A Proposed Scheme for Secured Data Access in Cloud Computing using Diffie-Hellman and Encryption Algorithm

1Kiran Raj R, 2Chirag Gor
1,2School of Information Technology, VIT University, Vellore - 632014, Tamilnadu, India
e-mail: svce.kiranraj73@yahoo.com’, chirag.gor2011@vit.ac.in

Abstract - Cloud computing is a fifth generation concept after the Mainframes, personal computers, client server computing and the web. Since “over the long term, absent of other barriers, economics always wins”. Cloud computing provides cheapest way of delivering and supporting applications in the vendors perspective and also for the customer perspective. Customers receive their demanded and required applications on time for which they pay without installation and management cost. This could be achieved in cloud computing through its layered architecture as services named as Infrastructure as a service (IaaS), Platform as a service (PaaS) and Software as a Service (SaaS).

Cloud computing can be seen as pool of virtualized computer resources in the perspective of customer. Since it is a open system architecture security is major concern. Cloud also vulnerable to almost all security issues arouse like physical layer security, network level security, virtualization level security and management level security. In network security we have to maintain Confidentiality, Integrity and Authentication which are the primary issues for securing data sharing in cloud computing environment. For maintaining these issues several algorithms and techniques have been proposed but still they have not fulfilled their purpose they meant for. The proposed model could be able to handle secured data access using encryption algorithms (MD5, DOUBLE encryption and for shared symmetric key exchange we are using D-H) will authenticate both the end of communication entities and also secure the data by scrambling them.

Keywords: Cloud, Diffie Hellman Algorithm, MD5, Public Key Encryption.

I. INTRODUCTION

Cloud computing is an Internet based computing environment where you pay only for resources that you use. It is the delivery of computing as a Service rather than a product, whereby shared-resources, software, and information are provided to computers and other devices as utility (like the electricity grid) over a network (Internet).

In the last few years, Cloud Computing has been grown in the vision of Business concept and it is one of the fast growing segments in the IT industry. Now a days, due to Recession-hit most of the IT companies highly realizing that simplicity by tapping into to the cloud is the better solution and they can gain faster access to their business applications or drastically boost their information resources, all at negligible cost. Cloud computing also faces the data-security challenges because as more and more information on individual and companies is placed in the cloud, concerns are beginning to grow about just how safe an environment it is. In recent past, various commercial models are developed that are described by “A as service (AaaS)” where A could be Software, Platform, Infrastructure, Storage etc. Some of the emerging Cloud Computing infrastructures are Microsoft Azure, Amazon's EC2 and Google App Engine.

II. PROPOSED SYSTEM MODEL

As data owners store their data on external servers, whenever the clients try to access their data due to increase in demand and concerns for authentication, data confidentiality and access control. Apart from confidentiality and privacy breaks, the external servers could also use part of the data or whole for their economic gain hence tarnishing the data owner market or even bringing financial losses to the data owner. Cloud servers are usually operated by the commercial (third party) vendors the data owners placing their data likely to be outside the trusted domain of the users, this originate to the concerns. The proposed cryptographic access control model as shown in Figure.1, which is also considered as the system model in our proposed scheme. The model shown in Figure.1 has three entities Data owner (DO), Cloud administrator (CA) and User. The Data Owner places the data on the Cloud administrator which user wants to access. As the Cloud administrator is un-trusted environment, Data Owner places the encrypted data on Cloud administrator. Once DO receive the data request from the user, It sends the required Key & Certificates to the user. Later user can present its certificates to the CA in order to access its data. Actually data is stored in encrypted form upon successful verification by CA allows to access the data.

The system model shown in Figure.1 ensures confidentiality, integrity and authentication, but the problem arises with this model is that the data owner should always be in online when the user wants to access the data. The key & certificate management between all the communications parties is also cumbersome. In some situations, a data owner with poor computing capabilities becomes bottleneck. Generally in
traditional access control architecture usually assume the data owner and the servers storing the data are in the same trusted domain, where the storage servers are modelled as an omniscient reference monitor entrusted to define and enforce access control policies.

III. OUR PROPOSED SCHEME
The scenario shown in Figure 2. Where cloud administrator must be trust-worthy, so that there is no guarantee it does not do any harm to the sensitive data. Which also demand data owner should be always be in online to access the new user request. Here in order to access the cloud data we must implement controlled access mechanism.

