IJARnD
INTERNATIONAL JOURNAL OF ADVANCE RESEARCH AND DEVELOPMENT
(Volume3, Issue4)
Available online at: www.ijarnd.com

Anonymous attribute-based encryption using privacy-preserving, shoulder surfing

Shital S. Salokhe[1], Nishita N. Patil[2]

[1]*Student, Ashokrao Mane Group of Institutions, Kolhapur, Maharashtra*
[2]*Professor, Ashokrao Mane Group of Institutions, Kolhapur, Maharashtra*

ABSTRACT

Cloud services is a great opportunity for the users to enjoy the on-demand cloud applications without considering the local infrastructure limitations. Several schemes have been proposed for access control of outsourced data in cloud computing. Security is the primary obstacle that prevents the wide adoption of this promising computing model. Identity-based encryption (IBE) is the sender of a message can specify an identity such that only a receiver with matching identity can decrypt it. In the KP-ABE, a ciphertext is associated with a set of attributes, and a private key is associated with a monotonic tree, which describes this user's identity. A user can decrypt the ciphertext if and only if the access tree in his private key is satisfied by the attributes in the ciphertext. However, the encrypter does not have entire control over the encryption policy. Furthermore, when a re-encryption occurs, all of the users in the same system must have their private keys re-issued so as to gain access to the re-encrypted files, and this process causes considerable problems in implementation. All problems and overhead are solved in the CP-ABE. In this, ciphertexts are created with an access structure, which specifies the encryption policy, and private keys are generated according to users' attributes. A user can decrypt the ciphertext if and only if his attributes in the private key satisfy the access tree specified in the ciphertext. Hence the encrypter holds the encryption policy. Also, issued private keys will not be modified till the whole system reboots.

Keywords: *Cloud computing, Shoulder surfing, Privacy-Preserving, Anonymous attribute-based encryption.*

1. FOUR MODULES HAVE BEEN PROPOSED IN THE SYNOPSIS TITLED AS,

I) File Management
II) Shoulder Surfing
III) Data De-Duplication
IV) File Retrieval

Out of the four modules mentioned in synopsis, work on the a four module, File management, Shoulder surfing, Data de-duplication, File retrieval module is completed.

1. File Management

On Cloud user can upload any kind of file, and the size of the file. Accepting file uploads from users has become extremely easy. The FileUpload control allows the user to browse for and select the file to be uploaded, providing a browse button and a text box for entering the filename. Once, the user has entered the filename in the text box by typing the name or browsing, the SaveAs method of the FileUpload control can be called to save the file to the disk.

2. Shoulder Surfing

Shoulder surfing is direct observation techniques, such as looking over someone's shoulder, to get information. Shoulder surfing will be provided within the system with attribute based encryption. Shoulder surfing is an effective way to get information in crowded places because it's relatively easy to stand next to someone and watch as they fill out a form. For example, someone might shoulder surf when you are entering your computer password, ATM pin, or credit card number.

3. Data De-duplication

De-duplication will be added where the server will store only a single copy of each file, regardless of how many users asked to store that file, depending upon the disk space of cloud servers. When you upload the same copy of the file this match the contents of file then it shows the notification “File already exist”, here it is actual work of de-duplication match with the file size.

Avoiding duplication of files used MD5 algorithm. One of the purposes of creating a hash from source data is to provide a way to see if data has changed over time, or to compare two values without ever working with the actual values. In either case, you need to compare two computed hashes, which is easy if they are both stored as hexadecimal strings However, it is quite possible that they will both be in the form of byte arrays.

