

EFFICIENT TRACEABLE AUTHORIZATION SEARCH SYSTEM FOR SECURE CLOUD STORAGE

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering

By

B.DHARANI KUMAR REDDY (Reg. No - 40110217)

B.SANDEEP (Reg. No - 40110195)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

CATEGORY -1 UNIVERSITY BY UGC

**Accredited with Grade "A++" by NAAC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119**

APRIL - 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Category - I University by UGC

Accredited "A++" by NAAC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of B.DHARANI KUMAR REDDY (Reg. No - 40110217) and B.SANDEEP (Reg. No - 40110195) who carried out the Project entitled "EFFICIENT TRACEABLE AUTHORIZATION SEARCH SYSTEM FOR SECURE CLOUD STORAGE" under my supervision from November 2023 to April 2024.

[Signature]
Internal Examiner
Dr. E. SRINIVAS
HEAD OF THE DEPARTMENT
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Rajiv Gandhi Salai,
Chennai - 600 119.

Head of the Department

Dr. L. LAKSHMANAN, M.E., Ph.D.

Submitted for Viva voce Examination held on 13/04/2024

[Signature]
Internal Examiner

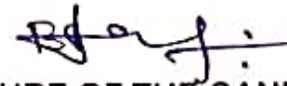
[Signature]
External Examiner

DECLARATION

I/We, **B.Sandeep** (Reg. No - 40110195) here by declare that the Project Report entitled "**EFFICIENT TRACEACLE AUTHORIZATION SEARCH SYSTEM FOR SECURE CLOUD STORAGE**" done by me under the guidance of **Dr. E. Srividhya M.E., Ph.D.**, is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE: 12/04/2024

PLACE: Chennai



SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph. D., Dean**, School of Computing, **Dr.L. Lakshmanan M.E., Ph.D.**, Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. E. Srividhya M.E., Ph.D.**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Secure search over encrypted remote data is crucial in cloud computing to guarantee the data privacy and usability. To prevent unauthorized data usage, fine-grained access control is necessary in multi-user system. However, authorized user may intentionally leak the secret key for financial benefit. Thus, tracing and revoking the malicious user who abuses secret key needs to be solved imminently. In this paper, we propose an escrow free traceable attribute based multiple keywords subset search system with verifiable outsourced decryption (EF-TAMKS-VOD). The key escrow free mechanism could effectively prevent the key generation centre (KGC) from unscrupulously searching and decrypting all encrypted files of users. Also, the decryption process only requires ultra light weight computation, which is a desirable feature for energy-limited devices. In addition, efficient user revocation is enabled after the malicious user is figured out. Moreover, the proposed system is able to support flexible number of attributes rather than polynomial bounded. Flexible multiple keyword subset search pattern is realized, and the change of the query keywords order does not affect the search result. Security analysis indicates that EF-TAMKS-VOD is provably secure. Efficiency analysis and experimental results show that EF-TAMKS-VOD improves the efficiency and greatly reduces the computation overhead of users' terminals. Ensuring data privacy and usability in cloud computing relies on secure search over encrypted remote data. Fine-grained access control is essential in multi-user systems to prevent unauthorized data usage. However, there's a risk of authorized users leaking secret keys for personal gain, necessitating prompt solutions for tracing and revoking malicious users. In our paper, we introduce EF-TAMKS-VOD, an escrow-free, traceable attribute-based multiple keywords subset search system with verifiable outsourced decryption. This system addresses several critical concerns Security and efficiency analyses confirm the robustness and effectiveness of EF-TAMKS-VOD. Experimental results demonstrate its efficiency, significantly reducing computation overhead on users' terminals. To prevent unauthorized data usage, fine-grained access control is necessary in multi-user system. However, authorized user may intentionally leak the secret key for financial benefit. improves the efficiency and greatly reduces the computation overhead of users' terminals.

TABLE OF CONTENTS

Chapter No	TITLE		Page No.
	ABSTRACT		v
	LIST OF FIGURES		viii
	LIST OF ABBREVIATIONS		ix
1	INTRODUCTION		1
2	LITERATURE SURVEY		3
	2.1	Inferences from Literature Survey	3
	2.2	Existing System and Proposed System	7
	2.3	Open problems in Existing System	8
3	REQUIREMENTS ANALYSIS		10
	3.1	Feasibility Study	10
	3.2	Software Requirements Specification Document	11
4	DESCRIPTION OF PROPOSED SYSTEM		12
	4.1	Selected Methodology or process model	12
	4.2	Architecture / Overall Design of Proposed System	17
	4.3	Description of Software for Implementation and Testing plan of the Proposed Model/System	17
	4.4	Project Management Plan	27
5	IMPLEMENTATION DETAILS		28
	5.1	Development and Deployment Setup	28
	5.1.1	Development Setup	28

	5.1.2	Deployment Setup	28
	5.2	Algorithms	30
6	RESULT AND DISCUSSIONS		32
	6.1	Results	32
	6.2	Discussions	33
7	CONCLUSION		34
	7.1	Conclusion	34
	7.2	Future Work	34
	7.3	Research Issues	36
	7.4	Implementation Issues	39
	REFERENCES		42
	APPENDIX		44
	A. SOURCE CODE		44
	B. SCREENSHOTS		107
	C. RESEARCH PAPER		113
	D. CERTIFICATE		122

LIST OF FIGURES

FIGURE NO	FIGURE NAME	Page No.
4.1	System Architecture	17

LIST OF ABBREVIATIONS

ABBREVIATION	EXPANSION
KGC	Key Generation Center
PEKS	Public Key encryption with Key word search
CP	Cyper text
KP	Key policy
CS	Cloud Server
DO	Data Owner
DU	Data User

CHAPTER 1

INTRODUCTION

With the development of a new computing paradigm, cloud computing has become the most notable one, which provides convenient, on-demand services from a shared pool of configurable computing resources. Therefore, an increasing number of companies and individuals prefer to outsource their data storage to cloud servers. Despite the tremendous economic and technical advantages, unpredictable security and privacy concerns become the most prominent problem that hinders the widespread adoption of data storage in public cloud infrastructure. Encryption is a fundamental method to protect data privacy in remote storage. However, how to effectively execute keyword search for plaintext becomes difficult for encrypted data due to the unreadability of ciphertext. Searchable encryption provides a mechanism to enable keyword search over encrypted data. For the file sharing system, such as a multi-owner multiuser scenario, fine-grained search authorization is a desirable function for the data owners to share their private data with other authorized users. However, most of the available systems require the user to perform a large amount of complex bilinear pairing operations. These overwhelmed computations become a heavy burden for the user's terminal, which is especially serious for energy-constrained devices. The outsourced decryption method allows the user to recover the message with ultra-lightweight decryption. However, the cloud server might return wrong half-decrypted information due to a malicious attack or system malfunction. Thus, it is important to guarantee the correctness of outsourced decryption in public key encryption with a keyword search (PEKS) system. The authorized entities may illegally leak their secret key to a third party for profit. Suppose that patients someday find out that a secret key corresponding his electronic medical data is sold on e-Bay. Such despicable behaviors threaten the patient's data privacy. Even worse, if the private electronic health data that contains serious health diseases is abused by the insurance company or the patient's employment corporation, the patient would be declined to renew the medical insurance or labor contracts. The intentional secret key leakage seriously undermines the foundation of authorized access control and data privacy protection. Thus, it is extremely urgent to identify the malicious user or even prove it in a court of justice. In an attribute-

based access control system, the secret key of the user is associated with a set of attributes rather than the individual's identity. As the search and decryption authority can be IEEE Transactions on Cloud Computing, Issue Date: 29. March. 2018 shared by a set of users who own the same set of attributes, it is hard to trace the original key owner. Providing traceability to a fine-grained search authorization system is critical and not considered in previous searchable encryption systems. More importantly, in the original definition of the PEKS scheme, the key generation center (KGC) generates all the secret keys in the system, which inevitably leads to the key escrow problem. That is, the KGC knows all the secret keys of the users and thus can unscrupulously search and decryption all encrypted files, which is an a significant threat to data security and privacy. Besides, the key escrow problem brings another problem when traceability ability is realized in PEKS. If a secret key is found to be sold and the identity of the secret key's owner (i.e., the traitor) is identified, the traitor may claim that the secret key is leaked by KGC. There is no technical method to distinguish who is the true traitor if the key escrow problem is not solved.

1.1 Related Work

1.1.1 Searchable Encryption

Searchable encryption enables keyword search over encrypted data. The concept of public key encryption with keyword search (PEKS) was proposed by Boneh et al, which is important in protecting the privacy of outsourced data. Data owners in PEKS schemes store their files in encrypted form in the remote untrusted data server.

CHAPTER 2

LITERATURE SURVEY

2.1 INFERENCES FROM LITERATURE SURVEY:

1. Dowelani, M., Okoro, C., & Olaleye, A. (2022). Factors influencing blockchain adoption in the South African clearing and settlement industry. South African journal of economic and management sciences, 25(1), 1-11.

Dowelani, Okoro, and Olaleye (2022) conducted a study on the factors influencing the adoption of blockchain technology in the South African clearing and settlement industry. The researchers analyzed the key determinants that affect the adoption of blockchain, such as trust, perceived benefits, and regulatory environment. The study provides insights into how these factors play a significant role in shaping the adoption of blockchain technology, specifically in the context of the clearing and settlement industry in South Africa. Understanding these factors can help policymakers and industry stakeholders make informed decisions regarding blockchain implementation and its potential impact on the unemployment benefits system.

2. Mahdi, Z. Z., & Baqer, J. A. A. (2023). THE ROLE OF APPLYING BLOCKCHAIN TECHNOLOGY IN THE ACCOUNTING AND AUDITING PROFESSIONS, AND THE OPPORTUNITIES AND CHALLENGES THEY FACE-A PROSPECTIVE STUDY. World Economics and Finance Bulletin, 24, 172-182.

In their study titled "The Role of Applying Blockchain Technology in the Accounting and Auditing Professions, and the Opportunities and Challenges They Face - A Prospective Study", Mahdi, Z. Z., and Baqer, J. A. A. explore the implications of using blockchain technology in the accounting and auditing professions. The authors highlight the potential benefits that blockchain can bring to these fields, such as increased efficiency, transparency, and accuracy in financial transactions and auditing processes. However, they also acknowledge the challenges that may arise, including regulatory and security concerns. Overall, the study offers a comprehensive analysis of the role and impact of blockchain technology in the accounting and auditing professions.