In order to achieve the secure and efficient data access from the cloud, we uniquely combining the capability based access control with cryptography. In this section we provide the pseudo-code of our algorithm. We also provide an example of our approach along with notations and data structures used in the algorithm.

A) Overview of our Proposed Scheme
The user request which are sent to the data owner and the encrypted data which is obtained from Cloud Architecture must be secured from any security attacks during the transmission. Our proposed scheme will explain in detail how we can achieve the controlled access mechanism and security requirements. In figure 3 we are illustrating our proposed scheme with example using educational scenario. Here data owner can be teacher, who posts the students grade into the cloud and user can any student who register for the particular course able to view his/her grade from the cloud.

When Data Owner sends the data to the Cloud Administrator he encrypts the Data using MD5 algorithm instead of SHA-1(128 bit) for maintaining the integrity of data. Although MD5 is not so much efficient than SHA-1 (160 bit) in security but here we are again encapsulating the message digest which is created by MD5 algorithm. Overall, it gives more security than SHA-1. It provides confidentiality and integrity between Data Owner and User. If new user comes then Data Owner adds new entry in capability list and Data Owner assigns some numbers for accessing the data (for example 1.for read only 2.for write only 3. both) .These numbers are mentioned in the AR field in (User Id,File Id,AR).All the thing sent from Data Owner to the Cloud Administrator in Encrypted form. First he encrypts with secret (private key) then encrypts with the public key of the Cloud Administrator.

Figure 2. Scenario represents unsecure and inefficient access to the cloud data.

In order to receive and store both the Encrypted data and Capability List CA.

B) Algorithm to update data and Capability list at CA
Figure 4 illustrates the procedure that CA will updates both encrypted data and capability list which are send from the data owner. When a new user wants to access the cloud data, it needs to send a registration request to the data owner with USERId, FILEId, nonce, Timestamp and access rights required for the data file.

When Data Owner sends the data to the Cloud Administrator他 encrypts the Data using MD5 algorithm instead of SHA-1(128 bit) for maintaining the integrity of data. Although MD5 is not so much efficient than SHA-1 (160 bit) in security but here we are again encapsulating the message digest which is created by MD5 algorithm. Overall, it gives more security than SHA-1. It provides confidentiality and integrity between Data Owner and User. If new user comes then Data Owner adds new entry in capability list and Data Owner assigns some numbers for accessing the data (for example 1.for read only 2.for write only 3. both) .These numbers are mentioned in the AR field in (User Id,File Id,AR).All the thing sent from Data Owner to the Cloud Administrator in Encrypted form. First he encrypts with secret (private key) then encrypts with the public key of the Cloud Administrator.

Figure 3. Proposed Scheme with respect to Education scenario.

Figure 4. Algorithm to receive and store both the Encrypted data and Capability List CA.

Figure 5 provides the pseudo code for this procedure. Once data owner receiving request from the user, it adds an entry into the capability list if it is a valid request. For simplicity here we are assuming that data owner has a separate procedure to check the user requests. The data owner sends the updated capability list and encrypted data which belongs to the particular user with all the key parameters needed at user for decrypting the data files to Cloud Administrator. CA now up-
dates its own capability list and sends the registration acknowledgment to the user over encryption, i.e., here you can notice we are encrypting twice using \( EN_{\text{PUBUSR}} \). Which meets our critical design goal of our proposed scheme, the key parameter required for decryption are still confidential to user (although these have come via CA). The nonce and timestamps parameter in the request and reply message serve the purpose of replay and man-in-middle attack avoidance.

C) Algorithm to register a new user

The below Figure 5, Used to register the new user in data owner as well as in cloud administrator. This encryption ensures the confidentiality of the data between CA and the user as CA is not able to read the content of the user’s data files. Here we are assuming that session key will get generated between CA and user that remain valid for some predefined time, which ensures the secure data transfer over a period of time and that also eliminating the use of D-H key exchange for every data access request. Once user receives the encrypted response from the CA can decrypt the data and try to calculate the digest by using the hash function. Because, the digest that is attached with the data is then compared with the newly calculated digest so that it will ensure the integrity of the data.

Figure 5. Algorithm to register a new requested user.

