Based on HDFS Safety Measures Construction Of Secret Storage Cloud

V.Raghavan
Assistant Professor
Rajiv Gandhi College of Engineering & Technology
Pondicherry
Ragh.durga2006@gmail.com

V.Ramsundhar
Sastra University
Thanjavur
sundar_varathu@yahoo.com

Abstract

With the growth of business, an enterprise would like to make its PSC (private storage cloud) approach an infrastructure service in a Partner/Public Cloud. In such PSCs, there are some new security issues. First, how to isolate the data stored in the PSC from internal and external attackers; Second, how to make secure intra-cloud data migration within an enterprise, Third, how to secure inter-cloud data migration between the PSC and the Partner/Public Cloud. In this paper, we propose an architecture of enforcing security services on the layer of HDFS, including Data Isolation Service, Secure Intra-Cloud Data Migration Service, and Secure Inter-Cloud Data Migration Service. Finally, a prototype has been implemented based on HDFS by our three custom security policies, and the time cost is given and evaluated.

Keywords

Private cloud storage; partner cloud; public cloud; isolation; intra-cloud data migration; inter-cloud data migration

I. INTRODUCTION

Storage Cloud is an emerging technology that leverages commodity hardware tied together to appear as a single storage device by the software, such as a cluster application, distributed file system(DFS) and grid computing[1]. HDFS (Hadoop DFS) [2] is an open source project and was designed to reliably store very large files across machines in a large cluster, and it has aroused concern from researchers and developers all over the world. Nowadays, there is growing interest for many enterprises to build a Private Storage Cloud (PSC) based on DFS like HDFS [3,4,5], which can be owned and managed internally. However, with the growth of business, an enterprise would like to make its PSC approach the infrastructure service (e.g. storage service) in a Partner/Public Cloud [6], which could greatly relieve them from management and maintenance of storage infrastructure. But in such a PSC, some new security issues may arise.

Data Isolation issue appears when a company has put some of its data on the PSC and others on Partner/Public cloud. In this case, there may be some risks during data access, 1) internal attackers may access unauthorized data of other sub-clouds(or storage clusters) in the cloud, which is separated by such attributes as departments, regions and security levels; or 2) external attackers in Partner/Public Cloud may intercept or tamper with the company’s data stored in both clouds. So, data isolation in PSC here mainly means enforcing access control between different sub-clouds, which may be in the private cloud or in the Partner/Public cloud. Moreover, encryption of the data in rest is an optional service for critical data protection in the storage cloud.

Intra-Cloud Data Migration issue may happen between different sub-clouds for data replication, load-balance, or restructuring within the PSC. Under the circumstances, there may be some risks during internal data migration, especially, 1) an attacker may make the data being migrated into a wrong location and make it accessible for some unauthorized users; or 2) a malicious user may fake an internal migration request from some legal user to initiate an unexpected migration and make the data inaccessible for the legal user. Thus, when this kind of migration happens, it should at first check if the source and target in the PSC are permitted to do the migration on the data or not.

Inter-Cloud Data Migration issue may occur between the PSC and Partner/Public cloud for business demand. During such external migration, it may bring some risks, for examples, 1) malicious intercept or modification risks may arise when sensitive/critical data are migrating between the two clouds; or 2) the availability of data is critical. For example, a fake response of successful migration from the target cloud will truncate the backup copies in the source cloud, leading to missing data. Thus, when such a migration happens, it should ensure that the authenticity of migration messages can be identified correctly and the security of data transfer can be enforced efficiently.

So, we propose the architecture of enforcing security services on the layer of HDFS in PSC, which includes Data Isolation Service, Secure Intra-Cloud Migration Service, and Secure Inter-Cloud Migration Service. The rest of this paper is organized as follows. In Section 2, we give the background and motivation. In Section 3, we propose and describe the security architecture in detail, and Section 4 is the prototype of implementation and Section 5 is our evaluation. And finally, Section 7 includes our conclusions and future work.

