Data Security Accessing for HDFS Based on Attribute-Group in Cloud Computing

Huixiang Zhou
State Key Laboratory of Networking and Switching Technology, BUPT
Beijing, China
Zhouhuixiang168@163.com

Qiaoyan Wen
State Key Laboratory of Networking and Switching Technology, BUPT
Beijing, China
wqy@bupt.edu.cn

Abstract—**To solve the security issues of network features and data sharing features in cloud storage service, this paper proposes a data security access scheme in cloud storage Based on Attribute-Group. In terms of access rights control and access control architecture for the access control mechanism research, in this scheme, data owners do not participate in the specific operation of the property and user rights, re-encryption on the NameNode, can reduce the cost of the computation and management of the client, and also reduce the complexity of rights management and property management**.

Key words: cloud storage, data security access, Hadoop, HDFS, attribute-Group.

I. INTRODUCTION

Research of IDC shows that from 2007 to 2012, more than 6 times the global growth in the amount of information, from 161EB increased 988EB (1EB = 1024PB). The face of petabytes of mass storage needs, there is bottleneck in the expansion of capacity and performance on the traditional SAN or NAS, and even some of the existing high-end storage devices obviously cannot meet this demand [1]. The mass storage demand for storage capacity, storage access performance and cost of the unprecedented challenges, cloud storage will become an ideal way for future storage because of its superior scalability.

However, whether the user be assured that put t he private data to various cloud is the key that whet her cloud storage can be widely promoted. And the user's sense of trust cannot simply rely moral constra ints or rules on the management and finally be resol ved. If the technically can makes private data only b e looked by its own, that can fundamentally solve th e problem that user trust in the cloud storage.

Key corresponds to a set of attributes, based on the group attributes encrypted cipher text strategy (Cipher text-policy Attribute-based Encryption, CP-ABE)[4] cipher text corresponds to an access structure and decrypted if and only if the set of attributes property to meet the access structure. CP-ABE applies the decrypting party fixed in a distributed environment, the decryption rule contains the encryption algorithm, CP-ABE this particular integrated multi-NameNode characteristics suitable for cloud storage, which can be removed from the cipher text access control that occur frequently in the cost of key distribution. Apache Hadoop[5] is an open-source software framework that supports data-intensive distributed applications, the Hadoop Distributed File System (HDFS)[6] is a distributed, scalable, and portable file system written in Java for the Hadoop framework, and HDFS is cloud storage the most widely used tool. This article is based on the CP-ABE and HDFS designing a secure cloud storage the cipher text control program. On the basis of the CP-ABE and symmetric encryption algorithm (such as AES), we propose a cloud-oriented storage efficient dynamic access control scheme cipher text.

II. RELATED RESEARCH

A. HDFS technology

HDFS is a distributed, scalable, and portable file system written in Java for the Hadoop framework. Each node in a Hadoop instance typically has a single NameNode, a cluster of DataNodes form the HDFS cluster. The situation is typical because each node does not require a DataNode to be present. Each DataNode serves up blocks of data over the network using a block protocol specific to HDFS. The file system uses the TCP / IP layer for communication. Clients use Remote procedure call (RPC) to communicate between each other. HDFS stores large files (an ideal file size is a multiple of 64 MB), across multiple machines. It achieves reliability by replicating the data across multiple hosts, and hence does not require RAID[7]b5 storage on hosts. With the default replication value , 3, data is stored on three nodes: two on the same rack, and one on a different rack. Data nodes can talk to each other to rebalance data, to move copies around, and to keep the replication of data high. HDFS is not fully POSIX compliant, because the requirements for a POSIX file system differ from the target goals for a Hadoop application. The tradeoff of not having a fully POSIX-compliant file system is increased performance for data throughput and support for non-POSIX operations such as append.