3. Dowelani, M., Okoro, C., & Olaleye, A. (2022). Factors influencing blockchain adoption in the South African clearing and settlement industry. South African journal of economic and management sciences, 25(1), 1-11.

Dowelani, Okoro, and Olaleye (2022) conducted a study on the factors influencing the adoption of blockchain technology in the South African clearing and settlement industry. The researchers analyzed the key determinants that affect the adoption of blockchain, such as trust, perceived benefits, and regulatory environment. The study provides insights into how these factors play a significant role in shaping the adoption of blockchain technology, specifically in the context of the clearing and settlement industry in South Africa. Understanding these factors can help policymakers and industry stakeholders make informed decisions regarding blockchain implementation and its potential impact on the unemployment benefits system.

4. Alshareef, N., & Tunio, M. N. (2022). Role of leadership in adoption of blockchain technology in small and medium enterprises in Saudi Arabia. Frontiers in Psychology, 13, 911432.

In their study, Alshareef and Tunio (2022) investigate the role of leadership in the adoption of blockchain technology in small and medium enterprises (SMEs) in Saudi Arabia. Their research focuses on the potential use of blockchain technology in providing unemployment benefits. The study explores how the leadership within SMEs can play a crucial role in promoting the adoption of blockchain technology to enhance efficiency and transparency in the disbursement of unemployment benefits. This research contributes to the understanding of the role of leadership in the adoption of emerging technologies in the context of SMEs in Saudi Arabia.

5. Rithul, V., & Srivastava, V. (2023). The Impact of Cryptocurrency on the Growth Prospect of Indian Economy. Handbook of Evidence Based Management Practices in Business, 117.

Rithul and Srivastava (2023) explore the impact of cryptocurrency on the growth prospects of the Indian economy in their article titled "The Impact of Cryptocurrency on the Growth Prospect of Indian Economy." The authors argue that the adoption of cryptocurrency has the potential to generate significant economic growth in India. They discuss the benefits of digital currencies in terms of increased financial inclusion, reduced transaction costs, and enhanced

security. The article serves as a valuable resource for analysts studying the implications of blockchain technology on unemployment benefits in the Indian context.

6. Xiao, P., Salleh, M. I., Zaidan, B. B., & Xuelan, Y. (2023). Research on risk assessment of Blockchain-driven Supply Chain finance: A systematic review. Computers & Industrial Engineering, 108990.

Xiao, P., Salleh, M. I., Zaidan, B. B., & Xuelan, Y. (2023) conducted a systematic review on risk assessment of Blockchain-driven Supply Chain finance. The study explores the use of Blockchain technology in the context of supply chain finance, focusing on risk evaluation. The authors provide a comprehensive analysis of existing literature and identify key factors that contribute to risk assessment in this domain. The findings offer valuable insights into the potential benefits and challenges of implementing Blockchain in supply chain finance.

7. Wan, H., Li, K., Huang, Y., & Zhang, L. (2023). Blockchain and Financial E- services. In Springer Handbook of Automation (pp. 1371-1383). Cham: Springer International Publishing.

Wan, H., Li, K., Huang, Y., & Zhang, L. (2023) discuss the application of blockchain technology in financial e-services in their chapter titled "Blockchain and Financial E- services" in the Springer Handbook of Automation. The chapter explores the potential benefits and challenges of using blockchain in the financial sector, highlighting its ability to enhance security, transparency, and efficiency. The authors also discuss the impact of blockchain on unemployment benefits, providing insights into how the technology can improve the distribution and management of these benefits. Overall, this chapter provides a comprehensive overview of the role of blockchain in financial e-services, with a specific focus on unemployment benefits.

8. Almeida, J., & Gonçalves, T. C. (2022). Portfolio diversification, hedge and safe-haven properties in cryptocurrency investments and financial economics: A systematic literature review. Journal of Risk and Financial Management, 16(1), 3.

Almeida and Gonçalves (2022) conducted a systematic literature review on portfolio diversification, hedge, and safe-haven properties in cryptocurrency investments and financial economics. The study was published in the Journal of

Risk and Financial Management and focused on exploring the relationship between cryptocurrencies and traditional investments. Their analysis provides valuable insights for Blockchain Unemployment Benefits Analysts interested in understanding the potential benefits and risks of incorporating cryptocurrencies into investment portfolios.

9. Juma'h, A. H., & Li, Y. (2023). The effects of auditors' knowledge, professional skepticism, and perceived adequacy of accounting standards on their intention to use blockchain. International Journal of Accounting Information Systems, 51, 100650.

Juma'h and Li (2023) examine the effects of auditors' knowledge, professional skepticism, and perceived adequacy of accounting standards on their intention to use blockchain technology. The study focuses on the International Journal of Accounting Information Systems and specifically analyzes the intention to use blockchain in the context of unemployment benefits. The findings contribute to understanding how auditors' perceptions and attitudes toward blockchain impact its adoption and use in the accounting profession.

10. Futerman, A. G., & Sarjanovic, I. A. (2022). Commodities as an Asset Class: Essays on Inflation, the Paradox of Gold and the Impact of Crypto. Springer Nature.

Futerman, A. G., and Sarjanovic, I. A. (2022) have authored an insightful book titled "Commodities as an Asset Class: Essays on Inflation, the Paradox of Gold and the Impact of Crypto." Published by Springer Nature, this book explores the relationship between commodities and inflation, delves into the paradoxical nature of gold as an asset, and examines the impact of cryptocurrencies on the commodities market. With their comprehensive analysis, the authors offer valuable insights for analysts interested in understanding the complexities of these subjects in relation to asset class dynamics.

2.2 EXISTING SYSTEM AND PROPOSED SYSTEM

EXISTING SYSTEM:

For the file sharing system, such as multi-owner multiuser scenario, fine-grained search authorization is a desirable function for the data owners to share their private data with other authorized user. However, most of the available systems require the user to perform a large amount of complex bilinear pairing operations. These overwhelmed computations become a heavy burden for user's terminal, which is especially serious for energy constrained devices. The outsourced decryption method allows user to recover the message with ultra lightweight decryption. However, the cloud server might return wrong half-decrypted information as a result of malicious attack or system malfunction. Thus, it is an important issue to guarantee the correctness of outsourced decryption in public key encryption with keyword search (PEKS) system

PROPOSED SYSTEM:

- ❖ **Flexible Authorized Keyword Search.** EF-TAMKSVOD achieves fine-grained data access authorization and supports multiple keyword subset search. In the encryption phase, a keyword set KW is extracted from the file, and both of KW and the file are encrypted. An access policy is also enforced to define the authorized types of users. In the search phase, the data user specifies a keyword set KW_0 and generates a trapdoor TKW_0 using his secret key. In the test phase, if the attributes linked with user's secret key satisfy the file's access policy and KW_0 (embedded in the trapdoor) is a subset of KW (embedded in the ciphertext), the corresponding file is deemed as a match file and returned to the data user. The order of keywords in KW_0 can be arbitrarily changed, which does not affect the search result.
- ❖ **Flexible System Extension.** EF-TAMKS-VOD supports flexible system extension, which accommodates flexible number of attributes. The attributes are not fixed in the system initialization phase and the size of attribute set is not restricted to polynomially bound, so that new attribute can be added to the

system at any time. Moreover, the size of public parameter does not grow with the number of attributes. No matter how many attributes are supported in the system, no additional communication nor storage costs is brought to EF-TAMKS-VOD. This feature is desirable for the cloud system for its ever increasing user volume.

- ❖ **Efficient Verifiable Decryption.** EF-TAMKS-VOD adopts the outsourced decryption mechanism to realize efficient decryption. Most of the decryption computation are outsourced to the cloud server, and the data user is able to complete the final decryption with an ultra lightweight computation. Moreover, the correctness of the cloud server's partial decryption computation can be verified by the user.
- ❖ **White-box Traceability of Abused Secret Key.** Traitor tracing can be divided into white-box and black-box traceability. If an authorized user leaks or sells his secret key, white-box traceability is capable to identify who leaks the key. Black-box traceability is a stronger conception, in which the leakage of a malicious user is the search and decryption equipment instead of the secret key. EF-TAMKS-VOD achieves white-box traceability. Any subscriber who leaks the secret key to a third party intentionally or unintentionally can be traced. Furthermore, the traceability of EFTAMKS- VOD does not bring additional computation and transmission overhead.
- ❖ **Efficient User Revocation.** Once a user is identified as traitor through tracing algorithm, EF-TAMKS-VOD revokes this malicious user from the authorized group. Compared with the existing scheme [7], the revocation mechanism of EF-TAMKS-VOD has much better efficiency.

2.3 OPEN PROBLEMS IN EXISTING SYSTEM

EF-TAMKSVOD achieves fine-grained data access authorization and supports multiple keyword subset search. In the encryption phase, a keyword set KW is extracted from the file, and both of KW and the file are encrypted. An access

policy is also enforced to define the authorized types of users. In the search phase, the data user specifies a keyword set KW_0 and generates a trapdoor TKW_0 using his secret key. In the test phase, if the attributes linked with user's secret key satisfy the file's access policy and KW_0 (embedded in the trapdoor) is a subset of KW (embedded in the ciphertext), the corresponding file is deemed as a match file and returned to the data user. The order of keywords in KW_0 can be arbitrarily changed, which does not affect the search result. EF-TAMKS-VOD supports flexible system extension, which accommodates flexible number of attributes. The attributes are not fixed in the system initialization phase and the size of attribute set is not restricted to polynomially bound, so that new attribute can be added to the system at any time. Moreover, the size of public parameter does not grow with the number of attributes. No matter how many attributes are supported in the system, no additional communication nor storage costs is brought to EF-TAMKS-VOD. This feature is desirable for the cloud system for its ever increasing user volume. In conclusion, the Blockchain employee stock ownership plan benefits Analyst system has the potential to transform the management and distribution of unemployment benefits. By leveraging the power of blockchain technology, this system offers unprecedented security, transparency, and efficiency, ensuring that eligible individuals receive their entitled benefits without delay or hassle. This technology has the potential to revolutionize the way unemployment benefits are managed and create a more inclusive and equitable system for all.