4. File Retrieval

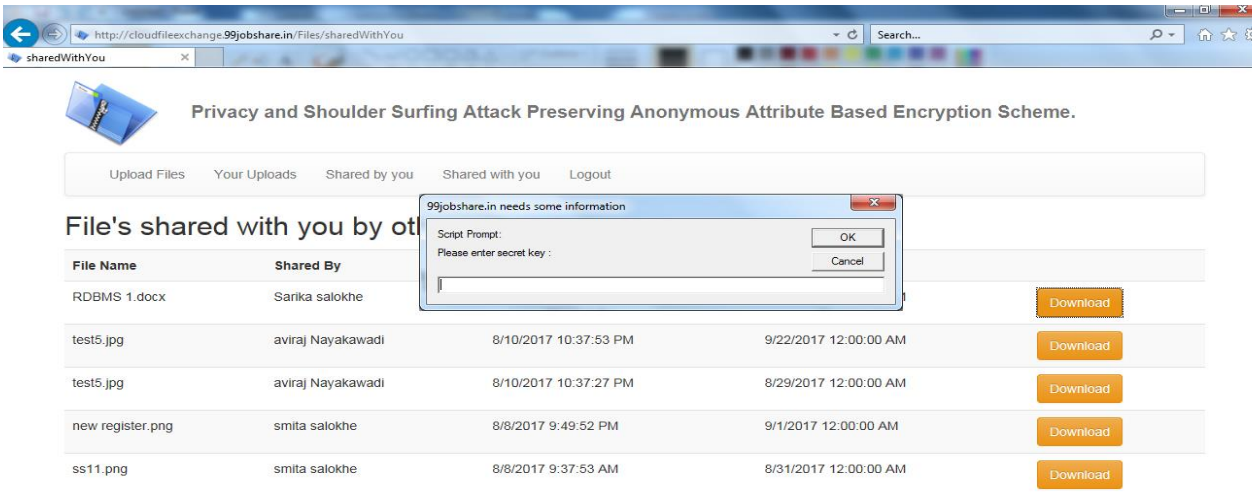
In this module, Decryption is used for file controlling (e.g. reading, modification, and deletion). It takes as input the public key PK, a ciphertext CT, and a private key SK_u, which has a set of attributes A^u and corresponds to its holder’s GID_u. A^u satisfies any tree in the set $\{T_p\}_{p\epsilon\{0,....,r1\}}$, the algorithm returns a message M or a verification parameter. The bigger part is that saving files directly to the database has a large overhead the easy way to handle files and databases is to save the files directly to the file system, and keep the path in the database along with other file-related data such as the user id that uploaded the file.

2. EXPERIMENTAL SETUP

This application implemented in ASP.Net. and MySQL database. Install Visual Studio 2008 or Visual Studio 2010 .Install MySQL database on local machine MySQL database admin tool that allows you to create a database and run SQL statements. Download and install MySQL Connector. MySQL is used because it is free open-source database management system. MySQL is a relational database management system based on SQL – Structured Query Language. The application is used for a wide range of purposes, including data warehousing, e-commerce, and logging applications.

2.1 File Shared with you by others

When user receives some files and he/she wants to download this files then click on download button after that shows one small window for enter secret key which is send through email and download the file .



Privacy and Shoulder Surfing Attack Preserving Anonymous Attribute Based Encryption Scheme.

File's shared with you by ot

99jobshare.in needs some information

Script Prompt:

Please enter secret key :

File Name	Shared By			
RDBMS 1.docx	Sarika salokhe			Download
test5.jpg	aviraj Nayakawadi	8/10/2017 10:37:53 PM	9/22/2017 12:00:00 AM	Download
test5.jpg	aviraj Nayakawadi	8/10/2017 10:37:27 PM	8/29/2017 12:00:00 AM	Download
new register.png	smita salokhe	8/8/2017 9:49:52 PM	9/1/2017 12:00:00 AM	Download
ss11.png	smita salokhe	8/8/2017 9:37:53 AM	8/31/2017 12:00:00 AM	Download

3. RESULT ANALYSIS

3.1 Comparative analysis

A. Encryption and Upload Time

To evaluate ABE performance, we have measured time taken to encrypt and upload the file on the SQL Database by using different file sizes—1 KB, 2 KB, 5 KB, 50 KB, 100 KB. To capture time used for encryption and uploading the file, we have used the following code.

```
Strmgst et;
et=DateTime.Now.Millisecond.ToString();
initet=Convert.ToInt16(et);
initst=Convert.ToInt16(st);
intft=iet-ist;
Label1.Text=ft.ToString();
```

We have written this code in the essential class files of my project and then execute it. This code will take the current system time and then run the code and the query in it and then captures the time which it takes for doing so. Then it subtracts that system time from the time it captures and returns the result in the label which is placed for that purpose.

B. Key Generation Time

The Key generation time returns the time to generate the keys. I have used same files for performing tests in order to capture the time taken to generate the secret key. Using this key only the users will be able to download the files. The file sizes which I have used for testing are 1KB, 2KB, 5KB, 50KB, 100KB.

C. Download and Decryption Time

Same files were used for performing tests for capturing time taken to download and decrypt these files using both proposed system (ABE) and existing system.
-Encrypted file is downloaded from file storage service to local machine.
-Requested file is decrypted using private key of the user.

From the table, it is clear that the proposed system is having the encryption time with different number of authorized users (No. of Authorities). Graphical representation of the result is as shown in the following figure. Blue bars shows encryption time. X-axis represents the No. of Authorities and Y-axis represents the time.