B. CP-ABE

Sahai and Waters first proposed based on fuzzy identity is encrypted[8], as the identity of the biological characteristics apply the information based on the concept

© 2014. The authors - Published by Atlantis Press

of the identity of the encryption scheme, Sahai introduced in the paper attributes, based on 2006, Goyal et al The fuzzy identity encryption scheme based on the attribute-based encryption scheme (ABE, attribute-based encryption). 2007, Bethencourt et al cipher text policy attribute-based encryption (CP-ABE, cipertext-policy ABE), the user's identity is represented as a collection of a property, and encrypt data access control structure Union, whether a user can decrypt the cipher text, depending on whether the cipher text associated set of attributes corresponding to the user identity-based access control structure match.

C. CPE-ABE algorithm

Attributes: Set $P=\{P_l,P_2,\mathrm{L}\ ,P_n\}$ collection of all properties, properties for each user A is a non-empty subset of P, $A\in 2^N$, then the N attribute can be used to identify 2^N user.

Access structure: Access structure T is a nonempty subset of the complete works $\{P_l,P_2,\mathrm{L}\ ,P_n\}$ and $T\in 2^{\{P_l,P_2,\mathrm{L}\ ,P_n\}}$ $A\in\{P_l,P_2,\mathrm{L}\ ,P_n\}\backslash\{\phi\}$. T represents an attribute judgment condition: in T attribute set is called the authorization set, not in the T attribute set is called the set of non-authorized.

III. SCHEME OF DATA SECURITY ACCESS BASED ON ATTRIBUTE-GROUP

In this paper, the properties of the cipher text CP-ABE encryption algorithm based on cloud storage data security access control scheme. Compared to the data owner directly distributed key distribution, centralized management of key distribution method and NameNode-based CP-ABE, easier to manage keys, but also more transparent to the user, that allows users to less involved key generation, key distribution, and other matters.

There is certain credibility, requiring CSP must be faithful to run the program and visits

Asked the agreement, yet may spy the contents of the data file, and assumes that all parameters and between the communication channel is secure.

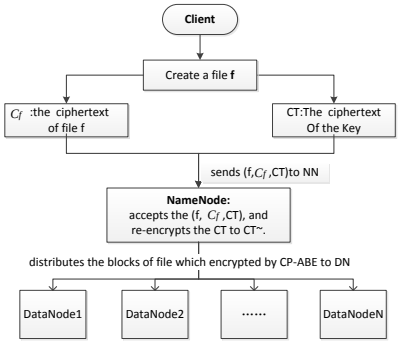

Fig1. the architecture of Data security access Scheme

A. System Initialization

The trusted authority CA 2 of order the primes p cyclic group G_1 and G_2 bilinear mapping $e{:}G_1\times G_{(}\to G_2$, g is the G_1 generator. Define the attribute space $\lambda=\{\lambda_1,\lambda_2\mathrm{L}\ ,\lambda_q\}$, each attribute $\lambda_i\in\lambda(1\le i\le q)$, randomly selected $w_i\in G_l$, $att(\lambda_i)$ return the property λ_i corresponding ID i, $U=\{u_l,u_2,\mathrm{L}\ ,u_n\}$ all users in the system, the attribute group $U_{\lambda_i}\subset U$ property λ_i set of users. The CA randomly selected $\alpha,\beta\in Z_\rho^*$ publish the public parameters $g,g^\beta,e(g,g)^\alpha,\{w_i\}_{i=l}^J\}$, kept secret master key $MK=(\beta,g^\alpha)$.

B. Key generation

The key generation include two parts: the user's private key generation algorithm and path key generation algorithm.

1) The user's private key generation algorithm:

Enter the user set U and set of attributes lambda, the algorithm outputs each user $u_t\in U$ set of attributes $\Lambda\subset\lambda$ of the corresponding private key SK_t. CA select $r\in Z_\rho^*$ randomly (corresponding to each user r), for each attribute $\lambda_i\in A$ randomly selected $r\in Z_\rho^*$ contingent. After generating the private key of each, $SK_t=(D=g^{(\alpha+r)/\beta},\forall\lambda_i\in A{:}D_i=g^r(w,lH(\cdot))^r,D_i=g^i)$ of user u_t.