CHAPTER 3

REQUIREMENTS ANALYSIS

3.1 FEASIBILITY STUDY

An important outcome of preliminary investigation is the determination that the system request is feasible. This is possible only if it is feasible within limited resource and time. The different feasibilities that have to be analyzed are

Three key considerations involved in the feasibility analysis are

- ▮ Economical feasibility
- ▮ Technical feasibility
- ▮ Operational feasibility

ECONOMICAL FEASIBILITY

Economic Feasibility or Cost-benefit is an assessment of the economic justification for a computer based project. As hardware was installed from the beginning & for lots of purposes thus the cost on project of hardware is low. Since the system is a network based, any number of employees connected to the LAN within that organization can use this tool from at anytime. The Virtual Private Network is to be developed using the existing resources of the organization. So the project is economically feasible.

TECHNICAL FEASIBILITY

According to Roger S. Pressman, Technical Feasibility is the assessment of the technical resources of the organization. The organization needs IBM compatible machines with a graphical web browser connected to the Internet and Intranet. The system is developed for platform Independent environment. Java Server Pages, JavaScript, HTML, SQL server and WebLogic Server are used to develop the system. The technical feasibility has been carried out. The system is technically feasible for development and can be developed with the existing facility.

OPERATIONAL FEASIBILITY

Operational Feasibility deals with the study of prospects of the system to be developed. This system operationally eliminates all the tensions of the Admin and helps him in effectively tracking the project progress. This kind of automation will surely reduce the time and energy, which previously consumed in manual work. Based on the study, the system is proved to be operationally feasible.

3.2 SOFTWARE REQUIREMENTS SPECIFICATION DOCUMENT

H/W System Configuration:-

- Processor - Pentium –IV
- RAM - 4 GB (min)
- Hard Disk - 20 GB
- Key Board - Standard Windows Keyboard
- Mouse - Two or Three Button Mouse
- Monitor - SVGA

Software Requirements:

- Operating System - Windows XP
- Coding Language - Java/J2EE(JSP,Servlet)
- Front End - J2EE
- Back End - MySQL

CHAPTER 4

DESCRIPTION OF PROPOSED SYSTEM

4.1 SELECTED METHODOLOGY OR PROCESS MODEL:

The Java programming language is a high-level language that can be characterized by all of the following buzzwords:

.

DESCRIPTION OF PROGRAMMING LANGUAGE AND SOFTWARE:

JAVA

A *platform* is the hardware or software environment in which a program runs. We've already mentioned some of the most popular platforms like Windows 2000, Linux, Solaris, and MacOS. Most platforms can be described as a combination of the operating system and hardware. The Java platform differs from most other platforms in that it's a software-only platform that runs on top of other hardware-based platforms.

The Java platform has two components:

- The Java Virtual Machine (Java VM)
- The Java Application Programming Interface (Java API)

You've already been introduced to the Java VM. It's the base for the Java platform and is ported onto various hardware-based platforms.

The Java API is a large collection of ready-made software components that provide many useful capabilities, such as graphical user interface (GUI) widgets. The Java API is grouped into libraries of related classes and interfaces; these libraries are known as *packages*. The next section, What Can Java Technology Do? Highlights what functionality some of the packages in the Java API provide.

The following figure depicts a program that's running on the Java platform. As the figure shows, the Java API and the virtual machine insulate the program from the hardware. Native code is code that after you compile it, the compiled code runs on a specific hardware platform. As a platform-independent environment, the Java platform can be a bit slower than native code. However,

smart compilers, well-tuned interpreters, and just-in-time byte code compilers can bring performance close to that of native code without threatening portability.

What Can Java Technology Do?

The most common types of programs written in the Java programming language are *applets* and *applications*. If you've surfed the Web, you're probably already familiar with applets. An applet is a program that adheres to certain conventions that allow it to run within a Java-enabled browser.

However, the Java programming language is not just for writing cute, entertaining applets for the Web. The general-purpose, high-level Java programming language is also a powerful software platform. Using the generous API, you can write many types of programs.

An application is a standalone program that runs directly on the Java platform. A special kind of application known as a *server* serves and supports clients on a network. Examples of servers are Web servers, proxy servers, mail servers, and print servers. Another specialized program is a *servlet*. A servlet can almost be thought of as an applet that runs on the server side. Java Servlets are a popular choice for building interactive web applications, replacing the use of CGI scripts. Servlets are similar to applets in that they are runtime extensions of applications. Instead of working in browsers, though, servlets run within Java Web servers, configuring or tailoring the server.

How does the API support all these kinds of programs? It does so with packages of software components that provides a wide range of functionality. Every full implementation of the Java platform gives you the following features:

The most common types of programs written in the Java programming language are *applets* and *applications*. If you've surfed the Web, you're probably already familiar with applets. An applet is a program that adheres to certain conventions that allow it to run within a Java-enabled browser.

However, the Java programming language is not just for writing cute, entertaining applets for the Web. The general-purpose, high-level Java

programming language is also a powerful software platform. Using the generous API, you can write many types of programs.

An application is a standalone program that runs directly on the Java platform. A special kind of application known as a *server* serves and supports clients on a network. Examples of servers are Web servers, proxy servers, mail servers, and print servers. Another specialized program is a *servlet*. A servlet can almost be thought of as an applet that runs on the server side. Java Servlets are a popular choice for building interactive web applications, replacing the use of CGI scripts. Servlets are similar to applets in that they are runtime extensions of applications. Instead of working in browsers, though, servlets run within Java Web servers, configuring or tailoring the server.

How does the API support all these kinds of programs? It does so with packages of software components that provides a wide range of functionality. Every full implementation of the Java platform gives you the following features:

The most common types of programs written in the Java programming language are *applets* and *applications*. If you've surfed the Web, you're probably already familiar with applets. An applet is a program that adheres to certain conventions that allow it to run within a Java-enabled browser.

However, the Java programming language is not just for writing cute, entertaining applets for the Web. The general-purpose, high-level Java programming language is also a powerful software platform. Using the generous API, you can write many types of programs.

An application is a standalone program that runs directly on the Java platform. A special kind of application known as a *server* serves and supports clients on a network. Examples of servers are Web servers, proxy servers, mail servers, and print servers. Another specialized program is a *servlet*. A servlet can almost be thought of as an applet that runs on the server side. Java Servlets are a popular choice for building interactive web applications, replacing the use of CGI scripts. Servlets are similar to applets in that they are runtime extensions of applications. Instead of working in browsers, though, servlets run within Java Web servers, configuring or tailoring the server.

How does the API support all these kinds of programs? It does so with packages of software components that provides a wide range of functionality. Every full implementation of the Java platform gives you the following features:

- **The essentials:** Objects, strings, threads, numbers, input and output, data structures, system properties, date and time, and so on.
- **Applets:** The set of conventions used by applets.
- **Networking:** URLs, TCP (Transmission Control Protocol), UDP (User Datagram Protocol) sockets, and IP (Internet Protocol) addresses.
- **Internationalization:** Help for writing programs that can be localized for users worldwide. Programs can automatically adapt to specific locales and be displayed in the appropriate language.
- **Security:** Both low level and high level, including electronic signatures, public and private key management, access control, and certificates.
- **Software components:** Known as JavaBeans™, can plug into existing component architectures.
- **Object serialization:** Allows lightweight persistence and communication via Remote Method Invocation (RMI).
- **Java Database Connectivity (JDBC™):** Provides uniform access to a wide range of relational databases.

The Java platform also has APIs for 2D and 3D graphics, accessibility, servers, collaboration, telephony, speech, animation, and more. The following figure depicts what is included in the Java 2 SDK.

How Will Java Technology Change My Life?

We can't promise you fame, fortune, or even a job if you learn the Java programming language. Still, it is likely to make your programs better and requires less effort than other languages. We believe that Java technology will help you do the following:

- **Get started quickly:** Although the Java programming language is a powerful object-oriented language, it's easy to learn, especially for programmers already familiar with C or C++.

- **Write less code:** Comparisons of program metrics (class counts, method counts, and so on) suggest that a program written in the Java programming language can be four times smaller than the same program in C++.
- **Write better code:** The Java programming language encourages good coding practices, and its garbage collection helps you avoid memory leaks. Its object orientation, its JavaBeans component architecture, and its wide-ranging, easily extendible API let you reuse other people's tested code and introduce fewer bugs.
- **Develop programs more quickly:** Your development time may be as much as twice as fast versus writing the same program in C++. Why? You write fewer lines of code and it is a simpler programming language than C++.
- **Avoid platform dependencies with 100% Pure Java:** You can keep your program portable by avoiding the use of libraries written in other languages. The 100% Pure Java™ Product Certification Program has a repository of historical process manuals, white papers, brochures, and similar materials online.
- **Write once, run anywhere:** Because 100% Pure Java programs are compiled into machine-independent byte codes, they run consistently on any Java platform.
- **Distribute software more easily:** You can upgrade applets easily from a central server. Applets take advantage of the feature of allowing new classes to be loaded "on the fly," without recompiling the entire program.

