International Journal of Engineering Science and Advanced Technology (IJESAT)

IJESAT (Enriching the Research)

Open Access Research Article

Volume: 23 Issue: 07

July, 2023

A PRACTICAL ATTRIBUTE-BASED DOCUMENT COLLECTION HIERARCHICAL ENCRYPTION IN CLOUD COMPUTING

Mrs. S. Nagavali [1], R. Narasimha Rao [2], Ch. Krishna Sri Sai [3], T. Vamsi Kiran [4], S.V.V. Sai Saran [5]

[1] Associate Professor, Department of CSE, Ramachandra College of Engineering, Eluru, A.P
[2,3,4,5] UG Students, Department of CSE, Ramachandra College of Engineering, Eluru, A.P

ABSTRACT

Ciphertext-policy attribute-based encryption can provide fine-grained access control and secure data sharing to the data users in cloud computing. However, the encryption/decryption efficiency of existing schemes can be further improved when encrypting a large document collection. In this paper, we propose a practical Ciphertext-Policy Attribute-Based Hierarchical document collection Encryption scheme named CP-ABHE. By practical, we mean that CP-ABHE is more efficient in both computation and storage space without sacrificing data security. In CP-ABHE, we first construct a set of integrated access trees based on the documents' attribute sets. We employ the greedy strategy to build the trees incrementally and grow the trees dynamically by combining the small ones. Then, all the documents on an integrated access tree are encrypted together. Different from existing schemes, the leaves in different access trees with the same attribute share the same secret number, which is employed to encrypt the documents. This greatly improves the performance of CP-ABHE. The security of our scheme is theoretically proved based on the decisional bilinear Daffier Hellman assumption. The simulation results illustrate that CP-ABHE performs very well in terms of security, efficiency, and storage size of the Ciphertext.

1. INTRODUCTION

Cloud computing collects and organizes a large amount of information technology resources to provide secure, efficient, flexible, and on-demand services. Attracted by these advantages, more and more enterprise and individual users tend to outsource local documents to the cloud. In general, the documents need to be encrypted before being outsourced to protect them against leaking. If the data owner wants to share these documents with an authorized data user, they can employ any searchable encryption techniques or privacy-preserving multi-keyword document search schemes to achieve this goal. However, all these schemes cannot provide fine-grained access control mechanisms to the encrypted documents. Attribute-based encryption (ABE) schemes can provide complicated systems to diversify the data users' access paths. In ABE schemes, each document is encrypted individually and a data user can decrypt a document if her attribute set matches the access structure of the document. Existing ABE schemes can be divided into Key-Policy ABE (KP-ABE) schemes and Cipher text-Policy ABE (CP-ABE) schemes. Com- pared with KP ABE schemes, CP-ABE schemes are more flexible and suitable for general applications. In the following, we must analyze the existing ABE schemes in detail and further present the novelty and innovation of the CP-ABHE scheme proposed in this paper. For convenience, we choose the schemes as typical examples of the KP-ABE scheme and the CP-ABE scheme, respectively. LetG0 andG1 be two multiplicative cyclic groups of prime order *p*. Let *g* be a generator of G0 and *e* be a bilinear map, *e* V G0 _ G0 D G1. Further, let *H* V f0; 1g_! G0 is a hash function that can map an attribute string to a random element in G0. Assume that we need to encrypt a set of documents F D fF1; *F*2; _ _ _; *FN* g. Attribute set A D fA1; *A*2; _ _ _; *AM*g is the common attribute dictionary of both documents and data users. We further assume that document *Fi* is related to a set of attributes, denoted as *att*(*Fi*). We encrypt F in two phases. First, each document *Fi* is encrypted by a proper symmetric encryption algorithm with a unique content key *cki*. Second, all the content keys of F are encrypted by ABE schemes. Note that, both the cipher texts of *Fi* and *cki* are provided to data users. In the decryption process, data users need to _rest decrypt *cki* based on their attribute-related secret keys and then decrypt document *Fi* based on *cki*. In this way, the cipher text of *Fi* can be decrypted only by the data users who have the matched attributes with *att*(*Fi*). Considering that the _rest encryption phase does not fall in the scope of this paper, we focus on the second phase which is strongly related to the proposed scheme. To encrypt all the content keys of F, the KP-ABE scheme is executed as follows. For each content key *cki* with attribute set *att*(*Fi*) and access tree T , the public key is calculated as *PK* D f*e*(*g*; *g*)_; 8*j* 2 *att*(*cki*); *Tj* D *grj* g where _ is a random number in Z*p* and *rj* is a number randomly chosen from Z*p* for attribute *j*. Then the ciphertext of *cki* is calculated as *CTcki* D fT ; *cki* _ *e*(*g*; *g*)_*s*; 8*j* 2 *att*(*Fi*); *Ej* D *T s j* g where *s* is a random number in Z*p*. The above process must be executed *N* times to encrypt all the content keys. The total number of elements in the ciphertext can be calculated as *Ncip* D *N* C P*Ni* D1 j*att*(*Fi*)j, where j*att*(*Fi*)j denotes the number of attributes in *att*(*Fi*). To decrypt the cipher text of *cki*, a data user needs to store the secret key *SK* D f8*j* 2 *att*(*Fi*); *Dj* D *g qj*(0) *rj* g, where *qj*(*x*) is the polynomial of the leaf node in T corresponding to attribute *j*. To decrypt all the content keys, *N* secret keys for the *N* access trees need to be stored by a data user and the number of total secret values in the keys can be calculated as *Nsk* D P*Ni* D1 j*att*(*Fi*)j. It can be observed that *Nsk* increases with the increasing of documents number and we call this **the secret key expanding problem**.

To encrypt all the content keys of F, the CP-ABE scheme is executed as follows. For each content key *cki* with attribute set *att*(*Fi*) and access tree T, the public key is calculated as *PK* D f*h* D *g*_; *e*(*g*; *g*)_g, where _ and _ are random numbers in *Zp*. Then the scheme calculates the cipher text of *cki* as *CTcki* D fT ; *cki* _ *e*(*g*; *g*)_*s*; *C* D *hs*; 8*j* 2 *att*(*Fi*); *Cj* D *gqj*(0); *C*0 *j* D *H*(*j*)*qj*(0)g, where *qj*(*x*) is the polynominal of the leaf node in T corresponding to attributes *j*. Similar to KP-ABE, the above process is also executed *N* times to encrypt all the content keys. The total number of elements in the cipher text can be calculated as *Ncip* D 2 _ *N* C 2 _ P*Ni* D1 j*att*(*Fi*)j. *Ncip* greatly expands with the increasing of documents number. To decrypt the cipher text of *cki*, the secret key of a data user is calculated as *SK* D f*D* D *g* (_C*r*) _; 8*j* 2 *att*(*Fi*); *Dj* D *grH*(*j*)*rj* ; *D*0 *j* D *grj* g where *r* is a random number in Z*p* and *rj* is a random number chosen from Z*p* for attribute *j*. Boththe KP-ABE and CP-ABE schemes are impractical to encrypt a large document collection because of the following reasons.

