Research Paper

Volume : 4 | Issue : 12 | December 2015 • ISSN No 2277 - 8179

Distributed Access Control Mechanism in Cloud Environment with Anonymous Authentication

Engineering

KEYWORDS : cloud storage, KDC, Access-control, authentication, ABS (Attribute-Based signature), ABE (Attribute-Based encryption).

K.V.N.Sai kumar — M.tech Student, Dept.Of computer science &Engineering Chalapathi Institute of Engineering and Technology A.R.Nagar, Tadikonda, Guntur-522016

Dr.Indira Priyadarshini — Professor, Dept.Of computer science &Engineering Chalapathi Institute of Engineering and Technology A.R.Nagar, Tadikonda, Guntur-522016

ABSTRACT

Here in this paper we propose a new access control scheme for providing security to the data stored in clouds. The other access control schemes which are already designed are centralized i.e. having a single KDC (KEY DISTRIBUTION CENTRE) nature. Our Access control scheme is decentralized that supports anonymous authentication. In our proposed system "Before storing data and without knowing users identity the cloud verifies the authenticity of the series". Access control is another feature which allows only valid users to decrypt the stored information from cloud. Our proposal successfully prevents replay attacks and also supports creation, modification and reading of data stored in the cloud.

Introduction:

We should provide security to the data stored in clouds. Because the data stored in clouds is highly sensitive (EX: Medical records, social network e.t.c).In cloud computing security and privacy are very important. The user must prove that he\she is authenticated users before they do any transaction and also there must be ensured that the cloud doesn't tamper with any out-sourced data. The other important issue in clouds is that efficient search; we can achieve this through searchable encryption. We can also search through a key-word-here an encrypted keyword is send to the cloud and cloud returns the result without knowing that what the actual keyword is.

In clouds security and privacy are challenging. a lot of research has been continue in this area for years. Storage security was addressed by Wang et.al [1] using erasure-codes and read Solomon codes. Access control in clouds is also important then only authorized users have access cloud services.

In clouds a huge amount of information is stored and also that information is very sensitive (EX. Important documents of an organization, health related information, personal information).so we must take care of this sensitive information and protect that information from un authorized access[2].

We have three types of access control mechanisms:

- User-Based Access control (UBAC).
- Role-Based Access control (RBAC).
- Attribute-Based access control (ABAC).

The first one User-Based access control is not work out in clouds because in this type of access control a access control list is maintained which is used to specify who are allowed get all the clod services, but this is not possible to list out all the users of a cloud because they are huge. The Role-Based and Attribute-Based access control extended in scope. The pros and cons of ABAC and RBAC are discussed in [3].In clouds same work has been in ABAC [4][5].A cryptographic primitive also known as Attribute based encryption used in all these work.

Related-Work:

In the existing work access control in cloud are centralized in nature. Some scheme doesn't support authentication (systematic key approach).some of the work support authentication access control [6] proposed by Zhao et al .most of the authors worked on the centralized approaches where a single key distribution centre(KDC) distributes secret keys and attributes to all users.

In centralized KDC if failure happens it shows its impact on huge number of cloud users because all the user services depend on a single KDC. Therefore cloud should take a decentralized approach where distributing secret keys and attributes by multiple KDCs.

In the existing system Yang et al [7] proposed a decentralized approach but the techniques used here are not authenticating the users. Another distributed access control proposed by rug et al [8] also didn't provide user authentication.

In this paper the access-control, authentication and privacy protection are addressed for this we use Attribute-based signature and Attribute-Based Encryption schemes [9][10].Here we also solve the problem of replay attacks.

Proposed Access control scheme:

What we are going to achieve in this work is:

- Avoiding stale information and also replay attacks.
- Key-management is decentralized through several KDCs.
- Support multiple read and simultaneously.
- Cost, computation, communication and storage are comparable to centralized systems.
- User identity is protected.
- It is collusion resistant in the authentication and access control cases.
- Only authorized users are allowed to access cloud services.