Table 1 Encryption Time

Encryption Time	No. of Authorities
00.04ms	2 ABAES authorities
00.08ms	4 ABAES authorities
00.10ms	6 ABAES authorities
00.12ms	8 ABAES authorities

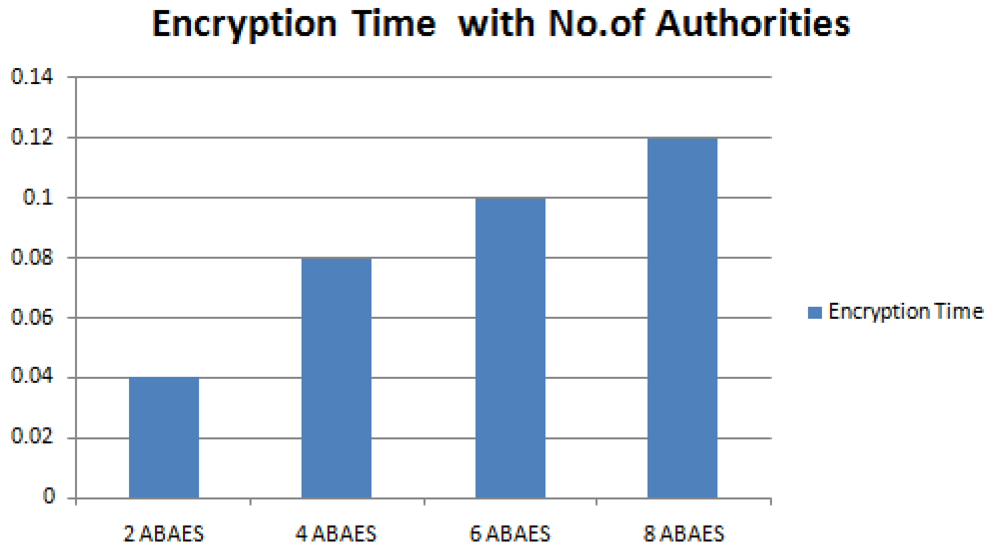


Figure 1: Encryption Time with No. of Authorities

Graphical representation of the result is as shown in the following figure. Blue bars show Encryption time. X-axis represents the No. of Authorities (No. of authorized users) and Y-axis represents the time.

Table 2 Decryption Time

Dcryption Time	**No. of Authorities**
00.04ms	2 ABAES authorities
00.08ms	4 ABAES authorities
00.10ms	6 ABAES authorities
00.12ms	8 ABAES authorities

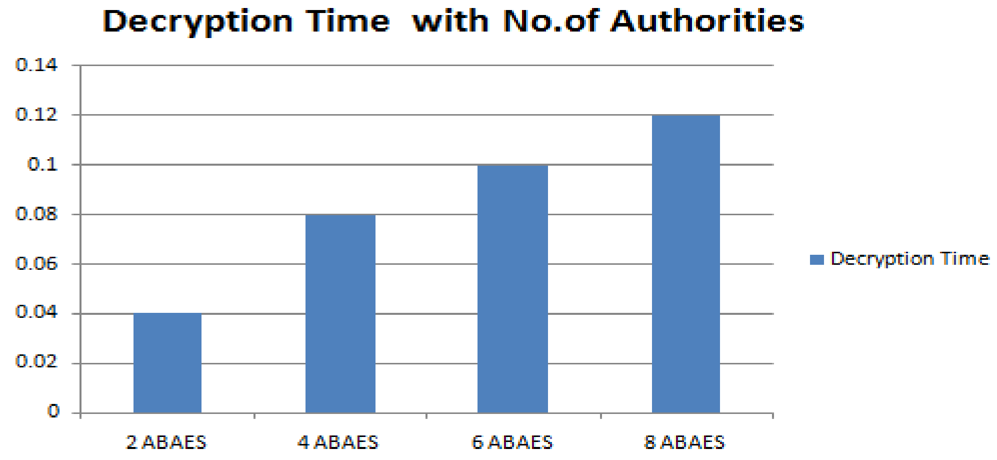


Figure 2: Decryption Time with No. of Authorities

Graphical representation of the result is as shown in the following figure. Blue bars show Decryption time. X-axis represents the No. of Authorities and Y-axis represents the time.