First, the encryption process in both two schemes is executed *N* times, leading to high computation complexity. Second, there is a tradeoff between the size of the content keys' ciphertext and data users' secret keys. In KP- ABE, the number of secret values in a data user's secret key is extremely large for document collection, imposing a heavy burden on the data user. In CP-ABE, the size of the cipher text is extremely large. Consequently, the CP-ABE scheme increases the data transmission amount between the cloud server and data users, which is a huge challenge for the network. This is reasonable considering that the access structure of each document must be bedded into the cipher text or the secret keys. Third, decrypting the cipher text is also time-consuming considering each document is encrypted individually. Recently, Wang *et al.* attempted to improve the encryption efficiency and propose an agile hierarchy attribute-based encryption scheme named FH-CP-ABE. However, this scheme focused only on how to encrypt a set of documents that share an integrated access tree and hence it also cannot be directly employed to encrypt a document collection.

In this paper, we design an attribute-based document hierarchical encryption scheme named CP-ABHE which performs well in terms of computation and storage space efficiency. The scheme consists of two modules including integrated access tree construction and tree encryption. We must propose an algorithm to generate the integrated access trees for a document collection. The most important design goal of the algorithm is decreasing the number of integrated access trees which can greatly improve the encryption/decryption efficiency. Then, the documents that share an access tree are encrypted together. Each node in a tree is assigned a secret number which is used to encrypt the content keys of documents on the node. The secret numbers of the nodes are constructed in a bottom-up manner and it is different from the methods in KP-ABE, CP-ABE, and FH-CP-ABE schemes. In this way, the number of all the elements in the content keys' cipher text is smaller than 2 _ *N* and it is much smaller than that inKP-ABE scheme and CP-ABE scheme. In addition, we decrease the number of secret values in the keys stored by the data users compared with KP-ABE. To decrypt all the documents in F, only 2 _ jAj C 1 secret values must be stored by a data user, where jAj is the size of A. In conclusion, both the encryption/decryption efficiency and storage efficiency of CP-ABHE are very high.

The security of our scheme is proved theoretically and we evaluate the scheme's efficiency through a series of simulations. The contributions of this paper are mainly summarized as follows: An algorithm to construct the integrated access trees incrementally for the document collection is proposed and it can significantly decrease the number of access trees. _ A document collection hierarchical encryption scheme is proposed. All the documents that share an integrated access tree are encrypted together which can significantly improve the encryption/decryption efficiency. Moreover, the **secret key expanding problem** is solved properly. _ The security of CP-ABHE is theoretically proved and the effectiveness of the integrated access tree construction algorithm is analyzed in detail. In addition, a thorough comparison between CP-ABHE, KP-ABE, and CP-ABE in terms of encryption /decryption efficiency and storage space is provided.

The rest of this paper is organized as follows: We present the system model and preliminaries in Section 2. The detailed process of access tree construction is given in Section 3 and Section 4 discusses the scheme to encrypt the document collection. We analyze the security and efficiency of our scheme theoretically in Section 5. Section 6 evaluates the performance of the integrated access trees and the efficiency of CP- ABHE is analyzed and simulated in Section 7. In Section 8, the related work is provided and this paper is concluded in Section.

2. LITERATURE SURVEY

1. Cloud data sharing by K. Liang et al. In this paper, for the first time, we define a general notion for proxy re-encryption (PRE), which we call deterministic finite automata-based functional PRE(DFA-based FPRE).
2. Fine-grained two-factor access control for Web-based cloud computing services by J. K. Liu,M. H. Au, and J. Li. In this paper, we introduce a new fine-grained two-factor authentication (2FA) access control system for web-based cloud computing services.
3. Ciphertext-policy attribute-based encryption by J. Bethencourt, A. Sahai, and B. Waters. In several distributed systems a user should only be able to access data if a user possesses a certain set of

credentials or attributes. Currently, the only method for enforcing such policies is to employ a trusted server to store the data and mediate access control.

4. Arbitrary-state attribute-based encryption with dynamic membership by C.-I. Fan, V. S.-M. Huang, and H.-M. Ruan. Attribute-based encryption (ABE) is an advanced encryption technology where the privacy of receivers is protected by a set of attributes. An encryptor can ensure that only the receivers who match the restrictions on predefined attribute values associated with the ciphertext can decrypt the ciphertext.

   A hierarchical attribute-based solution for flexible and scalable access control in cloud computing by Z. Wan, J. Liu, and R. H. Deng. To realize scalable, flexible, and fine-grained access control of outsourced data in cloud computing, in this paper, we propose hierarchical attribute-set-based encryption (HASBE) by extending ciphertext-policy attribute-set-based encryption (ASBE) with a hierarchical structure of users.

3. EXISTING SYSTEM

Attribute-based encryption schemes have been widely researched in the literature. The fuzzy identity-based encryption (Fuzzy IBE) scheme proposed by Sahai and Waters is widely treated as the origin of attribute-based encryption (ABE). Sahai and Waters first employ the term`attribute-based encryption (ABE)" in the field of information security. Inspired by Fuzzy IBE, many ABE schemes are designed including KP-ABE schemes and CP-ABE schemes. Goyal *et al.* extend the Fuzzy IBE scheme and propose the key-policy attribute-based encryption (KP-ABE). Though KP-ABE can provide fine-grained access control, it restricts its attention to the monotone access structure only. Ostrovsky *et al.* constructs a KP-ABE scheme that allows a user's private key can be expressed in terms of any access formula over attributes. Further, they prove the scheme's security based on the decisional bilinear Diffie- Hellman assumption. Yang *et al.* propose a scheme that performs well in terms of both access structure expressivity and security. CP-ABE schemes are more flexible and suitable for general applications and many varieties of CP-ABE schemes have been proposed in the literature. In CP-ABE schemes, the access structures are embedded in the ciphertext and each data user is assigned a set of attributes. A data user can decrypt a ciphertext if and only if there be matched with each other. Pirretti *et al.* introduce a novel secure information management architecture based on ABE primitives. A policy system that meets the needs of different data users is designed and used to encrypt distributed file systems. The hierarchical ABE (HABE) scheme is proposed by combining a hierarchical IBE scheme and a CP-ABE scheme. HABE scheme can help enterprise users to efficiently share confidential data in cloud computing by simultaneously achieving fine-grained access control, high performance, practicability, and scalability. Zhu *et al* also propose a file-sharing scheme in cloud computing based on ABE and the security and efficiency of the scheme are evaluated. Li *et al*, provide a CP-ABE scheme with efficient data user revocation for cloud storage. KSF-OABE scheme integrates the keyword search function into the ABE scheme which can improve the search efficiency of ciphertexts. Though all the above-proposed schemes can be used in cloud computing, they are designed for encrypting a single document. They cannot be directly employed to encrypt a large document collection, because the encryption/decryption efficiency is low if we encrypt each file singly.

DISADVANTAGES

In the existing work, the system is less secure due to the lack of **CP-ABHE, and KP-ABE.** The system’s security is very less due to a lack of strong cryptography techniques.