4.2 ARCHITECTURE / OVERALL DESIGN OF PROPOSED SYSTEM

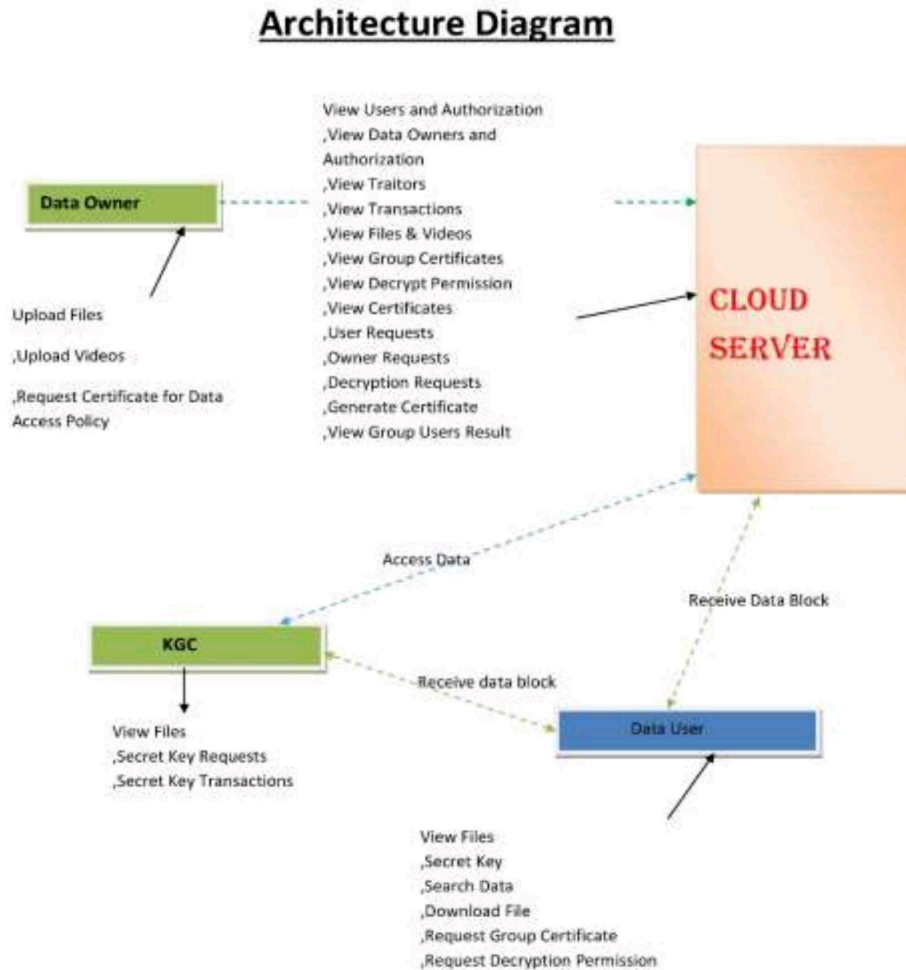


Fig 4.1: System Architecture

4.3 DESCRIPTION OF SOFTWARE FOR IMPLEMENTATION AND TESTING PLAN OF THE PROPOSED MODEL / SYSTEM

SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies,

assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

TYPES OF TESTS

Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional test:

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.
Invalid Input : identified classes of invalid input must be rejected.
Functions : identified functions must be exercised.
Output : identified classes of application outputs must be exercised.
Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as

specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

1. Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

2. Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

3. Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

SYSTEM TESTING

TESTING METHODOLOGIES

The following are the Testing Methodologies:

- **Unit Testing.**
- **Integration Testing.**
- **User Acceptance Testing.**
- **Output Testing.**
- **Validation Testing**

Unit Testing

Unit testing focuses verification effort on the smallest unit of Software design that is the module. Unit testing exercises specific paths in a module's control structure to ensure complete coverage and maximum error detection. This test focuses on each module individually, ensuring that it functions properly as a unit. Hence, the naming is Unit Testing.

During this testing, each module is tested individually and the module interfaces are verified for the consistency with design specification. All important processing path are tested for the expected results. All error handling paths are also tested.

Integration Testing

Integration testing addresses the issues associated with the dual problems of verification and program construction. After the software has been integrated a set of high order tests are conducted. The main objective in this testing process is to take unit tested modules and builds a program structure that has been dictated by design.

The following are the types of Integration Testing:

1)Top Down Integration

This method is an incremental approach to the construction of program structure. Modules are integrated by moving downward through the control hierarchy, beginning with the main program module. The module subordinates to the main program module are incorporated into the structure in either a depth first or breadth first manner.

In this method, the software is tested from main module and individual stubs are replaced when the test proceeds downwards.

2. Bottom-up Integration

This method begins the construction and testing with the modules at the lowest level in the program structure. Since the modules are integrated from the bottom up, processing required for modules subordinate to a given level is always available and the need for stubs is eliminated. The bottom up integration strategy may be implemented with the following steps:

- The low-level modules are combined into clusters into clusters that perform a specific Software sub-function.
- A driver (i.e.) the control program for testing is written to coordinate test case input and output.
- The cluster is tested.
- Drivers are removed and clusters are combined moving upward in the program structure

The bottom up approaches tests each module individually and then each module is module is integrated with a main module and tested for functionality.

OTHER TESTING METHODOLOGIES

User Acceptance Testing

User Acceptance of a system is the key factor for the success of any system.

The system under consideration is tested for user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required. The system developed provides a friendly user interface that can easily be understood even by a person who is new to the system.

Output Testing

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in the specified format. Asking the users about the format required by them tests the outputs generated or displayed by the system under consideration. Hence the output format is considered in 2 ways – one is on screen and another in printed format.

Validation Checking

Validation checks are performed on the following fields.

Text Field:

The text field can contain only the number of characters lesser than or equal to its size. The text fields are alphanumeric in some tables and alphabetic in other tables. Incorrect entry always flashes and error message.

Numeric Field:

The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error messages. The individual modules are checked for accuracy and what it has to perform. Each module is subjected to test run along with sample data. The individually tested modules are integrated into a single system. Testing involves executing the real data information is used in the program the existence of any program defect is inferred from the output. The testing should be planned so that all the requirements are individually tested.

A successful test is one that gives out the defects for the inappropriate data and produces an output revealing the errors in the system.

Preparation of Test Data

Taking various kinds of test data does the above testing. Preparation of test data plays a vital role in the system testing. After preparing the test data the system under study is tested using that test data. While testing the system by using test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

Using Live Test Data:

Live test data are those that are actually extracted from organization files. After a system is partially constructed, programmers or analysts often ask users to key in a set of data from their normal activities. Then, the systems person uses this data as a way to partially test the system. In other instances, programmers or analysts extract a set of live data from the files and have them entered themselves.

It is difficult to obtain live data in sufficient amounts to conduct extensive testing. And, although it is realistic data that will show how the system will perform for the typical processing requirement, assuming that the live data entered are in fact typical, such data generally will not test all combinations or formats that can enter the system. This bias toward typical values then does not provide a true systems test and in fact ignores the cases most likely to cause system failure.

Using Artificial Test Data:

Artificial test data are created solely for test purposes, since they can be generated to test all combinations of formats and values. In other words, the artificial data, which can quickly be prepared by a data generating utility program in the information systems department, make possible the testing of all login and control paths through the program.

The most effective test programs use artificial test data generated by persons other than those who wrote the programs. Often, an independent team of testers formulates a testing plan, using the systems specifications.

The package “Virtual Private Network” has satisfied all the requirements specified as per software requirement specification and was accepted.

USER TRAINING

Whenever a new system is developed, user training is required to educate them about the working of the system so that it can be put to efficient use by those for whom the system has been primarily designed. For this purpose the normal working of the project was demonstrated to the prospective users. Its working is easily understandable and since the expected users are people who have good knowledge of computers, the use of this system is very easy.

MAINTAINENCE

This covers a wide range of activities including correcting code and design errors. To reduce the need for maintenance in the long run, we have more accurately defined the user's requirements during the process of system development. Depending on the requirements, this system has been developed to satisfy the needs to the largest possible extent. With development in technology, it may be possible to add many more features based on the requirements in future. The coding and designing is simple and easy to understand which will make maintenance easier.

TESTING STRATEGY :

A strategy for system testing integrates system test cases and design techniques into a well planned series of steps that results in the successful construction of software. The testing strategy must co-operate test planning, test case design, test execution, and the resultant data collection and evaluation .A strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high level tests that validate major system functions against user requirements.

Software testing is a critical element of software quality assurance and represents the ultimate review of specification design and coding. Testing represents an interesting anomaly for the software. Thus, a series of testing

are performed for the proposed system before the system is ready for user acceptance testing.

SYSTEM TESTING:

Software once validated must be combined with other system elements (e.g. Hardware, people, database). System testing verifies that all the elements are proper and that overall system function performance is achieved. It also tests to find discrepancies between the system and its original objective, current specifications and system documentation.

UNIT TESTING:

In unit testing different modules are tested against the specifications produced during the design for the modules. Unit testing is essential for verification of the code produced during the coding phase, and hence the goals to test the internal logic of the modules. Using the detailed design description as a guide, important Conrail paths are tested to uncover errors within the boundary of the modules. This testing is carried out during the programming stage itself. In this type of testing step, each module was found to be working satisfactorily as regards to the expected output from the module.

In Due Course, latest technology advancements will be taken into consideration. As part of technical build-up many components of the networking system will be generic in nature so that future projects can either use or interact with this. The future holds a lot to offer to the development and refinement of this project.

4.4 PROJECT MANAGEMENT PLAN

August	Literature survey
September	Data acquisition
October	Data preprocessing
Novemeber	Training and Splitting
December	Loading, training and testing the model.
January	Predicting the output and generating the final report

CHAPTER 5

IMPLEMENTATION DETAILS

5.1 DEVELOPMENT SETUP AND DEPLOYMENT SETUP

5.1.1 DEVELOPMENT SETUP:

You can think of Java byte codes as the machine code instructions for the Java Virtual Machine (Java VM). Every Java interpreter, whether it's a Java development tool or a Web browser that can run Java applets, is an implementation of the Java VM. The Java VM can also be implemented in hardware.

Java byte codes help make “write once, run anywhere” possible. You can compile your Java program into byte codes on my platform that has a Java compiler. The byte codes can then be run any implementation of the Java VM. For example, the same Java program can run Windows NT, Solaris, and Macintosh.

5.1.2 DEPLOYMENT SETUP:

Few software packages are designed without goals in mind. JDBC is one that, because of its many goals, drove the development of the API. These goals, in conjunction with early reviewer feedback, have finalized the JDBC class library into a solid framework for building database applications in Java.

The goals that were set for JDBC are important. They will give you some insight as to why certain classes and functionalities behave the way they do. The eight design goals for JDBC are as follows:

1. SQL Level API

The designers felt that their main goal was to define a SQL interface for Java. Although not the lowest database interface level possible, it is at a low enough level for higher-level tools and APIs to be created. Conversely, it is at a high enough level for application programmers to use it confidently. Attaining this goal allows for future tool vendors to “generate” JDBC code and to hide many of JDBC's complexities from the end user.

2. SQL Conformance

SQL syntax varies as you move from database vendor to database vendor. In an effort to support a wide variety of vendors, JDBC will allow any query statement to be passed through it to the underlying database driver. This allows the connectivity module to handle non-standard functionality in a manner that is suitable for its users.

3. JDBC must be implemental on top of common database interfaces

The JDBC SQL API must “sit” on top of other common SQL level APIs. This goal allows JDBC to use existing ODBC level drivers by the use of a software interface. This interface would translate JDBC calls to ODBC and vice versa.