<table>
<thead>
<tr>
<th>Notations</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUB</td>
<td>Public Key</td>
</tr>
<tr>
<td>PR</td>
<td>private key</td>
</tr>
<tr>
<td>PUBUSR</td>
<td>private key of user</td>
</tr>
<tr>
<td>PUSCR</td>
<td>private key of cloud administrator</td>
</tr>
<tr>
<td>PUBCA</td>
<td>public key of cloud administrator</td>
</tr>
<tr>
<td>PUBOK</td>
<td>public key of the owner</td>
</tr>
<tr>
<td>EN</td>
<td>Encryption</td>
</tr>
<tr>
<td>DK</td>
<td>Symmetric key of owner</td>
</tr>
<tr>
<td>HS</td>
<td>Hash Algorithm</td>
</tr>
<tr>
<td>Cap List</td>
<td>Capability List</td>
</tr>
<tr>
<td>CA</td>
<td>Cloud Administrator</td>
</tr>
<tr>
<td>StorageArray</td>
<td>Array that stores capability and data files</td>
</tr>
<tr>
<td>AX</td>
<td>Access Rights</td>
</tr>
<tr>
<td>USER</td>
<td>User identity</td>
</tr>
<tr>
<td>DL</td>
<td>File identity</td>
</tr>
<tr>
<td>DS</td>
<td>Data Set</td>
</tr>
<tr>
<td>DO</td>
<td>Data Owner</td>
</tr>
<tr>
<td>OS</td>
<td>Object</td>
</tr>
<tr>
<td>Encodor</td>
<td>Encoded form of the Object</td>
</tr>
</tbody>
</table>

Diffie-Hellman Parameters

\[ \text{X} \text{, } \text{X}^a, \text{X}^b, \text{X}^{ab}, \text{X}^{abc}, \text{X}^{abcd}, \text{X}^{ab} \text{abcX}^{ab} \]  

\[ X \text{, } X^a, X^b, X^{ab}, X^{abc}, X^{abcd}, X^{ab} \text{abcX}^{ab} \]

IV. ANALYSIS OF OUR PROPOSED SCHEME

In this section, we are analysing our proposed scheme with respect to security properties as well as performance with respect to scalability and strength of cryptographic primitives.

A. Confidentiality

Confidentiality is a thing which is primary necessity of security. So in this model, confidentiality is maintained between Cloud Administrator and Data Owner. Here Cloud Administrator cannot see the data of Data Owner which he has uploaded in the cloud because, it is double encrypted (secret key & public key encryption) which is more secure than any other algorithm as shown in figure 4.

B. Authentication and Integrity

Authentication between Data Owner (DO) and CA (cloud administrator) is done through encrypted data and capability list using private key of Data Owner and authentication between CA and User is provided by the Keys & Certificate. Integrity is provided by MD5 as a hashing algorithm when Data Owner uploads the data to the Data Administrator then hash is generated by the Data Owner, the same thing Data Owner does for User also. When User receives the data from Data Administrator, he checks hash is matching or not. If it is then User accept otherwise complain to Data Owner that he got tampered data in this way integrity is maintained.

Figure: 7 Comparison between algorithms on the basis of Security Level

We have analysed Double key encryption with MD5, D-H Key exchange and Public Key encryption in fig:8 on the basis of security level. Double encryption gives more robust performance.

C. Capability based access control

In our proposed model, Data Owner have rights to create modify and delete the content of data which is placed inside the cap_list. All Users are categorised into the cap_list. They are divided horizontally (Row Wise) in the access matrix (AM). We have divided matrix on the basis of User not on the basis of file group which is going to be accessed. If we follow this scheme then it gives better performance because files (DATA) are accessed frequently by the assigned User as compared to other User.

To compute keys in public key encryption is very hard in comparison to the DH key exchange as shown in figure: 9. That’s why we have used DH key exchange in our proposed model between User and Data Owner.
V. CONCLUSION

The above paper presented the way that how can we provide security when data files are moving between Data Owner and Cloud Administrator as well as Cloud Administrator and user so that he can get desired data securely. For this, above proposed algorithms have been used like D-H Key exchange algorithm and Public Key exchange are used. DH Key exchange algorithm is used for secret key exchanging between User and Data Owner. And with the help of Double Key encryption data security is provided between Data Owner and Cloud Administrator. Finally, proposed scheme is robust and fully secure for data accessing in the cloud.

REFERENCES