4. PROPOSED SYSTEM

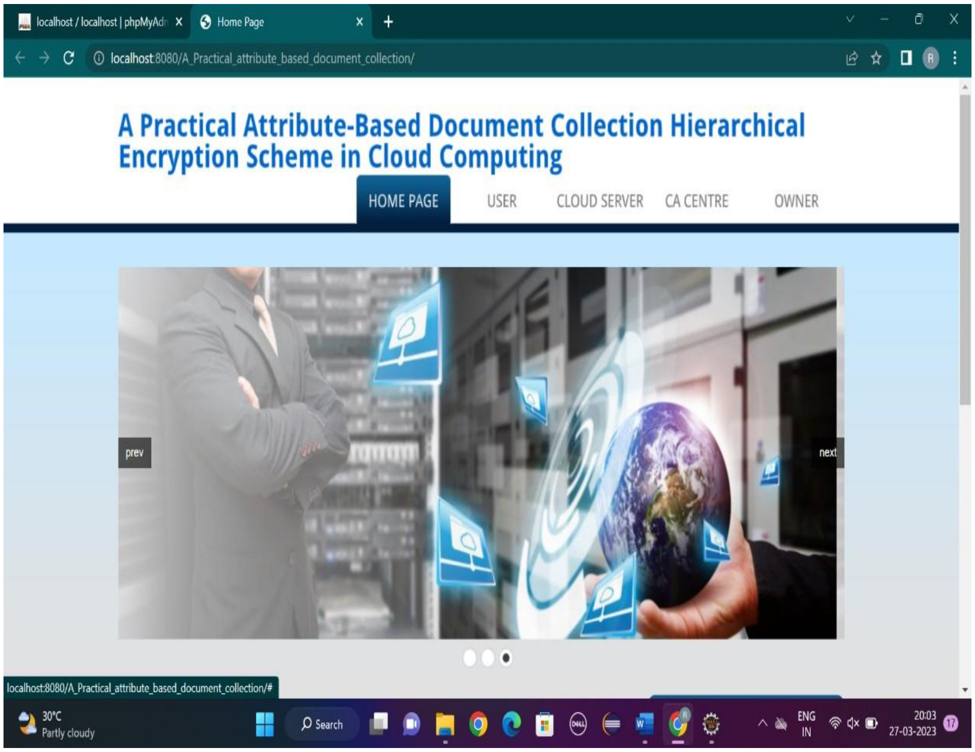
In the proposed system, the system designs an attribute-based document hierarchical encryption scheme named CP-ABHE which performs well in terms of computation and storage space efficiency. The scheme consists of two modules including integrated access tree construction and tree encryption. We first propose an algorithm to generate integrated access trees for a document collection. The most important design goal of the algorithm is decreasing the number of integrated access trees which can greatly improve the encryption/decryption efficiency. An algorithm to construct the integrated access trees incrementally for the document collection is proposed and it can significantly decrease the number of access trees. A document collection hierarchical encryption scheme is proposed. All the documents that share an integrated access tree are encrypted together which can significantly improve the encryption/decryption efficiency. Moreover, the **secret key expanding problem** is solved properly. The security of CP-ABHE is theoretically proved and the effectiveness of the integrated access tree construction algorithm is analyzed in detail. In addition, a thorough comparison between CP-ABHE, KP-ABE, and CP-ABE in terms of encryption/decryption efficiency and storage space is provided.

ADVANTAGES

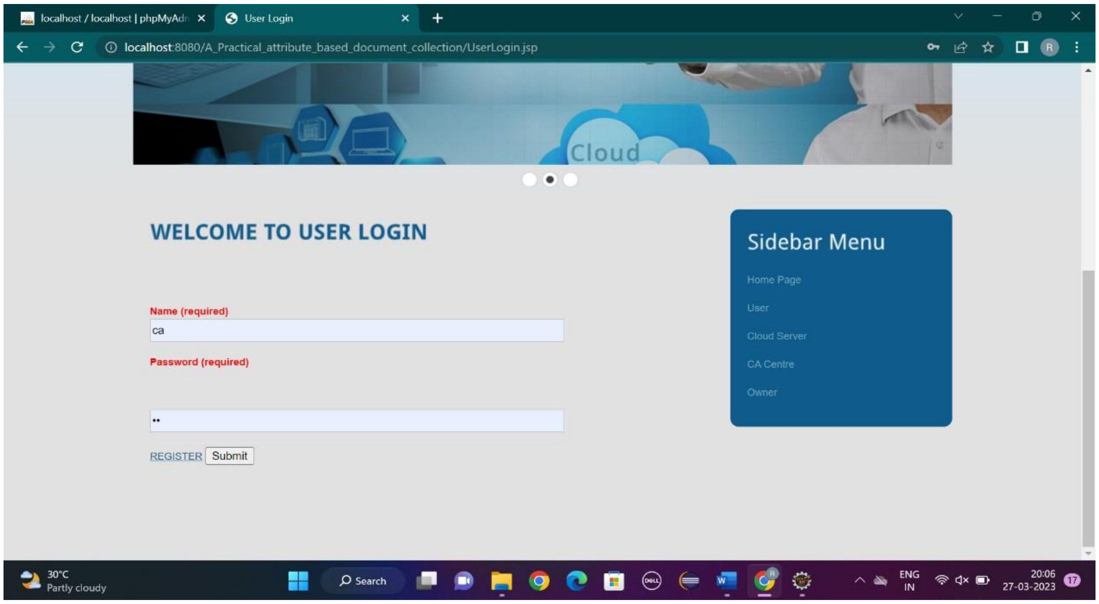
The system is implemented based on an Attribute-Based Encryption scheme which gives more security to data. The system is more secure due to CP-ABHE (Attribute Based Hierarchical Encryption).

5. RESULTS

HOME PAGE:


localhost / localhost | phpMyAdr
Home Page
localhost:8080/A_Practical_attribute_based_document_collection/
A Practical Attribute-Based Document Collection Hierarchical Encryption Scheme in Cloud Computing
HOME PAGE
USER
CLOUD SERVER
CA CENTRE
OWNER
prev
next
localhost:8080/A_Practical_attribute_based_document_collection/#
30°C
Partly cloudy
Search
ENG
IN
20:03
27-03-2023

USER LOGIN:


localhost:8080/A_Practical_attribute_based_document_collection/UserLogin.jsp

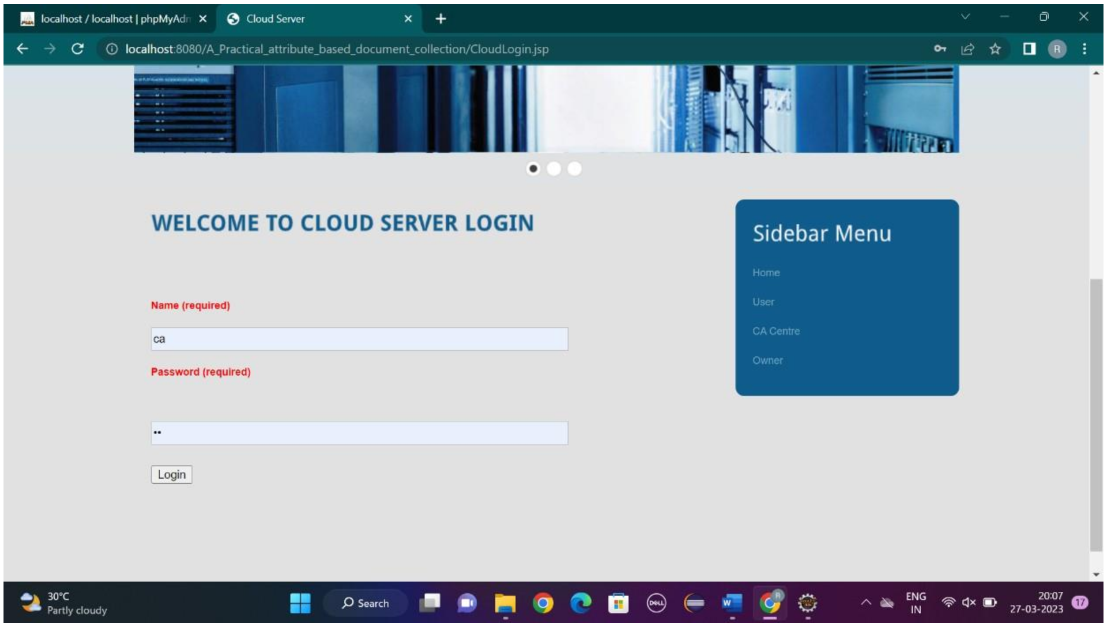
WELCOME TO USER LOGIN

Name (required)

ca

Password (required)

••


localhost:8080/A_Practical_attribute_based_document_collection/CloudLogin.jsp

WELCOME TO CLOUD SERVER LOGIN

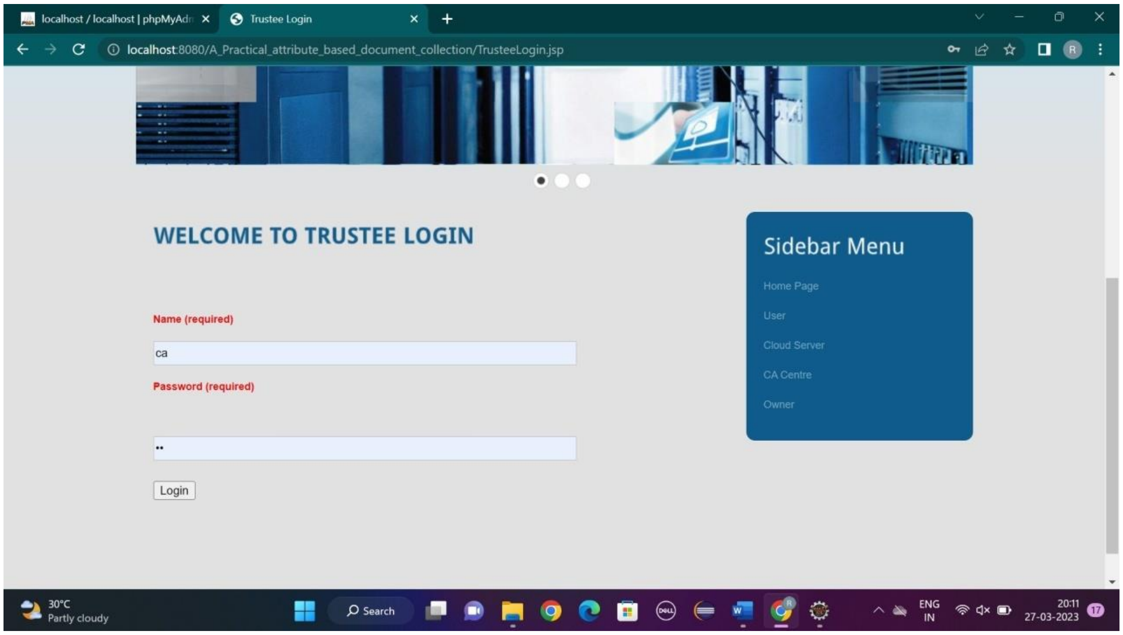
Name (required)

ca

Password (required)

••

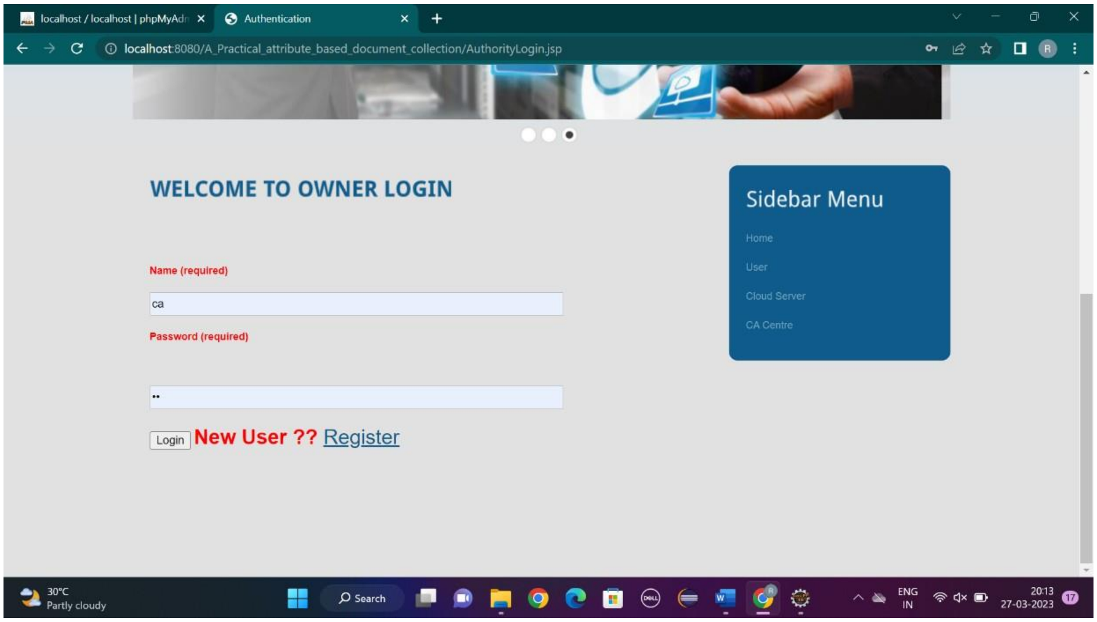
CA LOGIN:



WELCOME TO TRUSTEE LOGIN

Name (required): ca

Password (required): ••


localhost / localhost | phpMyAdn
Authentication
localhost:8080/A_Practical_attribute_based_document_collection/AuthorityLogin.jsp
Sidebar Menu
Home
User
Cloud Server
CA Centre
Login
30°C Partly cloudy
Search
ENG IN
20:13 27-03-2023

