The cloud does not track down the person behind the credentials it checks in order to access data. There is no central authority in charge of handing out keys. One restriction is that every cloud-stored data comes with its own predetermined access policy. We want to one day be able to conceal a user's profile details and security settings.

[V] REFERENCES

In 2012, the Proceedings of the IEEE/ACM International Symposium on Cluster, Cloud, and Grid Computing included a paper titled "Privacy Preserving Access Control with Authentication for Securing Data in Clouds" by S. Ruj, M. Stojmenovic, and A. Nayak.
According to [2] "Toward Secure and Dependable Storage Services in Cloud Computing," by C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, published in IEEE Trans. Services Computing, volume 5, issue 2, pages 220–232, April–June 2012.
Fuzzy Keyword Search Over Encrypted Text," by J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, and W. Lou.

Information Technology Conference Proceedings, IEEE INFOCOM,

pp. 441-445, 2010.

In 2010, during the 14th International Conference on Financial Cryptography and Data Security, S. Kamara and K. Lauter presented "Cryptographic Cloud Storage" to the audience.

According to [5] "Identity-Based Authentica-tion for Cloud Computing" by H. Li, Y. Dai, L. Tian, and H. Yang (published in 2009's Proceedings of the First International Conference on Cloud Computing: CloudCom), pages 157–166.

PhD dissertation by C. Gentry, Stanford University, 2009. Available online at http://www.crypto.stanford.edu/ craig as "A Fully Homomorphic Encryption Scheme."

Third International Conference on Trustworthy Computing (TRUST), pp. 417-429, 2010. A.-R. Sadeghi, T. Schneider, and M. Winandy. "Token-Based Cloud Computing."

Trustcloud: A Framework for Accountability," by R.K.L. Ko, P. Jagadpramana, M. Mowbray, S. Pearson, M. Kirchberg, Q. Liang, and B.S. Lee

transparency, and confidence in the cloud," HP Technical Report HPL-2011-

38, 2011/HPL-TechReports, http://www.hpl.hp.com/techreports.

2011-38.html, 2013.

The paper "Secure Provenance: The Essential of Bread and Butter of Data Forensics in Cloud Computing" [9] was written by R. Lu, X. Lin, X. Liang, and X. Shen for the Proceedings of the Fifth International Conference on Data Mining (ICDM).