II. Surroundings & Enthusiasm

A. HDFS Architecture

HDFS is designed to reliably store very large files as a sequence of data blocks across machines in a large cluster (see Figure 1). In HDFS, the Name Node executes file system
Name space operations like opening, closing, and renaming files and directories, and determines the mapping of blocks to Data Nodes. The Data Nodes are responsible for serving read and write requests from the file system’s clients. The Data Nodes also perform block creation, deletion, and replication upon instruction from the Name Node. HDFS’s default replica placement policy is to put two-thirds of replicas on different nodes within the same rack to improve performance, and put the other third of replicas on randomly chosen nodes on other different racks to improve availability.

**B. PSC Based on HDFS**

Nowadays many enterprises would like to build its own PSC(private storage cloud) based on HDFS[3,4,5]. As depicted in Figure 2, this kind of PSC consists of sub-clouds separated by different attributes (e.g. departments, regions or security levels) and could extend to a Partner/Public Cloud when necessary. In such a PSC, it comprises three layers: HDFS, the bottom layer for managing a large amount of storage resource pools; SLA(Service Level Agreement), the middle layer for defining service norms and charge; and Cloud Service Interface, the upper layer for users to use cloud services, such as web portals, network disk, custom interfaces(e.g. REST, SOAP), and standard access protocols(e.g. FTP, BT). Commonly distributed file systems like HDFS, there are mainly following five services.

1) Fault-Tolerant Service: is responsible for the reliability of the cloud, including failure nodes detection, data replication policy, data recycling, etc.

2) Storage Service: includes name and data storage service. In the distributed file system for storage cloud, meta-data are stored in a name server, the large file being cut into blocks of same size data, and saved in data nodes. Mapping from files to blocks and locating blocks in data nodes is the task of name server.

3) Node Service: is responsible of the configuration of the system and management of the cloud’s nodes, including the initialization of the system, adding and deleting data nodes automatically.

4) Data Transmission Service: makes use of encryption and parallel transmission techniques based on small data blocks with multi-replications to meet the requirements of security and high QoS of the data transmission.

5) Load Balance Service: is to spread data between two or more hardware resources dynamically, in order to get Optimal resource utilization and minimize response time.

**C. Enthusiasm**

For security, early versions of HDFS implements a permissions model for files and directories that share much of the POSIX model. Recently, Owen [7] chose Kerberos over SSL for authentication and access control to enhance HDFS’s security, and has released new version of HDFS so far. But it still does not solve above security risks depicted in Section 1. In order to enhance the security of PSCs, we introduce three security services as follows in the layer of HDFS. Firstly, using any one traditional access control policy appears to have a major problem. It is possible that the policy cannot fit the hybrid PSC with varied of access requirements and unexpected access will happen. So we add Data Isolation Service into the architecture. Secondly, as a user of the company in the cloud, it is totally possible that the user makes data replication or restructuring safely as his duty changes. In our solution, we provide Secure Intra-Cloud Data Migration Service for users to replicate or shift their data to some nodes with expected attributes safely within the PSC. Thirdly, in regard to data that has confidentiality or privacy, users do not want them to be lost or intercepted during the migration between clouds, so in this solution, we provide Secure Inter-Cloud Data Migration Service to enable user to transfer their data safely in parallel by distributing a series of temporary tickets for related data nodes.

**III. SAFETY MEASURES DESIGN OF PSC BASED ON HDFS**

We now describe the security architecture of the PSC based on HDFS. As shown in Figure 3, the architecture mainly includes three new security services: Data Isolation Service, Secure Intra-Cloud Data Migration Service and Secure Inter-Cloud Data Migration Service. These services mainly depend on above five common services described in Section II. First, when clients want to read/write a file, the Storage Service would query Data Isolation Service at first. Second, when Fault-Tolerant Service, Load Balance Service, or Node Service initiates data migration within the cloud, the Secure Intra-Cloud Data Migration Service would determine if the migration is permitted. Third, when a user wants to transfer his data from one cloud to another one through Data
Transmission Service, it must send a request to Secure Inter-Cloud Data Migration Service at first. Especially, internal or external data migration essentially is to read the data from source nodes and write them to target nodes. So, the latter two migration services lie on the Data Isolation Service.