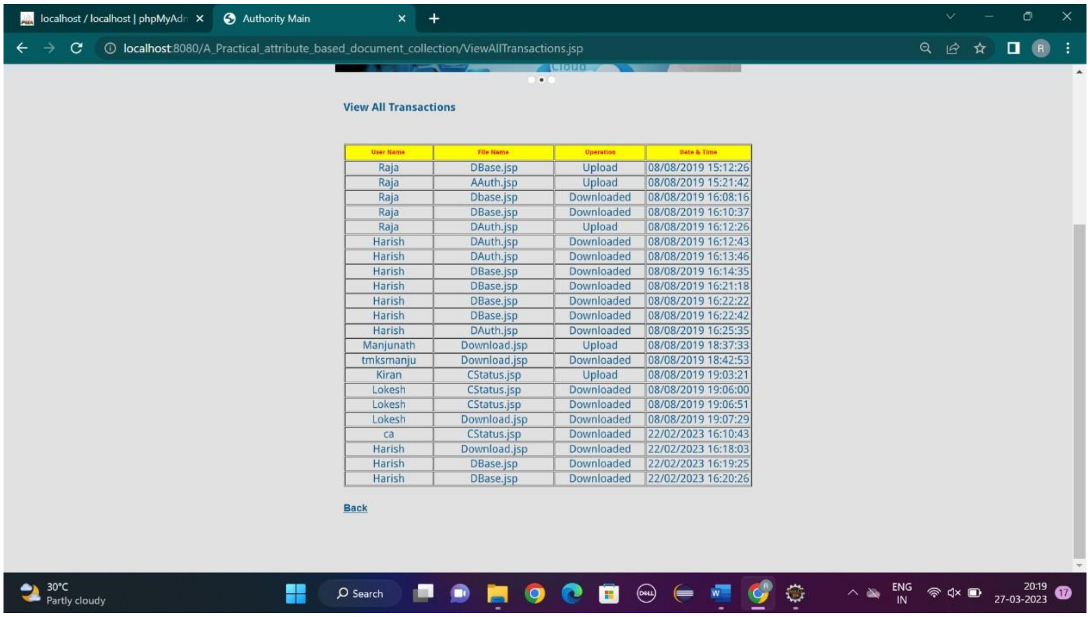
WELCOME TO OWNER LOGIN

Name (required): ca

Password (required): ••

New User ?? Register

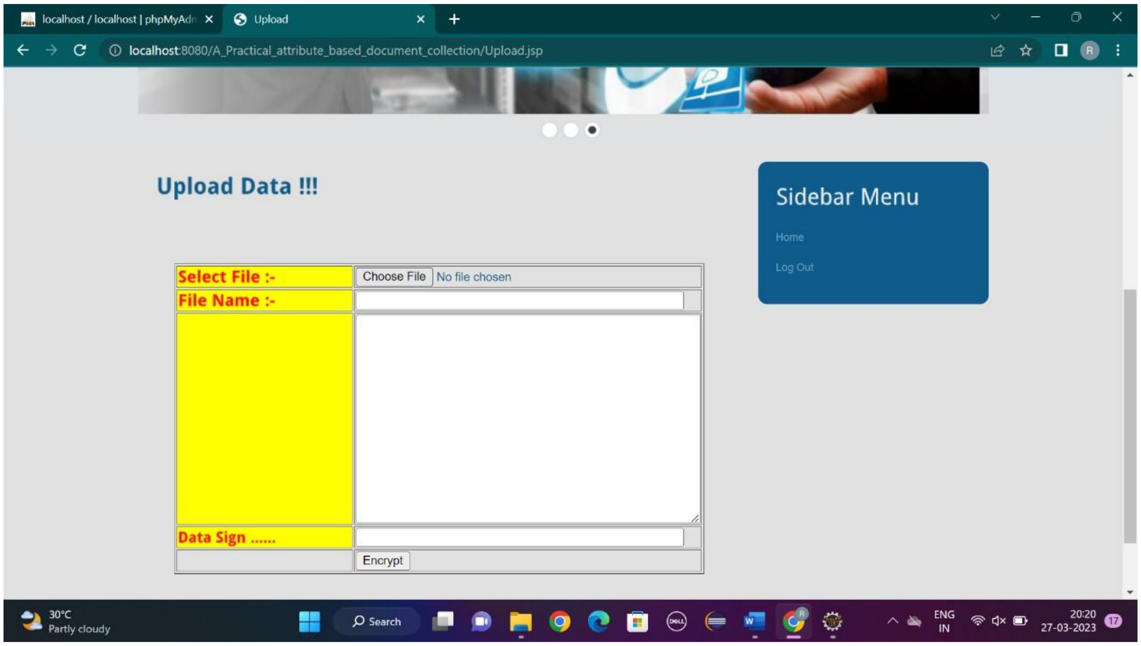
USES TRANSACTIONS:


localhost / localhost | phpMyAdm
Authority Main
localhost:8080/A_Practical_attribute_based_document_collection/ViewAllTransactions.jsp
30°C
Partly cloudy
Search
ENG
IN
20:19
27-03-2023

View All Transactions

User Name	File Name	Operation	Date & Time
Raja	DBase.jsp	Upload	08/08/2019 15:12:26
Raja	AAuth.jsp	Upload	08/08/2019 15:21:42
Raja	Dbase.jsp	Downloaded	08/08/2019 16:08:16
Raja	DBase.jsp	Downloaded	08/08/2019 16:10:37
Raja	DAuth.jsp	Upload	08/08/2019 16:12:26
Harish	DAuth.jsp	Downloaded	08/08/2019 16:12:43
Harish	DAuth.jsp	Downloaded	08/08/2019 16:13:46
Harish	DBase.jsp	Downloaded	08/08/2019 16:14:35
Harish	DBase.jsp	Downloaded	08/08/2019 16:21:18
Harish	DBase.jsp	Downloaded	08/08/2019 16:22:22
Harish	DBase.jsp	Downloaded	08/08/2019 16:22:42
Harish	DAuth.jsp	Downloaded	08/08/2019 16:25:35
Manjunath	Download.jsp	Upload	08/08/2019 18:37:33
tmksmanju	Download.jsp	Downloaded	08/08/2019 18:42:53
Kiran	CStatus.jsp	Upload	08/08/2019 19:03:21
Lokesh	CStatus.jsp	Downloaded	08/08/2019 19:06:00
Lokesh	CStatus.jsp	Downloaded	08/08/2019 19:06:51
Lokesh	Download.jsp	Downloaded	08/08/2019 19:07:29
ca	CStatus.jsp	Downloaded	22/02/2023 16:10:43
Harish	Download.jsp	Downloaded	22/02/2023 16:18:03
Harish	DBase.jsp	Downloaded	22/02/2023 16:19:25
Harish	DBase.jsp	Downloaded	22/02/2023 16:20:26

Back


localhost / localhost | phpMyAdm
Upload
localhost:8080/A_Practical_attribute_based_document_collection/Upload.jsp
30°C
Partly cloudy
Search
ENG
IN
20:20
27-03-2023

Upload Data !!!