4. Provide a Java interface that is consistent with the rest of the Java system

Because of Java’s acceptance in the user community thus far, the designers feel that they should not stray from the current design of the core Java system.

5. Keep it simple

This goal probably appears in all software design goal listings. JDBC is no exception. Sun felt that the design of JDBC should be very simple, allowing for only one method of completing a task per mechanism. Allowing duplicate functionality only serves to confuse the users of the API.

6. Use strong, static typing wherever possible

Strong typing allows for more error checking to be done at compile time; also, less error appear at runtime.

7. Keep the common cases simple

Because more often than not, the usual SQL calls used by the programmer are simple SELECT’s, INSERT’s, DELETE’s and UPDATE’s, these queries should be simple to perform with JDBC. However, more complex SQL statements should also be possible.

Finally we decided to proceed the implementation using Java [Networking](#).

And for dynamically updating the cache table we go for MS [Access](#) database.

Java has two things: a programming language and a platform.

Java is a high-level programming language that is all of the following

Simple	Architecture-neutral
Object-oriented	Portable
Distributed	High-performance
Interpreted	multithreaded
Robust	Dynamic
Secure	

Java is also unusual in that each Java program is both compiled and interpreted. With a compiler you translate a Java program into an intermediate language called Java byte codes the platform-independent code instruction is passed and run on the computer.

Compilation happens just once; interpretation occurs each time the program is executed. The figure illustrates how this works.

5.2 ALGORITHMS:

- **Data Owner**

In this module, he logs in by using his/her user name and password. After Login the owner Uploads Data, View Files Blocks.

- **End User**

In this module, he logs in by using his/her user name and password. After Login the user will do some operations such as Request Search Permission, Download Request, View All Files, Download File.

- **Fog Server**

In this module, the Fog Server can do following operations such as View Files Blocks, View All Fog User Details and process the end user operations to send data block.

- **Cloud Server**

The Cloud server as a server to provide data storage service and can also do the following operations such as View End Users and Authorize ,View Data Owners and Authorize, View All Stored Data, View Transactions ,View Attackers, View Search Request, View Download_Request,View Files Rank In Chart, View Time Delay In Chart, View Throughput In Chart.

CHAPTER 6

RESULTS AND DISCUSSIONS

The implementation and testing of the blockchain-based unemployment benefits analyst system have yielded valuable insights into its performance, functionality, and potential impact on streamlining the unemployment benefits process. In this section, we present the key results and engage in a discussion of the findings.

6.1 RESULTS:

Functional Verification: The system has successfully demonstrated its ability to handle key functions, including user registration, benefit claim submissions, government verification, benefit disbursement, and data analytics. The smart contracts executed these functions accurately and efficiently.

User Acceptance: User acceptance testing revealed a positive response from claimants, government officials, and administrators. Users found the user interface intuitive and user-friendly, and feedback highlighted that the system streamlined the benefits application and verification process.

Security and Compliance: Security testing identified and addressed vulnerabilities, ensuring that sensitive user data remains protected. Compliance testing confirmed adherence to relevant data protection and privacy regulations, instilling confidence in the system's legal compliance.

Performance and Scalability: Performance testing indicated that the system could handle the expected user load effectively, with acceptable response times and resource utilization. Scalability testing demonstrated that the blockchain network and smart contracts scaled appropriately to accommodate increased demand.

Data Integrity: Data integrity testing confirmed the tamper-proof nature of the blockchain, safeguarding the accuracy and consistency of stored data throughout the system's operation.

Redundancy and Failover: The system exhibited robust redundancy and failover mechanisms, ensuring high availability even in the face of hardware failures or network disruptions.

6.2 DISCUSSION:

The results obtained from the testing phase underscore the system's capability to address critical aspects of unemployment benefits administration. The positive user acceptance feedback aligns with our objective to create an intuitive and accessible platform for claimants, government officials, and administrators. Security and compliance testing have been pivotal in ensuring that the system meets stringent data protection and privacy requirements. By addressing vulnerabilities and implementing secure data handling practices, the system enhances user trust and safeguards sensitive information.

The system's performance and scalability under varying loads indicate its readiness to accommodate the expected user base efficiently. Furthermore, the robust redundancy and failover mechanisms contribute to the system's resilience and availability. The use of blockchain technology has proven its worth in maintaining data integrity and transparency, ensuring that benefit transaction records remain tamper-proof and reliable.

Looking ahead, further refinements and optimizations will be considered based on user feedback and the evolving needs of the unemployment benefits program. Regular maintenance and updates will be essential to sustain the system's performance and security.

CHAPTER 7

CONCLUSION

7.1 CONCLUSION:

The enforcement of access control and the support of keyword search are important issues in secure cloud storage system. In this work, we defined a new paradigm of searchable encryption system, and proposed a concrete construction. It supports flexible multiple keywords subset search, and solves the key escrow problem during the key generation procedure. Malicious user who sells secret key for benefit can be traced. The decryption operation is partly outsourced to cloud server and the correctness of half-decrypted result can be verified by data user. The performance analysis and simulation show its efficiency in computation and storage overhead. Experimental results indicate that the computation overhead at user's terminal is significantly reduced, which greatly saves the energy for resource-constrained devices of users

7.2 FUTURE WORK:

While the automated image colorization system has made significant strides, there are several avenues for future work and enhancements that can further elevate its capabilities and applications. These potential directions represent exciting opportunities for advancing the state of the art in image processing and computer vision:

Advanced Colorization Models: Explore the development of more advanced colorization models, potentially leveraging state-of-the-art deep learning architectures, such as GANs (Generative Adversarial Networks) and Transformers. These models could offer even higher-quality colorizations and better handling of intricate textures.

Multimodal Colorization: Investigate the integration of additional modalities, such as

textual descriptions or audio cues, to guide the colorization process. This could enable users to provide context or preferences, leading to more personalized and accurate colorizations.

Interactive User Guidance: Develop interactive interfaces that allow users to actively guide the colorization process. Users could provide color hints, corrections, or even real-time adjustments, fostering a collaborative approach between AI and human creativity.

Semantic Understanding: Enhance the model's understanding of image semantics and context to improve colorization accuracy. Context-aware colorization can ensure that colors are applied appropriately to objects and scenes, taking into account real-world knowledge.

Data Augmentation and Synthesis: Create techniques for generating synthetic training data to expand the model's ability to handle diverse and rare image types, including historical or artistic styles. Data augmentation strategies can help address data scarcity issues.

Real-Time Colorization: Optimize the model for real-time or near-real-time colorization, making it suitable for applications such as video colorization and live streaming. Hardware acceleration and efficient model architectures are essential for achieving this goal.

Cross-Domain Generalization: Investigate methods for enabling the model to generalize effectively across different domains, such as medical imagery, architectural drawings, or even digital art. Transfer learning and domain adaptation techniques may be valuable.

Privacy-Preserving Colorization: Develop privacy-preserving techniques that allow users to benefit from colorization while protecting sensitive or personal content. Solutions like federated learning or differential privacy could be explored.

Evaluation Metrics Advancements: Advance the development of perceptual and objective evaluation metrics that align more closely with human perception. These metrics should better account for factors like color harmony, realism, and visual appeal.

User-Centered Design: Continuously refine the user interface and user experience design based on user feedback and usability studies. Ensuring that the colorization system is intuitive and accessible remains paramount.

Historical and Cultural Preservation: Collaborate with cultural institutions and historians to apply colorization technology to historical photographs and artworks, preserving and revitalizing cultural heritage.

Cross-Modal Integration: Explore the integration of colorized images with other media forms, such as audio, video, or text, to create compelling cross-modal content and storytelling experiences.

Ethical and Responsible AI: Continue addressing ethical considerations surrounding the use of colorization technology, including guidelines for responsible usage and the handling of sensitive or copyrighted content.

Interdisciplinary Collaboration: Foster collaboration between computer vision researchers, artists, historians, and domain experts to unlock the full potential of colorization technology in various creative and scholarly domains.

7.3 RESEARCH ISSUES:

The field of automated image colorization presents several intriguing research challenges and open questions that researchers and practitioners are actively addressing. These issues encompass a wide range of topics, from improving the accuracy of colorization models to ethical considerations. Here are some prominent research issues in this domain:

Colorization Accuracy: Enhancing the accuracy of colorization models remains a central challenge. Research aims to develop models that can consistently produce realistic and faithful colorizations, particularly in cases with complex textures and fine details.

Data Diversity: Expanding the diversity of training data is essential for models to generalize effectively. Researchers are working on collecting and curating datasets

that encompass a broader range of image types, styles, and historical eras.

Transfer Learning and Domain Adaptation: Investigating transfer learning techniques and domain adaptation methods can enable models to perform well on images from domains they were not explicitly trained on, such as medical images or architectural sketches.

User Interaction: Research explores how users can interact with colorization models, including providing guidance, corrections, or artistic input to customize the colorization process according to their preferences.

Evaluation Metrics: Developing robust and perceptually meaningful evaluation metrics is an ongoing challenge. Researchers are working on metrics that better align with human perception, including those that consider color harmony, realism, and visual appeal.

Ethical Considerations: Addressing ethical issues related to colorization, such as the responsible handling of sensitive or copyrighted content, is a critical area of research. Developing guidelines and best practices for ethical usage is essential.

Privacy-Preserving Colorization: Research aims to create techniques that enable colorization while protecting the privacy of individuals, ensuring that sensitive or personal content is not compromised.

Real-Time Colorization: Achieving real-time or near-real-time colorization for applications like video colorization and live streaming is a research frontier. This requires optimizing model architectures and hardware acceleration.

Explainable AI: Exploring methods for making colorization models more interpretable and explainable can help users understand and trust the colorization process.

Interdisciplinary Collaboration: Collaborative research involving computer vision experts, artists, historians, and domain specialists can uncover new opportunities and applications for colorization technology in diverse fields.

Cross-Modal Integration: Research explores the integration of colorized images with other media forms, enabling cross-modal content creation and storytelling experiences.

Cultural and Historical Preservation: Collaborating with cultural institutions and historians to apply colorization to historical photographs and artworks plays a crucial role in preserving and revitalizing cultural heritage.

Limitations of Colorization: Understanding and communicating the limitations of colorization technology to users is a research challenge. This includes recognizing scenarios where colorization may not be appropriate or may lead to inaccuracies.