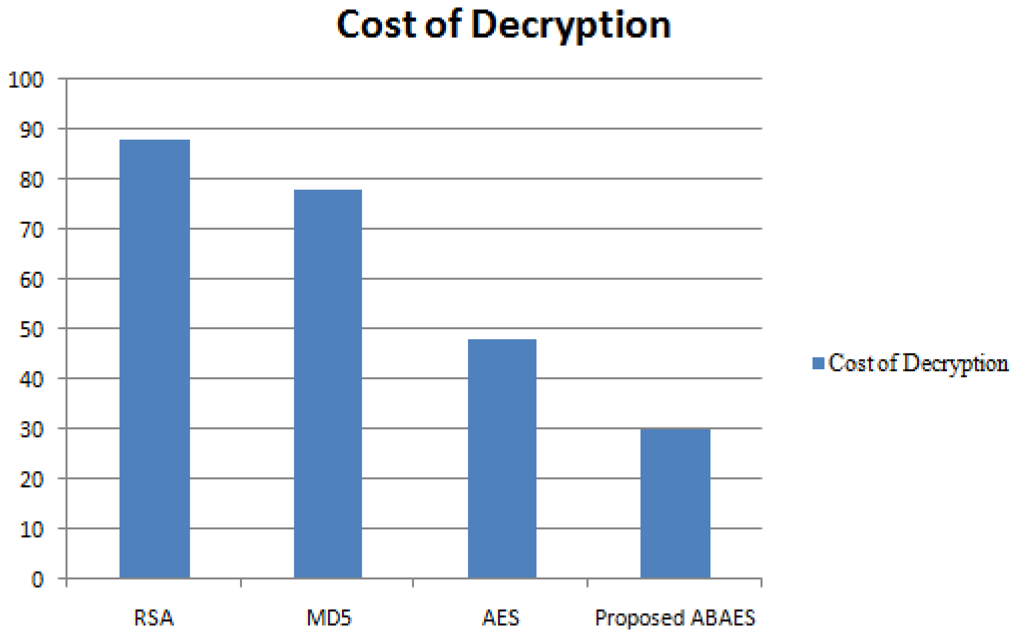


Figure 4: Cost of Decryption

Figure 3and Figure 4 bar graph based results depicts that the proposed approach is relatively better on multiple parameters and effective than the other approaches as shown in the results. The proposed ABAES based approach is effectual.

5. REFERENCES

[1] Techo Jung, Xiang-Yang, "Control Cloud Data Access Privilege and Anonymity With Fully Anonymous attribute-Based Encryption", IEEE Transaction 2015.
[2] M satish kumar, B Uday Kumar, Ch. Arun Kumar, "Attribute Based Data Sharing with Attribute Revocation to Control Cloud Data Access", International Journal of Computational Science, Mathematics and Engineering, February-2016.

[3] Praveen N.R and Renju Samuel, "Enhanced Efficient User Revocation Mechanism on Top of Anonymous Attribute Based Encryption", International Journal of Emerging Technology in Computer Science & Electronics, AUGUST 2016.

[4] P. Mell and T. Grance, "Draft NIST Working Definition of Cloud Computing, National Institute of Standards and Technology", USA, 2009.

[5] A. Mishra, R. Jain, and A. Durresi, "Cloud Computing: Networking and Communication Challenges", IEEE Communications Magazine, vol. 50, no.9, pp, 24-25, 2012.

[6] Muhammad Yasir Shabir, Asif Iqbal, Zahid Mahammad, and Ataullah Ghafoor, "Analysis of Classical Encryption Techniques in Cloud Computing", ISSN 1007-0214 09/10 pp102-119 Vol. 21, Number 1, February 2014.

[7] K. Hwang and D. Li, "Trusted Cloud Computing with Secure Resources and Data Coloring, IEEE Internet Computing", vol. 14, no. 5, pp. 14-22, 2010.

[8] J. Chen, Y. Wang, and X. Wang, "On-Demand Security Architecture for Cloud Computing, Computer", vol. 45, no. 7, pp. 73-78, 2012.

[9] Y. Zhu, H. Hu, G. Ahn, and M. Yu, "Cooperative Provable Data Possession for Integrity Verification in Multi-cloud Storage", IEEE Transactions on Parallel and Distributed Systems, vol. 23, no, 12, pp. 2231-2244, 2012.

[10] H. Wang, "Proxy Provable Data Possession in Public Clouds", IEEE Transactions on Services Computing,[online] ieeexplore.ieee.org 2012.

[11] Ming Li, Shucheng Yu, Yao Zheng, "Scalable and Secure Sharing of Personal Health Records in Cloud Computing Using Attribute-Based Encryption", 2013.

[12] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing", IEEE Transactions on Parallel and Distributed Systems, vol. 22, no. 5, pp.847-859, 2011.

[13] C. Wang, K. Ren, W. Lou, J, Lou, "Toward Publicly Auditable Secure Cloud Data Storage Services", IEEE Network, vol. 24, no.4, pp.19-24, 2010.

[14] L. A. Dunning and R. Kresman, Privacy Preserving Data Sharing With Anonymous ID Assignment, IEEE Transactions on Information Forensics and Security, vol. 8, no. 2, pp. 402-413, 2013.

[15] Jain Zhao, Haiying Gao and Junqi Zhang, "Attribute-Based Encryption for Circuits on Lattices", ISSN 1007-0214 05/13 pp463-469 Vol. 19, Number 5, October 2014.

[16] Xinhua Dong, Ruixuan Li, Heng He, Wanwan Zhou, Zhengyuan Xue and Hao Wu, "Secure Sensitive Data Sharing on a Big Data Platform", ISSN 1007-0214 08/11 pp72-80 Vol. 20, Number 1, February 2015.

[17] Allim Swami, "Privacy Preserving Data Sharing With Anonymous ID Assignment Using AIDA Algorithm", IJCERT, Vol. 1, Issue 1, PP 23-29, July 2014.