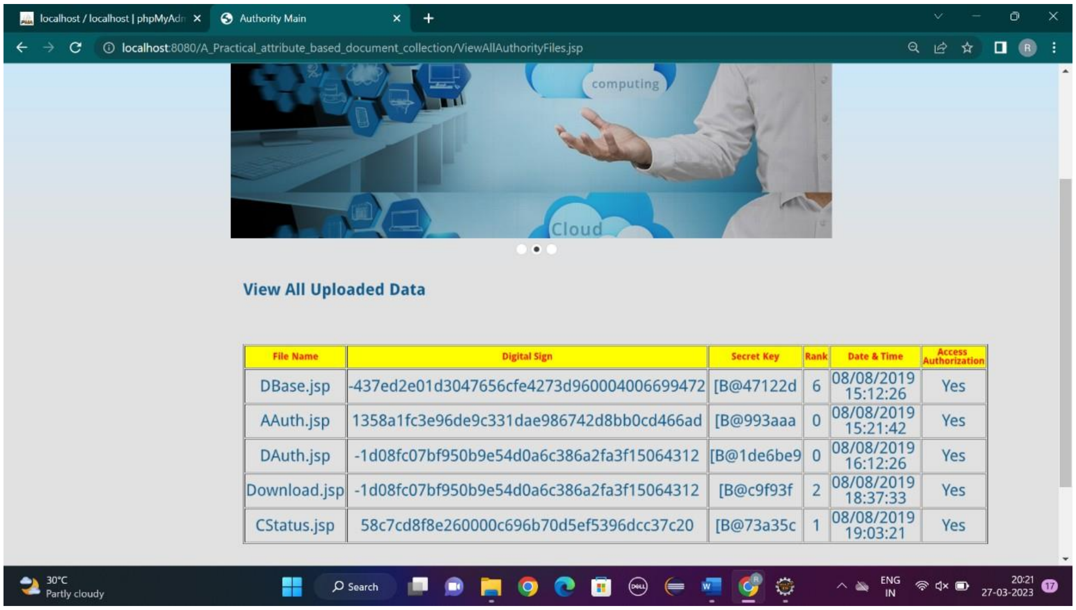
Select File :-	Choose File No file chosen
File Name :-	
Data Sign	
	Encrypt

Sidebar Menu

Home

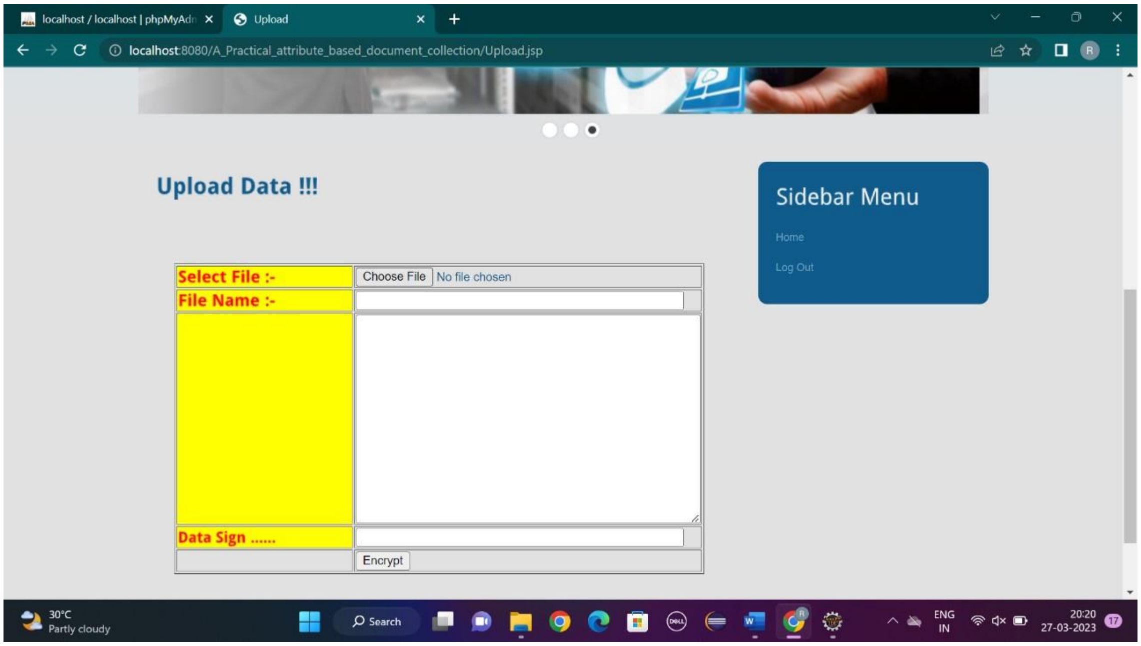
Log Out

UPLOADED DATA:


localhost / localhost | phpMyAdm
Authority Main
localhost:8080/A_Practical_attribute_based_document_collection/ViewAllAuthorityFiles.jsp
computing
Cloud
30°C
Partly cloudy
Search
ENG
IN
20:21
27-03-2023

View All Uploaded Data

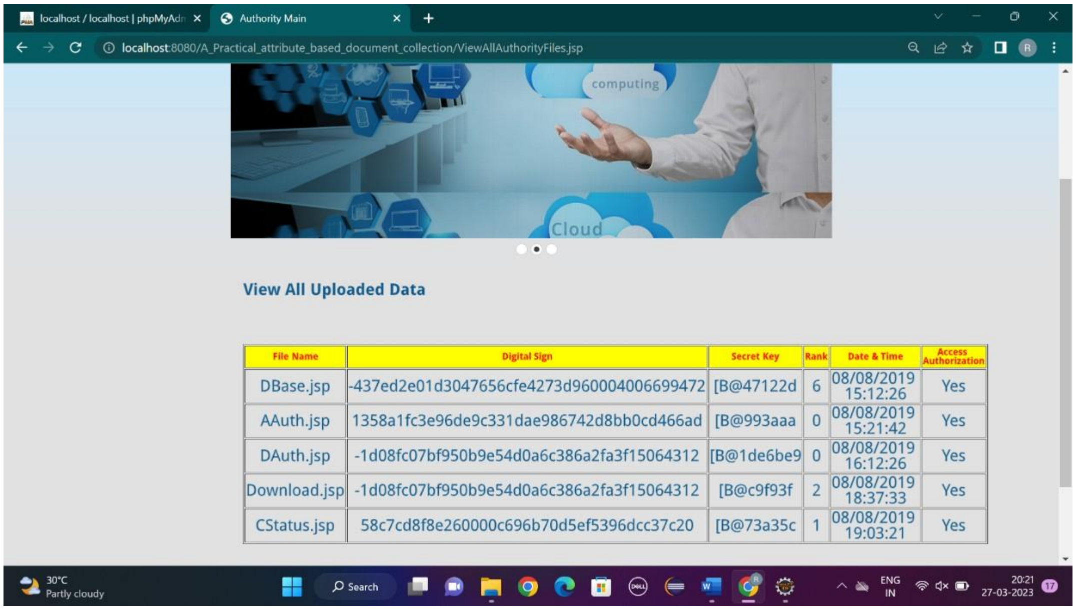
File Name	Digital Sign	Secret Key	Rank	Date & Time	Access Authorization
DBase.jsp	-437ed2e01d3047656cfe4273d960004006699472	[B@47122d	6	08/08/2019 15:12:26	Yes
AAuth.jsp	1358a1fc3e96de9c331dae986742d8bb0cd466ad	[B@993aaa	0	08/08/2019 15:21:42	Yes
DAuth.jsp	-1d08fc07bf950b9e54d0a6c386a2fa3f15064312	[B@1de6be9	0	08/08/2019 16:12:26	Yes
Download.jsp	-1d08fc07bf950b9e54d0a6c386a2fa3f15064312	[B@c9f93f	2	08/08/2019 18:37:33	Yes
CStatus.jsp	58c7cd8f8e260000c696b70d5ef5396dcc37c20	[B@73a35c	1	08/08/2019 19:03:21	Yes


localhost / localhost | phpMyAdm
Upload
localhost:8080/A_Practical_attribute_based_document_collection/Upload.jsp
Sidebar Menu
Home
Log Out
Choose File
Encrypt
30°C
Partly cloudy
Search
ENG
IN
20:20
27-03-2023

Upload Data !!!