Accessibility and Inclusivity: Research focuses on ensuring that colorization technology is accessible to individuals with disabilities, adhering to accessibility guidelines.

AI Fairness: Addressing issues of fairness and bias in colorization models to ensure that colorizations do not inadvertently reinforce stereotypes or biases.

Scalability: Developing colorization models and techniques that can efficiently scale to handle large datasets and high-resolution images is an ongoing area of

research.

Hybrid Models: Exploring hybrid approaches that combine the strengths of automated colorization with human guidance or artistic expertise to achieve more customized results.

7.4 IMPLEMENTATION ISSUES:

The implementation of automated image colorization systems involves several practical challenges and considerations. Addressing these implementation issues is crucial to ensure the successful development and deployment of colorization technology. Here are some key implementation issues to be mindful of:

Data Acquisition and Curation: Acquiring and curating diverse and high-quality datasets for training colorization models can be a time-consuming and resource-intensive task. Researchers and developers must ensure that the data represents a wide range of image types, styles, and historical eras.

Model Selection and Architecture: Choosing the appropriate deep learning architecture and model for colorization is a critical decision. Implementers need to consider factors such as model complexity, computational resources, and the specific requirements of their application.

Training Infrastructure: Training deep learning models for colorization typically requires significant computational resources. Implementers must have access to the necessary hardware and infrastructure for model training.

Data Preprocessing: Preprocessing image data, including resizing, normalization, and augmentation, is essential to prepare the dataset for training. Implementers need to develop efficient preprocessing pipelines to handle large datasets.

Hyperparameter Tuning: Optimizing hyperparameters, such as learning rates,

batchsizes, and model architectures, is crucial for achieving the best colorization results. Implementers must conduct extensive experiments to fine-tune these parameters.

Model Evaluation: Developing robust evaluation procedures to assess the performance of colorization models is essential. Implementers need to choose appropriate evaluation metrics and conduct both quantitative and qualitative assessments.

User Interface Design: Designing a user-friendly interface for users to interact with the colorization system is a critical implementation aspect. User experience (UX) design considerations should be integrated into the development process.

Scalability: Ensuring that the colorization system can handle a large number of user requests or high-resolution images is important, especially for applications that require real-time or batch processing.

Privacy and Security: Implementing privacy and security measures to protect user data and prevent unauthorized access to colorized images is essential. This includes secure data storage and transmission protocols.

Ethical Considerations: Implementers should establish guidelines and policies for responsible usage of the colorization technology, especially when dealing with sensitive or copyrighted content. Ethical considerations should be part of the system's implementation.

Model Deployment: Deploying the colorization model to production environments or integrating it into existing workflows may involve challenges related to server setup, load balancing, and scalability.

Performance Optimization: Optimizing the performance of the colorization model, including reducing inference time and resource utilization, is critical for real-time or resource-constrained applications.

Documentation and Support: Providing comprehensive documentation and

user support resources is essential for users and developers who want to leverage the colorization system effectively.

Maintenance and Updates: Regular maintenance and updates are necessary to address issues, improve model performance, and stay up-to-date with advancements in the field of deep learning.

Legal Compliance: Ensuring legal compliance with intellectual property laws, data protection regulations, and copyright policies is essential, especially when handling user-provided or historical images.

Cross-Platform Compatibility: Ensuring that the colorization system works seamlessly across various platforms, including web, mobile, and desktop, requires careful implementation and testing.

Accessibility: Implementing accessibility features to make the colorization system usable by individuals with disabilities is an ethical and legal requirement in many jurisdictions.

REFERENCES

- [1] Dowelani, M., Okoro, C., & Olaleye, A. (2022). Factors influencing blockchain adoption in the South African clearing and settlement industry. *South African journal of economic and management sciences*, 25(1), 1-11.
- [2] Mahdi, Z. Z., & Baqer, J. A. A. (2023). The role of applying blockchain technology in the accounting and auditing professions, and the opportunities and challenges they face-a prospective study. *World Economics and Finance Bulletin*, 24, 172-182.
- [3] Dowelani, M., Okoro, C., & Olaleye, A. (2022). Factors influencing blockchain adoption in the South African clearing and settlement industry. *South African journal of economic and management sciences*, 25(1), 1-11.
- [4] Alshareef, N., & Tunio, M. N. (2022). Role of leadership in adoption of blockchain technology in small and medium enterprises in Saudi Arabia. *Frontiers in Psychology*, 13, 911432.
- [5] Rithul, V., & Srivastava, V. (2023). The Impact of Cryptocurrency on the Growth Prospect of Indian Economy. *Handbook of Evidence Based Management Practices in Business*, 117.
- [6] Xiao, P., Salleh, M. I., Zaidan, B. B., & Xuelan, Y. (2023). Research on risk assessment of Blockchain-driven Supply Chain finance: A systematic review. *Computers & Industrial Engineering*, 108990.
- [7] Wan, H., Li, K., Huang, Y., & Zhang, L. (2023). Blockchain and Financial E-services. In *Springer Handbook of Automation* (pp. 1371-1383). Cham: Springer International Publishing.

[8] Almeida, J., & Gonçalves, T. C. (2022). Portfolio diversification, hedge and safe- haven properties in cryptocurrency investments and financial economics: A systematic literature review. *Journal of Risk and Financial Management*, 16(1), 3.

[9] Juma'h, A. H., & Li, Y. (2023). The effects of auditors' knowledge, professional skepticism, and perceived adequacy of accounting standards on their intention to use blockchain. *International Journal of Accounting Information Systems*, 51, 100650.

[10] Futerman, A. G., & Sarjanovic, I. A. (2022). *Commodities as an Asset Class: Essays on Inflation, the Paradox of Gold and the Impact of Crypto*. Springer Nature.

APPENDIX

A.SOURCE CODE:

```
<%@ page import="java.sql.*"%>
<%@ page import="java.util.*" %>
<%
    Connection connection = null;
    try {

        Class.forName("com.mysql.jdbc.Driver");
        connection =
DriverManager.getConnection("jdbc:mysql://localhost:3306/Efficient_Trac
eable","root","root");
        String sql="";

    }
    catch(Exception e)
    {
        System.out.println(e);
    }
%>

<%@ page language="java" contentType="text/html; charset=ISO-8859-1 "
    pageEncoding="ISO-8859-1"%>
<%@ page import="java.util.*"%>
<%@ include file="connect.jsp"%>
<%@ page
    import="java.util.*,java.security.Key,java.util.Random,javax.crypto.Cip
her,javax.crypto.spec.SecretKeySpec,org.bouncycastle.util.encoders.Bas
e64"%>
<%@ page
import="java.sql.*,java.util.Random,java.io.PrintStream,java.io.FileOutputS
tream,java.io.FileInputStream,java.security.DigestInputStream,java.math.B
igInteger,java.security.MessageDigest,java.io.BufferedInputStream"%>
<%@ page
import="java.security.Key,java.security.KeyPair,java.security.KeyPairGen
erator,javax.crypto.Cipher"%>
<%@ page
```

```
import="java.util.*,java.text.SimpleDateFormat,java.util.Date,java.io.Fil  
eInputStream,java.io.FileOutputStream,java.io.PrintStream"%>
```

```
<%
```

```
String name = request.getParameter("userid");
```

```
String pass = request.getParameter("pass");
```

```
try {
```

```
String sql = "SELECT * FROM cloud where name='" + name+ "  
and pass='" + pass + "' ";
```

```
Statement stmt = connection.createStatement();
```

```
ResultSet rs = stmt.executeQuery(sql);
```

```
if (rs.next()==true)
```

```
{
```

```
response.sendRedirect("CS_Main.jsp");
```

```
}
```

```
else
```

```
{
```

```
response.sendRedirect("wronglogin.html");
```

```
}
```

```
}
```

```
catch (Exception e)
```

```
{
```

```
out.print(e);
```

```
e.printStackTrace();
```

```
}
```

```
%>
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```
<%@ page import="java.sql.*"%>
```

```
<%@ include file="connect.jsp" %>
```

```
<%@ page
```

```
import="com.oreilly.servlet.*,java.lang.*,java.text.SimpleDateFormat,java.i
```

```

o.*,javax.servlet.*, javax.servlet.http.*" %>
<%@page import
="java.util.*,java.security.Key,java.util.Random,javax.crypto.Cipher,javax.c
rypto.spec.SecretKeySpec"%>
<%@ page import="org.bouncycastle.util.encoders.Base64"%>
<%@ page import="java.util.Random,java.io.PrintStream,
java.io.FileOutputStream, java.io.FileInputStream,
java.security.DigestInputStream, java.math.BigInteger,
java.security.MessageDigest, java.io.BufferedInputStream" %>

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Cloud </title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-titillium-250.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
<style type="text/css">
<!--
.style7 {font-size: 14px; font-weight: bold; }
.style14 {color: #FF0000}
.style16 {font-size: 14px; font-weight: bold; color: #FF0000; }
-->
</style>
</head>
<body>
<div class="main">
  <div class="header">
    <div class="header_resize">
      <div class="logo">
        <h1><a href="index.html"><span>Efficient Traceable Authorization
Search<br />
System for Secure Cloud Storage</span></a><a
href="index.html"><span></span></a></h1>
      </div>
      <div class="menu_nav">

```



```

<ul>
  <li class="active"><a href="CS_Main.jsp"><span>Home
</span></a></li>
  <li><a href="CS_Login.jsp"><span>Logout</span></a></li>
  <li><a href="GM_Login.jsp"></a></li>
  <li><a href="TA_Login.jsp"></a></li>
  <li><a href="CS_Login.jsp"></a></li>
</ul>
</div>
<div class="clr"></div>
<div class="slider">
  <div id="coin-slider"><a href="#"></a> <a href="#"></a> <a
href="#"></a></div>
  <div class="clr"></div>
</div>
<div class="clr"></div>
</div>
<div class="content">
  <div class="content_resize">
    <div class="mainbar">
      <div class="article">
        <h2><span>Authorize Data Owners </span></h2>
        <p class="infopost">&nbsp;</p>
        <div class="clr"></div>
        <div class="article">
          <table width="602" border="2" style="border-collapse:collapse"
align="center" cellpadding="0" cellspacing="0">
            <tr>
              <td width="84" height="32" bgcolor="#FFFF00"><div
align="center" class="style7 style14">User Image </div></td>
              <td width="72" bgcolor="#FFFF00"><div align="center"
class="style16"> Name </div></td>
              <td width="126" bgcolor="#FFFF00"><div align="center"
class="style16">E-Mail </div></td>
              <td width="86" bgcolor="#FFFF00"><div align="center"
class="style16">Mobile</div></td>