Finally, CA via a secure channel SK_l send u_t, each attribute λ_i corresponding attribute group the U_{λ_i} send data administrator NameNode. Set of attributes of user u_l,u_2,u_3 and u_4 is $\{\lambda_1,\lambda_2\},\{\lambda_2,\lambda_3,\lambda_4\},\{\lambda_1,\lambda_3\}$, and $\lambda_1,\lambda_{12},\lambda_3,\lambda_4$. CA property $\lambda_1,\lambda_{12},\lambda_3,\lambda_4$ corresponding attribute group FF, $U_{\lambda_1}=\{u_l,u_3,u_4\}$, $U_{\lambda_2}=\{u_l,u_2,u_4\}$, $U_{\lambda_3}=\{u_2,u_3,u_4\}$ and $U_{\lambda_4}=\{u_2,u_4\}$ sent to the NameNode.

2) Path key generation algorithm:

NameNode to construct a binary tree to (called KEK tree), where the leaf nodes of the tree members of each user in U, v_j assignment of each node in the tree a random number $KEK_j\in Z_\rho^*$. Corresponding to the root of the tree with the user u_t There is a path between the leaf node, u_t path defined that all nodes on the path corresponding to the random number key PK_t.

The NameNode sents PK_t to the user via a secure channel $u_t\in U$. KEK tree as shown in Fig1, the user sets $U=\{u_l,u_2,u_3,u_4,u_5,u_6,u_7\}$, the path of the user key $U=\{u_l,u_2,u_3,u_4,u_5,u_6,u_7\}$ of u_1.

C. file creation

Data owners do first create a data file f, then randomly select a symmetric key k_f using the symmetric

encryption algorithm E and k_f encryption f to data cipher text $U=\{u_l,u_2,u_3,u_4,u_5,u_6,u_7\}$. To construct an access tree T, each leaf of the tree Node represents an attribute. k_τ set the threshold value of each node in the T x, node x q_x of a polynomial k_X-1.

Random selection $s\in Z_\rho^*$ to the root node R, the $q_R(0)=s$, For non-root node x, the $q_x(0)=q_{\rho^{"}rent(X)}(index(x))$ parent (x) Returns the parent node of the node x index (x) returns the serial number of the node x in its sibling. Assume that Y is a collection of all the leaf nodes of T, DO generate key cipher text of k_f:

$CT=(T,C=k_fe(g,g)^{\alpha s},C=g^{\beta s},\forall y\in Y:$

$C_y=g^{4_y(0)},C_y^I=(w,"t(\lambda_J))^{4_y(0)})$

Finally, do (C_f,CT) sent along to the CSP.

D. File re-encryption

Data Services NameNode received (C_f,CT), CT following the re-encryption operation:

1) For each attribute $y\in Y$ corresponding attribute group U_{λ_J} randomly selected $K_{\lambda_J}\in Z_\rho^*$ of U_{λ_J} group key calculation:

$CT`=(T,\hat{C}=k_fe(g,g)^{\alpha s},C=g^{\beta s},$

$\forall y\in Y:C_y=g^{(0)},C_y^1=((w_{att(\lambda y)})^{q_y(0)})^{K\lambda_y})$

2) at the KEK tree, looking for the U_{λ_i} able to cover all users the best KID tree, and the root node of the sub tree corresponding to the random number is defined as the path of the attribute group key, denoted $KEK(U_{\lambda_i})$. For example, in Fig1, if the attribute group $U_{\lambda_L}=\{u_l,u_3,u_4\}$ KEK tree can override the user u_1, u_3 and u_4 minimal root of the tree $\{v5,v8\}$, U_{λ_L} then the path key $\{KEK_5,KEK_8\}$, that is, only the U_{λ_L} user permission to know $KEK(U_{\lambda_L})$.