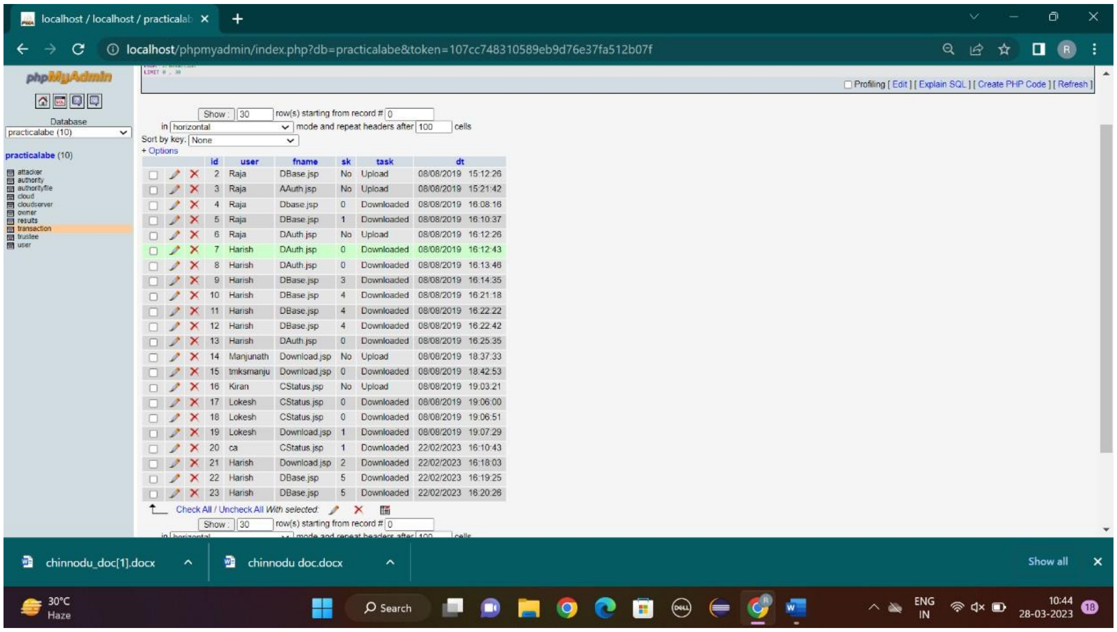
Select File :-	Choose File No file chosen
File Name :-	
Data Sign	
	Encrypt

UPLOADED DATA:


localhost / localhost | phpMyAdm
Authority Main
localhost:8080/A_Practical_attribute_based_document_collection/ViewAllAuthorityFiles.jsp
computing
Cloud
30°C
Partly cloudy
Search
ENG
IN
20:21
27-03-2023

View All Uploaded Data

File Name	Digital Sign	Secret Key	Rank	Date & Time	Access Authorization
DBase.jsp	-437ed2e01d3047656cfe4273d960004006699472	[B@47122d	6	08/08/2019 15:12:26	Yes
AAuth.jsp	1358a1fc3e96de9c331dae986742d8bb0cd466ad	[B@993aaa	0	08/08/2019 15:21:42	Yes
DAuth.jsp	-1d08fc07bf950b9e54d0a6c386a2fa3f15064312	[B@1de6be9	0	08/08/2019 16:12:26	Yes
Download.jsp	-1d08fc07bf950b9e54d0a6c386a2fa3f15064312	[B@c9f93f	2	08/08/2019 18:37:33	Yes
CStatus.jsp	58c7cd8f8e260000c696b70d5ef5396dcc37c20	[B@73a35c	1	08/08/2019 19:03:21	Yes