A user create a file and store it securely In the cloud. Here we consists 2 protocols

1. ABS (Attribute-Based signature)
2. ABE (Attribute-Based encryption)

Let us assume that there are three users

- A creator
- A reader
- A writer

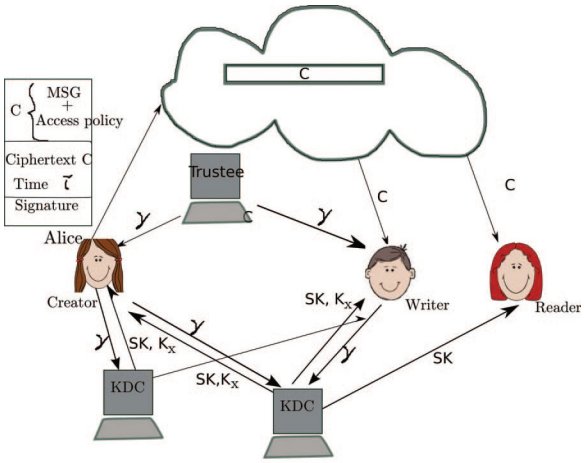
A creator receives a token from any trusty. (a trusty must be honest).Trusty provides token when he/she presenting his/her ID(health/social insurance number).Our scheme is decentralized access i.e. multiple KDCs are scattered.

A Token γ is getting from any authorized trusty.

A creator can get keys from one (or) more KDCs are S_k,p_K for en-

cryption/decryption and $AS_{K,}AP_{K}$ for signature/verification.

A user can get attribute and keys on presenting this token γ. The MSG is encrypted under the access policy X.



The access policy X decides who can access the data in the cloud. The cipher text C under the signature is sent to the cloud. The cloud verifies signature, only authorized users can access this cipher text. The cipher text only decrypted when the users have attributes matching with the access policy.

Cipher text (c) = ABE. Encrypt (MSG, X).

Here the user can not send the MSG as it is. The user also has a claim policy y to enable the cloud to authenticate the user. H(c) ||t where t-time stamp. By sending this t-time stamp there is no way to the user to write stale MSG back to the cloud. In the existing system magi et.al [11] suffered from this replay attacks, we can successfully overcome that problem through our policy. The message signature is calculated as follows:

SIG=ABS.Sign(PK_{I},PK_{KDS},TOKEN,SIGNED KEY,MSG, access-claim).

Where PK_{I}= Public key of trustee;

PK_{KDS}= Public key of KDCs

The information sent to the cloud can verify the access claim. The creator can also check whether the file is successfully stored in the cloud or not trough the value V.

V= ABS. verify (TPK, SIG, c, y).

If the value V=0 means file not accepted by cloud for storing into it. else(c,t) stored.

Mathematical Representation of our proposed system:
γ = (u,Kbase,K0, ρ) Where ρ is the signature on the $u||K_{base}$ signed with the trustees private key T_{sig}.

After getting this token γ from any trusty it will sent to one or more KDC for getting attributes and secret keys.

2.1Let C and D are two users and have attributes x_c and x_d respectively, they have the following information Kbasec, Kxc and Kbased,Kxd.

$K_{xc}=k_{basec}^{1/(a+bx)}$ cannot be calculated by D, because (a, b) values are not known by D. Now the authentication is collusion secure.

C=MSG+access policy(X)
H(c)||t -→ cipher text
Now encrypt the cipher text
c=(C,t,SIG,Y)
here Y is a claim policy constructed by a creator.

Depending on this claim policy users can get access rights. M-W-1-R means that many users can write while one user can read. 1-W-M-R means that one user can write and many can read. We see that most schemes do not support many writes which is supported by our scheme. Our scheme is robust;

Now cloud verifies the signature by using the function ABS.verify[12].