3) with a symmetric encryption algorithm E, and the path of the attribute group key to encrypt the group key, Generate the header message is as follows:

$Hdr=(\forall y\in Y:\{E_K(K_{\lambda_J})\}_{K\in KBK(U_l)})$ (4)

The data is stored in NameNode $(Hdr,C_f,CT`)$.

E. File Access

User access to the data file f: the decryption header message HDR group Key, decryption key secret text CT` get symmetric key k_f, the decrypt, according to data cipher text C_f data file f. Specifically described as follows:

1) Assuming the user the u_t to a CSP initiated requests for access to data files f, NameNode returns u_t to $(Hdr,C_f,CT`)$ e.

2) The u_t an attribute group corresponding group key decryption Hdr. We may assume that the user u_t is a legitimate property λ_i , then u_t to use $KEK\in KEK(U_{\text{var}})\,\mathrm{I}\ PK_t$ decryption Hdr get the attribute group U_{λ_L} corresponding group key K_{λ_l}.

If the attribute group $U_{\lambda_L}=\{u_1,u_3,u_4\}$ key due to the path of the attribute group aa12 path keys

$KEK(U\cdot)=\{KEK_5,KEK_8\}$ of U_{λ_L} ,the user

$PK_1=\{KEK_1,KEK_2,KEK_4,KEK_s\}$ of

u_1, $PK_4=\{KEK_1,KEK_2,KEK_5,KEK_{11}\}$ of u_4. So the

$KEK_8\in KEK(U_{\lambda_L})\,\mathrm{I}\ PK_1$ and u_4 use

$KEK_5\in KEK(U_{\lambda_L})\,\mathrm{I}\ PK_4$ can be decrypted HDR attribute group. The $KEK_5\in KEK(U_{\lambda_L})\,\mathrm{I}\ PK_4$ the corresponding group key K_{λ_L}. u_t update their private key as follows:

$SK_t=(D=g^{(\alpha+r)/\beta},\forall\lambda_i\hat{\in}:D_i=g(w,l"(\lambda\cdot))^{r_i},D_i^I=(g^{r_i})^{K_{ji}})$

3) u_t uses new private key to decrypt the text CT `. For non-leaf node x Let x child node set $\{z_j\}$, $j\geq k_x$, calculated each child node z_j corresponding

$F_{Z_j}=DecryptNode(CT^I,SK,z_j)=e(g,g)^{rq_{Z_j}\{0\}}$

, and then select the k_x sub node F_{zj} as Lagrange Interpolation the polynomial interpolation nodes $F_x=e(g,g)^{rq_x\{0\}}$. For the root node R, the $A=DecryptNode(CT`,SK,R)=e(g,g)^{rq_R(0)}=e(g,g)^{rs}$. If the a u_t property set lambda meet the access tree T, the symmetric key $k_f=C/(e(C,D)/A)$.

4) u_t uses k_f decrypts the C_f data express file f.

IV. SECURITY ANALYSIS

The new program has data confidentiality. Proof: data cipher text C_f due to the symmetric key randomly generate k_f by the DO, k_f for the attacker is a random number, the NameNode know attribute group key, but `decryption CT k_f face decipher CP-ABE cryptosystem, so the new program with the confidentiality of the data cipher text. Set of attributes because the attacker cannot meet the access tree T, so cannot be calculated $e(g,g)^{\alpha s}$, and thus cannot be restored k_f, the property if the attacker does not have the property λ_i Group U_{λ_i} group key K_{λ_i} the

attacker is a random number, so the new program with key cipher text confidentiality.

If the attacker from the header message HDR group key, equivalent to decipher AES symmetric encryption system, to date, the AES and CP-ABE is a safe enough strength encryption algorithm, the new scheme has the header information Confidentiality. Shown in the foregoing, the new program has data confidentiality.

The new scheme has anti-collusion attack. Proof: we can see, the only attribute set to meet the access tree to calculate the $e(g,g)^{rs}$ (assuming a single unauthorized users access only part of the property to meet the tree, but several unauthorized users conspiracy attack by the construction of the access tree T.