Show : 30 row(s) starting from record # 0

in horizontal mode and repeat headers after 100 cells

Sort by key: None

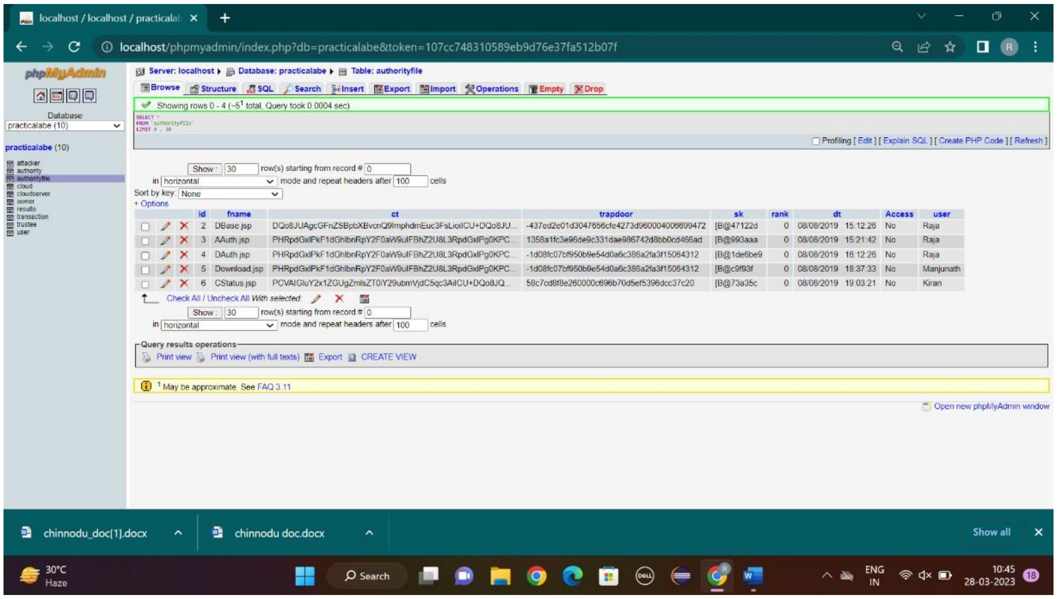
+ Options

			id	user	fname	sk	task	dt
□	✎	✗	2	Raja	DBase.jsp	No	Upload	08/08/2019 15:12:26
□	✎	✗	3	Raja	AAuth.jsp	No	Upload	08/08/2019 15:21:42
□	✎	✗	4	Raja	Dbase.jsp	0	Downloaded	08/08/2019 16:08:16
□	✎	✗	5	Raja	DBase.jsp	1	Downloaded	08/08/2019 16:10:37
□	✎	✗	6	Raja	DAuth.jsp	No	Upload	08/08/2019 16:12:26
□	✎	✗	7	Harish	DAuth.jsp	0	Downloaded	08/08/2019 16:12:43
□	✎	✗	8	Harish	DAuth.jsp	0	Downloaded	08/08/2019 16:13:46
□	✎	✗	9	Harish	DBase.jsp	3	Downloaded	08/08/2019 16:14:35
□	✎	✗	10	Harish	DBase.jsp	4	Downloaded	08/08/2019 16:21:18
□	✎	✗	11	Harish	DBase.jsp	4	Downloaded	08/08/2019 16:22:22
□	✎	✗	12	Harish	DBase.jsp	4	Downloaded	08/08/2019 16:22:42
□	✎	✗	13	Harish	DAuth.jsp	0	Downloaded	08/08/2019 16:25:35
□	✎	✗	14	Manjunath	Download.jsp	No	Upload	08/08/2019 18:37:33
□	✎	✗	15	tmksmanju	Download.jsp	0	Downloaded	08/08/2019 18:42:53
□	✎	✗	16	Kiran	CStatus.jsp	No	Upload	08/08/2019 19:03:21
□	✎	✗	17	Lokesh	CStatus.jsp	0	Downloaded	08/08/2019 19:06:00
□	✎	✗	18	Lokesh	CStatus.jsp	0	Downloaded	08/08/2019 19:06:51
□	✎	✗	19	Lokesh	Download.jsp	1	Downloaded	08/08/2019 19:07:29
□	✎	✗	20	ca	CStatus.jsp	1	Downloaded	22/02/2023 16:10:43
□	✎	✗	21	Harish	Download.jsp	2	Downloaded	22/02/2023 16:18:03
□	✎	✗	22	Harish	DBase.jsp	5	Downloaded	22/02/2023 16:19:25
□	✎	✗	23	Harish	DBase.jsp	5	Downloaded	22/02/2023 16:20:26

Check All / Uncheck All *With selected:* ✎ ✗

Show : 30 row(s) starting from record # 0

AUTHORITY FILES:



localhost / localhost / practicalab
localhost/phpmyadmin/index.php?db=practicalabe&token=107cc748310589eb9d76e37fa512b07f
Server: localhost ▸ Database: practicalabe ▸ Table: authorityfile
Browse
Structure
SQL
Search
Insert
Export
Import
Operations
Empty
Drop

Showing rows 0 - 4 (~5[1] total, Query took 0.0004 sec)

```
SELECT *
FROM `authorityfile`
LIMIT 0 , 30
```

□ Profiling [Edit] [Explain SQL] [Create PHP Code] [Refresh]

Show : 30 row(s) starting from record # 0
in horizontal mode and repeat headers after 100 cells
Sort by key: None
+ Options

id	fname	ct	trapdoor	sk	rank	dt	Access	user
2	DBase.jsp	DQo8JUAgcGFnZSBpbXBvcnQ9ImphdmEuc3FsLioiICU+DQo8JU...	-437ed2e01d3047656cfe4273d960004006699472	[B@47122d	0	08/08/2019 15:12:26	No	Raja
3	AAuth.jsp	PHRpdGxlPkF1dGhlbnRpY2F0aW9uIFBhZ2U8L3RpdGxlPg0KPC...	1358a1fc3e96de9c331dae986742d8bb0cd466ad	[B@993aaa	0	08/08/2019 15:21:42	No	Raja
4	DAuth.jsp	PHRpdGxlPkF1dGhlbnRpY2F0aW9uIFBhZ2U8L3RpdGxlPg0KPC...	-1d08fc07bf950b9e54d0a6c386a2fa3f15064312	[B@1de6be9	0	08/08/2019 16:12:26	No	Raja
5	Download.jsp	PHRpdGxlPkF1dGhlbnRpY2F0aW9uIFBhZ2U8L3RpdGxlPg0KPC...	-1d08fc07bf950b9e54d0a6c386a2fa3f15064312	[B@c9f93f	0	08/08/2019 18:37:33	No	Manjunath
6	CStatus.jsp	PCVAIGluY2x1ZGUgZmlsZT0iY29ubmVjdC5qc3AiICU+DQo8JQ...	58c7cd8f8e260000c696b70d5ef5396dcc37c20	[B@73a35c	0	08/08/2019 19:03:21	No	Kiran

Check All / Uncheck All *With selected:*

Show : 30 row(s) starting from record # 0
in horizontal mode and repeat headers after 100 cells

Query results operations
Print view Print view (with full texts) Export CREATE VIEW

[1] May be approximate. See FAQ 3.11

Open new phpMyAdmin window

6. CONCLUSION

In this paper, we design a hierarchical document collection encryption scheme. We must design an incremental algorithm to construct the integrated access trees of the documents and decrease the number of trees. Then, each integrated access tree is encrypted together and the documents in a tree can be decrypted at a time. Different from existing schemes, we construct the secret numbers for the nodes of the trees in a bottom-up manner. In this way, the sizes of cipher text and secret keys significantly decrease. At last, a thorough performance evaluation is provided including security analysis, efficiency analysis, and simulation. Results show that the proposed scheme outperforms KP-ABE and CP-ABE schemes in terms of encryption/decryption

efficiency and storage space. Our scheme can be further improved in several aspects: First, the access policy discussed in Section III assumes that the access trees are composed of only ``AND'' gates. Extending the flexibility and versatility of the access policy is one of the most important research directions. Second, the documents are encrypted before outsourcing and a promising task is how to efficiently search the interested documents over the cipher texts. At last, we focus our attention on a static document collection, and how to efficiently encrypt/decrypt a dynamic document collection will be also researched in the future.

REFERENCES

1. Li, W. Yao, Y. Zhang, H. Qian, and J. Han, ``Flexible and _ne-grained attribute-based data storage in cloud computing,'' IEEE Trans. Services Comput., vol. 10, no. 5, pp. 785_796, Sep./Oct. 2017.

2. Z. Fu, K. Ren, J. Shu, X. Sun, and F. Huang, ``Enabling personalized search over encrypted outsourced data with efficiency improvement," IEEE Trans. Parallel Diatribe. Syst., vol. 27, no. 9, pp. 2546_2559, Sep. 2016. C. Chen et al., ``An ef_cient privacy-preserving ranked keyword search method," IEEE Trans.

3. Parallel Distrib. Syst., vol. 27, no. 4, pp. 951_963, Apr. 2016.

4. Z. Liu, Z. Cao, and D. S. Wong, ``Traceable CP-ABE: How to trace

5. decryption devices found in the wild," IEEE Trans. Inf. Forensics Security, vol. 10, no. 1, pp. 55_68, Jan. 2015.

6. N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, ``Privacy-preserving multi-keyword ranked search over encrypted cloud data," IEEE Trans. Parallel Distrib. Syst., vol. 25, no. 1, pp. 222_233, Jan. 2014.

7. H. Deng et al., ``Ciphertext-policy hierarchical attribute-based encryption with short ciphertexts," Inf. Sci., vol. 275, pp. 370_384, Aug. 2014. W. Liu, J. Liu, Q. Wu, B. Qin, and Y. Zhou, ``Practical direct chosen

8. ciphertext secure key-policy attribute-based encryption with public cipher-text test," in Proc. Eur. Symp. Res. Comput. Secure. Cham, Switzerland: Springer, 2014, pp. 91_108. □ J. Ning, Z. Cao, X. Dong, L. Wei, and X. Lin, ``Large universe ciphertext-policy attribute-based encryption with white-box traceability," in Proc.

9. Eur. Symp. Res. Comput. Secur., vol. 8713, 2014, pp. 55_72.