6.{ if(ABS.verify) (C,t)store Else Discard the file }

7. writing to the cloud
7.1 users who wants to write into already existing file they must send a request message with a claim policy-Y done during the file creation[13].
7.2 Now cloud verifies this policy and allow the user for writing if they are authentic users.
8. Reading from cloud
8.1 User send a request message to the cloud.
8.2 cloud sends cipher text c if they are authentic users.
8.3 Now user decrypt it using algorithm ABE.decrypt(c,{$sk_{i,u}$}). where $sk_{i,u}$ secret key given by set of KDCs to the users.

Conclusion:
e successfully avoid replay attacks and provide privacy preserving authentication where most of the other schemes doesn't support it. we achieve access-control through a decentralized and robust scheme. The cloud doesn't know the users identity who store information. Here our access control scheme is secure i.e. no outsider or even cloud cannot decrypt cipher text.

REFERENCE

[1] S. Ruj, M. Stojmenovic, and A. Nayak, "Privacy Preserving Access | Control with Authentication for Securing Data in Clouds," Proc. | IEEE/ACM Int'l Symp. Cluster, Cloud and Grid Computing, pp. 556- | 563, 2012. | [2] C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, "Toward | Secure and Dependable Storage Services in Cloud Computing," | IEEE Trans. Services Computing, vol. 5, no. 2, pp. 220-232, Apr.- | June 2012. | [3] J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, and W. Lou, "Fuzzy | Keyword Search Over Encrypted Data in Cloud Computing," | Proc. IEEE INFOCOM, pp. 441-445, 2010. | [4] S. Kamara and K. Lauter, "Cryptographic Cloud Storage," Proc. | 14th Int'l Conf. Financial Cryptography and Data Security, pp. 136- | 149, 2010. | [5] H. Li, Y. Dai, L. Tian, and H. Yang, "Identity-Based Authentication | for Cloud Computing," Proc. First Int'l Conf. Cloud Computing | (CloudCom), pp. 157-166, 2009. | [6] C. Gentry, "A Fully Homomorphic Encryption Scheme," PhD | dissertation, Stanford Univ., http://www.crypto.stanford.edu/ | craig, 2009. | [7] A.-R. Sadeghi, T. Schneider, and M. Winandy, "Token-Based | Cloud Computing," Proc. Third Int'l Conf. Trust and Trustworthy | Computing (TRUST), pp. 417-429, 2010. | [8] R.K.L. Ko, P. Jagadpramana, M. Mowbray, S. Pearson, M. | Kirchberg, Q. Liang, and B.S. Lee, "Trustcloud: A Framework | for Accountability and Trust in Cloud Computing," HP Technical | Report HPL-2011-38, http://www.hpl.hp.com/techreports/ | 2011/HPL-2011-38.html, 2013. | [9] R. Lu, X. Lin, X. Liang, and X. Shen, "Secure Provenance: The | Essential of Bread and Butter of Data Forensics in Cloud | Computing," Proc. Fifth ACM Symp. Information, Computer and | Comm. Security (ASIACCS), pp. 282-292, 2010. | [10] D.F. Ferraiolo and D.R. Kuhn, "Role-Based Access Controls," Proc. | 15th Nat'l Computer Security Conf., 1992. | [11] D.R. Kuhn, E.J. Coyne, and T.R. Weil, "Adding Attributes to Role- | Based Access Control," IEEE Computer, vol. 43, no. 6, pp. 79-81, | June 2010. | [12] M. Li, S. Yu, K. Ren, and W. Lou, "Securing Personal Health | Records in Cloud Computing: Patient-Centric and Fine-Grained | Data Access Control in Multi-Owner Settings," Proc. Sixth Int'l | ICST Conf. Security and Privacy in Comm. Networks (SecureComm), | pp. 89-106, 2010. | [13] S. Yu, C. Wang, K. Ren, and W. Lou, "Attribute Based Data | Sharing with Attribute Revocation," Proc. ACM Symp. Information,Computer and Comm. Security (ASIACCS), pp. 261-270, 2010. |