A. Data Isolation Service

Data Isolation Service is responsible for maintaining the security of data access (e.g. read or write) and storage in the PSCs. (Note, if a user tries to access a company’s data stored in the Public/Partner Cloud, it also relies on this service in the PSC). It mainly consists of four separate modules: access policy management, and access decision and authorization, access protocol security and private data encryption. The following are their functional descriptions respectively.

Access Policy Management (APM) is in charge of the management and configuration of a flexible access control policy (see section IV) enforced in the architecture, including labeling data blocks by some attributes with hierarchical relationship (e.g. tags for different countries, regions, or departments); labeling data nodes (racks or clusters) by some attributes (e.g. tags for different security levels); defining permission rules by logical expressions of object tags, and assigning role-permission relationships. This job can only be done in the master server by the administrator of the PSC. Any other users cannot modify and bypass this policy.

Access Decision and Authorization (ADA) is in charge of secure access decision and authorization enforced in the master server, including making decision for an access request according to the predefined policies in the APM, and producing a token with authorized rights. This token will be encrypted with a key shared between the master sever and related data nodes, which may be in PSC or other Clouds.

Access Protocol Security (APS) is in charge of securing access protocol between the master and the client, and the client and the data nodes, in other words, enhancing security for data transmission management in the PSC. They would negotiate some measures (e.g. token encryption algorithm and key used in ADA) to protect the token from malicious attackers to eavesdrop and to access the data accordingly.

Private Data Encryption (PDE) is an optional service for critical data protection based on the client of the cloud system. To make the data stored in the cloud more securely, the data would be encrypted with the help of TPM embedded in the client’s platform, without requiring storage of keys within the cloud or third-party key management[8]. Only the user himself in the trusted platform can decrypt the data with the key. And also, the client can decide whether to migrate and share the key with users in other trusted platforms.

B. Secure Intra-Cloud Data Migration Service

Secure Intra-Cloud Data Migration Service is responsible for ensuring the security of data replication and shifting caused by the requests from administrators or normal users, and those from load balance service, fault-tolerant service, or node service automatically within the PSC. It mainly consists of two separate modules: internal migration policy management and secure target selection. Following are the functional descriptions of them respectively.

Internal Migration Policy Management (IMP) is in charge of the management and configuration of a flexible internal migration control policy (see section IV) enforced in our architecture, including labeling data nodes by some attributes with hierarchical relationship (e.g. tags for different countries, regions, or departments): labeling data nodes (racks or clusters) by some attributes (e.g. tags for different security levels); defining permission rules by the mapping function (block tags node tags). This job can only be done in the master server by the administrator of the PSC. Any other users cannot modify and bypass this policy.

Secure Target Selection (STS) is in charge of judging the migration request referring to predefined migration security policies in IMP. Normally, when a request is associated with some expected target nodes (e.g. a request from load balance service), this module would do selection for a right node from them, depending on whether it is in accord with the mapping function. But when a request is only associated with an expected attribute about target nodes (e.g. a request from users or administrators), it would at first get all permitted nodes by permission rules and then further select them according to operations in the storage service.

C. Secure Inter-Cloud Data Migration Service

Secure Inter-Cloud Data Migration Service is responsible for the security of data migration between the PSC and Partner/Public Cloud. It will be also a new service in current HDFS for PSC. It mainly consists of two separate modules: source and target attestation, and data transmission protocol security. Following are their functions respectively.