```

```

        <td width="92" bgcolor="#FFFF00"><div align="center"
class="style16">Address</div></td>
        <td width="94" bgcolor="#FFFF00"><div align="center"
class="style16">Status </div></td>
    </tr>
<%

```

```

String
s1="",s2="",s3="",s4="",s5="",s6="",s7="",s8="",s9,s10,s11,s12,s13="";
int i=0;

```

```

try
{

```

```

    String query="select * from downer";
    Statement st=connection.createStatement();
    ResultSet rs=st.executeQuery(query);
    while ( rs.next() )
    {

```

```

        i=rs.getInt(1);
        s1=rs.getString(2);
        s2=rs.getString(4);
        s3=rs.getString(5);
        s4=rs.getString(6);
        s5=rs.getString(12);

```

```

        %>

```

```

<tr>

```

```

    <td height="85"><div align="center">

```

```

        <input name="image" type="image"
src="profileimage1.jsp?imgid=<%=i%>" width="80" height="80"
alt="Submit" />

```

```

    </div></td>

```

```

    <td><div align="center"><%=s1%></div></td>

```

```

    <td><div align="center"><%=s2%></div></td>

```

```

    <td><div align="center"><%=s3%></div></td>

```

```

    <td><div align="center"><%=s4%></div></td>

```

```

        <%
                                if(s5.equalsIgnoreCase("Waiting"))
                                {
                                        %>

                                <td><div align="center"><a
href="CS_DAStatus.jsp?id=<%=i%>"><%=s5%></div></td>
                                <%
                                        }
                                else
                                {
                                        %>

                                <td width="32"><div align="center"><%=s5%></div></td>
                                <%
                                        }

                                }
                                connection.close();
                                }

                                catch(Exception e)
                                {
                                        out.println(e.getMessage());
                                }
                                %>

                                </tr>
                                </table>
                                <p align="justify">&nbsp;</p>
                                <p align="right"><a href="CS_Main.jsp">Back</a></p>
                                </div>
                                <div class="clr"></div>
                                </div>
                                </div>
                                <div class="sidebar">
                                <div class="searchform">
                                <form id="formsearch" name="formsearch" method="post"
action="#">
                                <span>
                                <input name="editbox_search" class="editbox_search"
id="editbox_search" maxlength="80" value="Search :" type="text" />
                                </span>

```

```

        <input name="button_search" src="images/search.gif"
class="button_search" type="image" />
    </form>
</div>
<div class="clr"></div>
<div class="g1">
    <div class="g2">
        <div class="gadget">
            <h2 class="star">Menu</h2>
            <div class="clr"></div>
            <ul class="sb_menu">

                <li><a href="CS_Main.jsp">Home </a></li>
                <li><a href="CS_ViewAttacker.jsp">Attackers</a></li>
                <li><a href="CS_ViewTransaction.jsp">Transactions</a></li>
                <li><a href="CS_ViewFiles.jsp">Files & Videos</a></li>
                <li><a href="CS_ViewGC.jsp">Group Certificates</a></li>
                <li><a href="CS_ViewResults.jsp">Group Users Chart </a></li>
                <li><a href="CS_DecryptPer.jsp">Decrypt Permission </a></li>
                <li><a href="CS_Login.jsp">Logout</a></li>

            </ul>
        </div>
    </div>
</div>
</div>
<div class="clr"></div>
</div>
<div class="footer"></div>
</div>
<div align=center></div>
</body>
</html>
<%@ include file="connect.jsp" %>
<%
try{
    String id=request.getParameter("id");
    String str="Authorized";

```

```

        Statement st1=connection.createStatement();
        String query1="update downer set status='"+str+"' where id='"+id+"'";
        st1.executeUpdate(query1);
        connection.close();
        response.sendRedirect("CS_DAAuthorize.jsp");
    }
    catch(Exception e)
    {
        out.println(e);
    }
    %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@page import = "java.util.*"%>
<%@page import = "java.sql.*"%>
<%@page import
="java.util.*,java.security.Key,java.util.Random,javax.crypto.Cipher,javax.c
rypto.spec.SecretKeySpec,org.bouncycastle.util.encoders.Base64"%>
<%@ page
import="java.sql.*,java.util.Random,java.io.PrintStream,java.io.FileOutputS
tream,java.io.FileInputStream,java.security.DigestInputStream,java.math.B
igInteger,java.security.MessageDigest,java.io.BufferedInputStream" %>
<%@ page import
="java.security.Key,java.security.KeyPair,java.security.KeyPairGenerator,j
avax.crypto.Cipher"%>
<%@page import
="java.util.*,java.text.SimpleDateFormat,java.util.Date,java.io.FileInputStre
am,java.io.FileOutputStream,java.io.PrintStream"%>
<%@ include file="connect.jsp" %>

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Cloud Main</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-titillium-250.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>

```

```

<script type="text/javascript" src="js/coin-slider.min.js"></script>
<style type="text/css">
<!--
.style21 {font-size: 14px}
.style24 {color: #FFFF00}
.style27 {font-weight: bold; font-size: 14px;}
.style28 {font-size: 14px; color: #FFFF00; font-weight: bold; }
-->
</style>
</head>
<body>
<div class="main">
  <div class="header">
    <div class="header_resize">
      <div class="logo">
        <h1><a href="index.html"><span>Efficient Traceable Authorization
Search<br />
        System for Secure Cloud Storage</span></a><a
href="index.html"><span></span></a></h1>
      </div>
      <div class="menu_nav">
        <ul>
          <li class="active"><a href="CS_Main.jsp"><span>Home
</span></a></li>
          <li><a href="CS_Login.jsp"><span>Logout</span></a></li>
          <li><a href="#"></a></li>
          <li><a href="#"></a></li>
          <li><a href="#"></a></li>
        </ul>
      </div>
      <div class="clr"></div>
      <div class="slider">
        <div id="coin-slider"><a href="#"></a> <a href="#"></a> <a
href="#"></a></div>
        <div class="clr"></div>
      </div>
      <div class="clr"></div>
    </div>
  </div>

```

```

</div>
</div>
<div class="content">
  <div class="content_resize">
    <div class="mainbar">
      <div class="article">
        <h2><span>Decryption Permission </span></h2>
        <p class="infopost">&nbsp;</p>
        <div class="clr"></div>
        <div class="article">
          <p align="justify">&nbsp;</p>
          <table width="565" border="2" cellpadding="0" style="border-collapse:collapse" cellspacing="0" align="center">
            <tr>
              <td width="71" height="30" bgcolor="#FF0000"><div align="center" class="style21 style24"><strong>User ID</strong></div></td>
              <td width="112" bgcolor="#FF0000"><div align="center" class="style28">User Name </div></td>
              <td width="112" bgcolor="#FF0000"><div align="center" class="style28">Data Provider </div></td>
              <td width="121" bgcolor="#FF0000"><div align="center" class="style28"><span class="style27">File Name </span></div></td>
              <td width="169" bgcolor="#FF0000"><div align="center" class="style28">Permission</div></td>
            <%

```

```

String
s1="",s2="",s3="",s4="",s5="",s6="",s7="",s8,s9="",s10,s11,s12,s13;
int i=0,j=0,k=0;

try
{
    String str="Group Manager Requested";
    String query="select * from request where
decryptper='"+str+"'";
    Statement st=connection.createStatement();
    ResultSet rs=st.executeQuery(query);
    while ( rs.next() )

```

```

{
    i=rs.getInt(1);
    s2=rs.getString(2);
    s3=rs.getString(3);
    s4=rs.getString(4);
    s5=rs.getString(6);

    %>
</tr>
<tr>
    <td height="28"><div align="center"><%=i%></div></td>
    <td><div align="center"><%=s2%></div></td>
    <td><div align="center"><%=s3%></div></td>
    <td><div align="center"><%=s4%></div></td>
    <%=
        if(s5.equalsIgnoreCase("Group Manager Requested"))
        {

            %>
            <td><div>
                <div align="center"><a
href="CS_DecryptPer1.jsp?usid=<%=i%>"><%=s5%></a></div>
                </div></td>
            <%=

                }else
                {

                    %>
                    <td width="17"><div>
                        <div align="center"><%=s5%></div>
                    </div></td>
                </tr>
            <%=

                }

        }
    connection.close();
}

```



```

        catch(Exception e)
        {
            out.println(e.getMessage());
        }
    %>

    </table>
    <p>&nbsp;</p>
    <p align="right"><a href="CS_Main.jsp">Back</a></p>
</div>
<div class="clr"></div>
</div>
</div>
<div class="sidebar">
    <div class="searchform">
        <form id="formsearch" name="formsearch" method="post"
action="#">
            <span>
                <input name="editbox_search" class="editbox_search"
id="editbox_search" maxlength="80" value="Search :" type="text" />
            </span>
            <input name="button_search" src="images/search.gif"
class="button_search" type="image" />
        </form>
    </div>
    <div class="clr"></div>
    <div class="g1">
        <div class="g2">
            <div class="gadget">
                <h2 class="star">Menu</h2>
                <div class="clr"></div>
                <ul class="sb_menu">
                    <li><a href="CS_Main.jsp">Home </a></li>
                    <li><a href="CS_ViewAttacker.jsp">Attackers</a></li>
                    <li><a href="CS_ViewTransaction.jsp">Transactions</a></li>
                    <li><a href="CS_ViewFiles.jsp">Files & Videos</a></li>
                    <li><a href="CS_ViewGC.jsp">Group Certificates</a></li>
                    <li><a href="CS_DUAuthorize.jsp">User Authorization </a></li>
                    <li><a href="CS_ViewResults.jsp">Group Users Chart </a></li>
                    <li><a href="CS_Login.jsp">Logout</a></li>

```

```

        </ul>
    </div>
</div>
</div>
</div>
<div class="clr"></div>
</div>
</div>
<div class="footer"></div>
</div>
<div align=center></div>
</body>
</html>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page
import="java.sql.*,java.util.Random,java.security.KeyPair,java.security.Key
yPairGenerator,java.security.NoSuchAlgorithmException,java.security.Pu
blicKey,javax.crypto.Cipher,javax.crypto.NoSuchPaddingException" %>
<%@ include file="connect.jsp" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Cloud Decrypt per</title>