When attribute set to meet the access tree T. By the user's private key generation algorithm shows that r corresponding to the different users, so each user's private key $D_i = g^r (W_{att(\lambda_i)})^{ri}$ different from each other. The multiple unauthorized users can only be calculated for each $e(g,g)^{rq_x(0)}$ at the value of the corresponding node x, but cannot get $e(g,g)^{rs}$, so that the attacker cannot decrypt the k_f. As the group key for external users is a random number, unauthorized collusion attack cannot attribute group the group key. Therefore, the new scheme has anti-collusion attack.

Before the new program has to confidentiality and after to confidentiality. Proof: The privileges revoked, NameNode the cipher text component transformation of the secret s corresponding secret
s `the corresponding cipher text assembly. When the user leaves the attribute group $U_{\lambda_i}$, the user know that the old group key $K_{\lambda_i}$, but do not know the group key update left $K`_{\lambda_i}$, s `is a random number, so the user knows $e(g,g)^{as}$, but the cipher text corresponding withdrawal of property $e(g,g)^{a(s+s`)}$ cannot be calculated, the new program has a forward secrecy. When a user joins the attribute group U_{λ_i}, new users to know updated group key $K`_{\lambda_i}$, but do not know the group key $K_{\lambda_i}$, randomness by s `shows new users to know $e(g,g)^{a(s+s`)}$, but has close before the property cannot be calculated text corresponding $e(g,g)^{as}$, so the new program has to confidentiality.

V. CONCLUSIONS

Cloud storage security issues affecting the development of cloud storage, reasonable and effective data security access control method can improve the trust of the users for cloud storage services. To solve the security issues of network features and data sharing features in cloud storage service, this paper proposes a data security access scheme storage Based on Attribute-Group in cloud, so that the data owners do not participate in the specific operation of the property and user rights, while the re-encrypted transfer to the NameNode-side, reduce the amount of computation and management costs of the client, ensure the confidentiality of user data, and also Achieve its purpose that the cipher text file sharing. Although the attribute-group-based scheme proposed in this paper has high security and reliability, but the efficiency of the implementation has yet to be improved which is our next step to research.

ACKNOWLEDGMENT

This work is supported by NSFC (Grant Nos. 61300181, 61272057, 61202434, 61170270, 61100203, 61121061), the Fundamental Research Funds for the Central Universities (Grant No. 2012RC0612, 2011YB01).

REFERENCES

[1] Hadoop.http://Hadoop.apaehe.org,2010.

[2] Qinyin91 L, Xiaolinl G, Deqin S, et al. Secure storage for cloud storage Strategy [J]. Journal of Computer Research and Development, 2011, 48: 240-243

[3] WANGLQ,ZHANG J,LVSW,etal.An efficient cryptosystem based hierarchical access control scheme and its analysis[J].Computer Engineering and Applications, 2005,33,7-10.

[4] RAYI,RAI,NARASIMHAMURTHIN.A cryptographic solution to
implement access control in a hierarchy and more[A].Proceedings of the Seventh ACM Symposium on Access Control Models and Technologies[C].Monterey, California, USA, 2007,5-16.

[5] BETHENCOURTJ,SAHAIA,WATERSB. Cipher text-policyattrib
ute-based encryption[A]. 2007 IEEE Symposiumon Security and Privacy(SP'07)[C]. Berkeley, California, USA. 2007.

[6] Borthakur D. The hadoop distributed file system: Architecture and design[J]. Hadoop Project Website, 2007, 11: 21.

[7] Liu X, Han J, Zhong Y, et al. Implementing WebGIS on Hadoop: A case study of improving small file I/O performance on HDFS[C]//Cluster Computing and Workshops, 2009. CLUSTER'09. IEEE International Conference on. IEEE, 2009: 1-8.

[8] Patterson D A, Gibson G, Katz R H. A case for redundant arrays of inexpensive disks (RAID)[M]. ACM, 1988.