Source and Target Attestation (STA) is in charge of attesting the trustworthiness between two clouds, in the way of remote attestation mechanism proposed by TCG[9]. Remote attestation is aimed to conclude whether to trust an attesting system based on testing and verifying each measurement list entry independently. As the two clouds are transparent to each other, the source and the target during attesting mean the master servers in two clouds. As remote attestation will cost much time, STA is optionally enforced here, only when the migrated data is critical to its owner.

Data Transmission Protocol Security (DTP): is in charge of securing data transmission during inter-cloud migration, enhancing communication management in HDFS. Before the migrating, the source and the target will negotiate some measures over SSL to ensure the data’s confidentiality, integrity and authenticity. SSL-based protocol would choose

---

**Figure 3. Security Architecture of PSC based on HDFS**
an encryption algorithm and keys for confidentiality, a Hash algorithm for integrity, a MAC algorithm for authentication, and temporary authorization tickets for related data nodes of two clouds to do secure transfer in parallel. Only when the destination has received all data correctly, it returns a success response to the source. The source then clears up the replicas of original blocks. If the response shows the migrating failed, the DTP should interact with the fault tolerant service in order to restore the source to its original secure state.

IV. IMPLEMENTATION

We implemented three security policies based on HDFS for PSC: 1) a flexible access control policy based on RBAC (Role-Based Access Control) and CW (Chinese-Wall); 2) a label-based intra-cloud data replicating and restructuring policy, 3) an temporary-ticket based parallel inter-cloud data transmission policy. Each policy works on a set of security labels, or permission expression definition separately. The assignment of security labels and the definition of permission expressions are both administrative tasks.

A. Security Policies

RBAC & CW Based Access Policy: is to ensure that 1) the data owned by a user of one sub-cloud wouldn’t be accessed by any unauthorized users of other sub-clouds in the PSC, and 2) the data owned by one firm in PSC wouldn’t be crossed by other ones, and this policy could be easily customized to fit variant requirements in enterprises.

This policy is proposed based on RBAC (Role Based Access Control) and CW (Chinese-Wall policy). It defines a kind of organization label (CW-org), which is assigned to all users and data of the enterprise in the PSC, ensuring that any user associated with a different CW-org cannot access the data in the PSC, and vice versa. It defines a series of roles (RBAC-roles) for subjects (e.g. users) and security tags for objects (e.g. files) both with inheritance relationship. All data in a sub-cloud are assigned to the same set of tags. For any role r associated with subject s, only when all tags of object o meet the logical expression of permission defined by the role manager of the PSC, could the data be accessed by s.

Here, we assume that this organization label would be identified or transformed to a kind of unique security label, which can differentiate the entities of the enterprise from those of other enterprises in the same Partner/Cloud Public.

Label-Based Intra-Cloud Data Replication & Shifting Policy: is to ensure that 1) the data owned by one user would always be on the nodes with the expected attribute and accessible for legal users before and after migrating, and 2) illegal internal migration request can be recognized and restricted before the unexpected migrating happens.

This label-based policy is to label data blocks and data nodes, and handing the data block with a group of expected users. Each data block will be associated with the users’ common attribute tags, such as departments, regions and countries. Each data node will be assigned to a series of tags, which could meet the security requirements of different users, for example, security levels. For any migration request on data block o, only when all tags of o meet predefined mapping function of permission (block tags node tags), could the data be migrated to one of data nodes with the expected security attribute and the location accessible for legal users. Once a migration request cannot meet expected mapping, it will be rejected directly and following unexpected migration does not occur.

Temporary-Ticket based Inter-Cloud Data Transmission Policy: is to ensure that 1) data blocks could be recognized and protected during the migrating from the source cloud to target cloud, 2) the source and target’s trustworthiness before the migrating could be attested, and 3) the parallel transmission could be protected and completed reliably.

This policy demands: First, before the migration process, the master servers of the source and target clouds should attest each other to ensure their trustworthiness. Second, the master servers of two clouds should do security negotiation over SSL and return a series of encrypted temporary tickets for all source data nodes to hold (This ticket is associated with the source node ID, the source path of being migrated data block, and a one-time usage information). Once the target master server decrypts the ticket and verifies its validity, then it will return the target data node ID for the data block transmission. Third, the data block transmission is in parallel between the source and target data nodes directly and the data block would be encrypted with the key during the previous phase of security negotiation over SSL.

B. Policy Management

To minimize security-related code complexity in HDFS, the policy management translates an XML-based policy into a binary one that is both systems independent and efficient to use. And in order to enhance overall system performance, during the HDFS prototype starting up, all binary policies are loading in memory, which will store and maintain the policy-dependent Rules configuration pools (Rule Pools) and Tag configuration pools (Tag Pools). In the system running time, if the subject needs to refer to related Tags (see section A), they can be accessed directly in the Rule Pools/Tag Pools in memory without time-consuming I/O operations.

C. Policy Enforcement

1) AOP Method

We implemented security services on the layer of HDFS by using Aspect-Oriented Programming (AOP) [10] method. First, by using AOP method with the property of jvm, we can add permission check logic into the source code without too much modification. Second, AOP model can reuse the code which reduces duplicate code design as much as possible. Third, AOP could provide the pluggable feature, because users can weave the security logic into system or remove it from the system via specific command.

As shown in Figure 4, "Security Policy Weaving Points" specifies where security checks are needed in the HDFS-based system. But it does not in itself provide any security checks and processing logic, it has only two functions, the first function is to declare AOP cut points, making "weaver" know which executing points need to weave into the implementation of additional security checks and processing logic; the second one is to intercept the context parameters.
required for the use of security modules according to the interface needs of security policy, and to determine whether to continue or interrupt the original logic, depending on the results returned by the security module and execution environmental contexts. The core security framework consists of three sub-modules, the interface part of the policy states a series of abstract interfaces, which is implemented only by concrete classes inherited from them. "Abstract mechanism" is the abstract class of these policy interfaces, its main task is to provide some default process without involving any security-related logic. But the real implementation of security checks and processing logic is in the "concrete mechanism”. According to the parameters obtained information and program execution context information, developers can customize corresponding security check logic and processing logic. "Application security check points” refer to security check logic enforced in policy-dependent commands or applications.

2) Reference Monitor

In our prototype based on HDFS [2], above security services have been implemented, strictly separating policy enforcement from policy management with AOP method.

Data Isolation Service: In HDFS, enhanced isolation control is implemented as the method after the ACL (access control list) check logic weaved in read(), write() methods and secure access protocols between Client and Name Node, Client and Data Nodes.

In addition, PDE is implemented as an optional service embedded in the Client of HDFS. With the cryptographic function provided by TPM and jTSS (An API used for communicate with TPM), we can upload the data with an encrypted format and also download with a decrypted format. The whole encryption and decryption processes are transparent to users. Table 1 lists a sample AOP point where we intercept code and plant our code for data protection.

Secure Intra-Cloud Data Migration Service. First, in HDFS, the blocks of a file are always replicated by Name Node when a Data Node becomes unavailable, a replica becomes corrupted or a hard disk on a Data Node fails. Second, as a cluster can easily become imbalanced when a new data node joins the cluster or when one data node is full but the other is free, it often redistributes some data blocks for rebalancing. Therefore, in our prototype, secure intra-cloud migration service was implemented in the function for block replication and cluster rebalancing, which mainly make target selection by the mapping policy (see section A) from the security tags of data blocks to those of Data Nodes. Table 1 also lists the sample AOP points where Secure Intra-Cloud Data Migration Service planted.

Secure Inter-Cloud Data Migration Service. In HDFS, distcp is a tool used for large inter-cluster copying, and it is implemented as a MapReduce job where the work of copying is done by the maps that run in parallel across the cluster. So, as shown in Table 1, the secure inter-cloud migration control was implemented in the distcp and related Map Reduce methods, including the secure distcp command, secure communication protocols between two Name Nodes and between Data Node and Name Node, and block transmission protection. During the communication of two Name Nodes, platform attestation is implemented but optionally called.

V. EVALUATION

We deployed a private cloud and a partner cloud with above security services based on HDFS and evaluated their effects on the original cloud system based on HDFS.

For data isolation control, there are three major aspects which will have influence on read/write time cost, including policy computing, ticket creation and ticket en/decrypting. By testing separately, we find that the overhead is all among the data access time is, e.g. 0.15ms~15.96ms, which is almost negligible, as the data access time is, e.g. 5889ms for reading a 64M data(1block) and 67847ms for writing a 512M data(8 blocks) in the system (see figure 5).

For the intra-cloud data migration from block replication,
a file uploaded by a user. But the number of Data Nodes does less impact on the cost of security decision as each block’s uploading is decided by the Name Node. And the average cost of this operation is less than 5%. For rebalancing, the migrated data capacity depends on the average usage rate (the used capacity/total capacity in all Data Nodes), and the sum of all Data Nodes’ excess capacity over the average usage rate is our migrated data capacity. So, the size of data capacity has obvious influence on the cost of migration. And the average cost of rebalancing is about 10% as it computes the migrated data capacity in all Data Nodes (see figure 6).

In HDFS, distcp is a typical operation for inter-cloud migration. For secure distcp, the time cost is mainly from three aspects: SSL negotiation, temporary ticket distribution and verification, file encrypting/decrypting for secure transmission regardless of platform attestation, which is optional for critical data migrating. By testing in our prototype (see figure 7), the temporary ticket distribution and verification only need 0.15-3ms overhead, and SSL negotiation 890ms. The cost of en/decrypting operation is larger than the cost of above two phases, e.g. 290726ms for a 384M big file. But comparing to the confidentiality of data, this cost is worth for most users.

Table 2. Time consumption of up/downloading a file

<table>
<thead>
<tr>
<th>File size</th>
<th>1K</th>
<th>1M</th>
<th>32M</th>
<th>128M</th>
<th>1G</th>
</tr>
</thead>
<tbody>
<tr>
<td>Upload Cost (%)</td>
<td>21.4</td>
<td>21.1</td>
<td>14.1</td>
<td>4.54</td>
<td>1.23</td>
</tr>
<tr>
<td>Download Cost (%)</td>
<td>23.6</td>
<td>20.7</td>
<td>11.7</td>
<td>8.5</td>
<td>3.7</td>
</tr>
</tbody>
</table>

Compared to HDFS without a PDE service, the time we will use for PDE service (when uploading a file) includes the time it costs to generate a symmetric key, the time it costs for encryption of the file upload filter, and the time it costs for encryption of the symmetric key and persistence of the symmetric key. Especially, the time cost for encrypting the file upload filter is offset by the time cost for uploading the file. Table 2 shows the time we cost when we upload various kinds of data. With the PDE service comparing the uploading without a PDE service. Table 2 shows the time we cost when we download files of various amounts. As shown in the tables, the average overhead of PDE service is about 13%, and this is acceptable for the private data protection.

VI. CONCLUSION AND FUTURE WORK

This paper describes security issues on data isolation, intra-cloud data migration and inter-cloud data migration under the environment of a Private Storage Cloud (PSC) extended with a Partner/Public Cloud. The security solutions based on the HDFS layer, with master/slave architecture, for the PSC are proposed. And an implementation of these security services is given with AOP method. The performance analysis of them proves the efficiency of the security design. In future, we will make our security services compatible with other cloud storage software systems.

ACKNOWLEDGMENT

This work is supported by IBM SUR Project 2009 and the National Natural Science Foundation of China under Grant No.60873238, 61073156, and 61070237.

REFERENCES