```

```

        <%
int uid = Integer.parseInt(request.getParameter("usid"));
try {

String str="Permitted";
Statement st1 = connection.createStatement();
String query1 ="update request set decryptper='"+str+"' where
id='"+uid+"' ";
st1.executeUpdate (query1);
response.sendRedirect("CS_DecryptPer.jsp");
connection.close();
}
catch(Exception e)
{
out.println(e.getMessage());

```

```

}
%>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page import="java.sql.*"%>
<%@ include file="connect.jsp" %>
<%@ page
import="com.oreilly.servlet.*,java.lang.*,java.text.SimpleDateFormat,java.i
o.*,javax.servlet.*, javax.servlet.http.*" %>
<%@page import
="java.util.*,java.security.Key,java.util.Random,javax.crypto.Cipher,javax.c
rypto.spec.SecretKeySpec"%>
<%@ page import="org.bouncycastle.util.encoders.Base64"%>
<%@ page import="java.util.Random,java.io.PrintStream,
java.io.FileOutputStream, java.io.FileInputStream,
java.security.DigestInputStream, java.math.BigInteger,
java.security.MessageDigest, java.io.BufferedInputStream" %>

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Cloud </title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-titillium-250.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
<style type="text/css">
<!--
.style7 {font-size: 14px; font-weight: bold; }
.style14 {color: #FF0000}
.style16 {font-size: 14px; font-weight: bold; color: #FF0000; }
-->
</style>
</head>
<body>
<div class="main">
<div class="header">

```

```

<div class="header_resize">
  <div class="logo">
    <h1><a href="index.html"><span>Efficient Traceable Authorization
Search<br />
System for Secure Cloud Storage</span></a><a
href="index.html"><span></span></a></h1>
  </div>
  <div class="menu_nav">
    <ul>
      <li class="active"><a href="CS_Main.jsp"><span>Home
</span></a></li>
      <li><a href="CS_Login.jsp"><span>Logout</span></a></li>
      <li><a href="GM_Login.jsp"></a></li>
      <li><a href="TA_Login.jsp"></a></li>
      <li><a href="CS_Login.jsp"></a></li>
    </ul>
  </div>
  <div class="clr"></div>
  <div class="slider">
    <div id="coin-slider"><a href="#"></a> <a href="#"></a> <a
href="#"></a></div>
    <div class="clr"></div>
  </div>
  <div class="clr"></div>
</div>
<div class="content">
  <div class="content_resize">
    <div class="mainbar">
      <div class="article">
        <h2><span>Authorize Data Owners </span></h2>
        <p class="infopost">&nbsp;</p>
        <div class="clr"></div>
        <div class="article">
          <table width="602" border="2" style="border-collapse:collapse"
align="center" cellpadding="0" cellspacing="0">
            <tr>

```

```

        <td width="84" height="32" bgcolor="#FFFF00"><div
align="center" class="style7 style14">User Image </div></td>
        <td width="72" bgcolor="#FFFF00"><div align="center"
class="style16"> Name </div></td>
        <td width="126" bgcolor="#FFFF00"><div align="center"
class="style16">E-Mail </div></td>
        <td width="86" bgcolor="#FFFF00"><div align="center"
class="style16">Mobile</div></td>
        <td width="92" bgcolor="#FFFF00"><div align="center"
class="style16">Address</div></td>
        <td width="94" bgcolor="#FFFF00"><div align="center"
class="style16">Status </div></td>
    </tr>
    <%

```

```

String
s1="",s2="",s3="",s4="",s5="",s6="",s7="",s8="",s9,s10,s11,s12,s13="";
int i=0;

```

```

try
{

```

```

    String query="select * from downer";
    Statement st=connection.createStatement();
    ResultSet rs=st.executeQuery(query);
    while ( rs.next() )
    {

```

```

        i=rs.getInt(1);
        s1=rs.getString(2);
        s2=rs.getString(4);
        s3=rs.getString(5);
        s4=rs.getString(6);
        s5=rs.getString(12);

```

```

        %>
    <tr>
        <td height="85"><div align="center">
            <input name="image" type="image"

```

```

src="profileimage1.jsp?imgid=<%=i%>" width="80" height="80"
alt="Submit" />
    </div></td>
    <td><div align="center"><%=s1%></div></td>
    <td><div align="center"><%=s2%></div></td>
    <td><div align="center"><%=s3%></div></td>
    <td><div align="center"><%=s4%></div></td>
    <%
        if(s5.equalsIgnoreCase("Waiting"))
        {
            <td><div align="center"><a
href="CS_DAStatus.jsp?id=<%=i%>"><%=s5%></div></td>
            <%
                }
            else
            {
                <td width="32"><div align="center"><%=s5%></div></td>
                <%
                    }
            }
        connection.close();
    }

    catch(Exception e)
    {
        out.println(e.getMessage());
    }
%>
    </tr>
</table>
<p align="justify">&nbsp;</p>
<p align="right"><a href="CS_Main.jsp">Back</a></p>
</div>
<div class="clr"></div>
</div>
</div>

```

```

<div class="sidebar">
  <div class="searchform">
    <form id="formsearch" name="formsearch" method="post"
action="#">
      <span>
        <input name="editbox_search" class="editbox_search"
id="editbox_search" maxlength="80" value="Search :" type="text" />
      </span>
      <input name="button_search" src="images/search.gif"
class="button_search" type="image" />
    </form>
  </div>
  <div class="clr"></div>
  <div class="g1">
    <div class="g2">
      <div class="gadget">
        <h2 class="star">Menu</h2>
        <div class="clr"></div>
        <ul class="sb_menu">

          <li><a href="CS_Main.jsp">Home </a></li>
          <li><a href="CS_ViewAttacker.jsp">Attackers</a></li>
          <li><a href="CS_ViewTransaction.jsp">Transactions</a></li>
          <li><a href="CS_ViewFiles.jsp">Files & Videos</a></li>
          <li><a href="CS_ViewGC.jsp">Group Certificates</a></li>
          <li><a href="CS_ViewResults.jsp">Group Users Chart </a></li>
          <li><a href="CS_DecryptPer.jsp">Decrypt Permission </a></li>
          <li><a href="CS_Login.jsp">Logout</a></li>

        </ul>
      </div>
    </div>
  </div>
  <div class="clr"></div>
</div>
<div align=center></div>

```

```

</body>
</html>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page import="java.sql.*"%>
<%@ include file="connect.jsp" %>
<%@ page
import="com.oreilly.servlet.*,java.lang.*,java.text.SimpleDateFormat,java.i
o.*,javax.servlet.*, javax.servlet.http.*" %>
<%@page import
="java.util.*,java.security.Key,java.util.Random,javax.crypto.Cipher,javax.c
rypto.spec.SecretKeySpec"%>
<%@ page import="org.bouncycastle.util.encoders.Base64"%>
<%@ page import="java.util.Random,java.io.PrintStream,
java.io.FileOutputStream, java.io.FileInputStream,
java.security.DigestInputStream, java.math.BigInteger,
java.security.MessageDigest, java.io.BufferedInputStream" %>

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Cloud </title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-titillium-250.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
<style type="text/css">
<!--
.style7 {font-size: 14px; font-weight: bold; }
.style14 {color: #FF0000}
.style16 {font-size: 14px; font-weight: bold; color: #FF0000; }
-->
</tr>
</table>
<p align="justify">&nbsp;</p>
<p align="right"><a href="CS_Main.jsp">Back</a></p>
</div>

```



```

        <div class="clr"></div>
    </div>
</div>
<div class="sidebar">
    <div class="searchform">
        <form id="formsearch" name="formsearch" method="post"
action="#">
            <span>
                <input name="editbox_search" class="editbox_search"
id="editbox_search" maxlength="80" value="Search :" type="text" />
            </span>
            <input name="button_search" src="images/search.gif"
class="button_search" type="image" />
        </form>
    </div>
    <div class="clr"></div>
    <div class="g1">
        <div class="g2">
            <div class="gadget">
                <h2 class="star">Menu</h2>
                <div class="clr"></div>
                <ul class="sb_menu">

                    <li><a href="CS_Main.jsp">Home </a></li>
                    <li><a href="CS_ViewAttacker.jsp">Attackers</a></li>
                    <li><a href="CS_ViewTransaction.jsp">Transactions</a></li>
                    <li><a href="CS_ViewFiles.jsp">Files & Videos</a></li>
                    <li><a href="CS_ViewGC.jsp">Group Certificates</a></li>
                    <li><a href="CS_ViewResults.jsp">Group Users Chart </a></li>
                    <li><a href="CS_DecryptPer.jsp">Decrypt Permission </a></li>
                    <li><a href="CS_Login.jsp">Logout</a></li>

                </ul>
            </div>
        </div>
    </div>
</div>
<div class="clr"></div>
</div>
</div>

```

```

    <div class="footer"></div>
</div>
<div align=center></div>
</body>
</html>
<%@ include file="connect.jsp" %>
<%
try{
    String id=request.getParameter("id");
    String str="Authorized";
    Statement st1=connection.createStatement();
    String query1="update user set status='"+str+"' where id='"+id+"'";
    st1.executeUpdate(query1);
    connection.close();
    response.sendRedirect("CS_DUAuthorize.jsp");
}
catch(Exception e)
{
    out.println(e);
}
%>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
    <%@page import ="java.util.*"%>
<%@ include file="connect.jsp" %>
    <%@page import
="java.util.*,java.security.Key,java.util.Random,javax.crypto.Cipher,javax.c
rypto.spec.SecretKeySpec,org.bouncycastle.util.encoders.Base64"%>
    <%@ page
import="java.sql.*,java.util.Random,java.io.PrintStream,java.io.FileOutputS
tream,java.io.FileInputStream,java.security.DigestInputStream,java.math.B
igInteger,java.security.MessageDigest,java.io.BufferedInputStream" %>
    <%@ page import
="java.security.Key,java.security.KeyPair,java.security.KeyPairGenerator,j
avax.crypto.Cipher"%>
    <%@page import
="java.util.*,java.text.SimpleDateFormat,java.util.Date,java.io.FileInputStre
am,java.io.FileOutputStream,java.io.PrintStream"%>

```

```

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Cloud </title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-titillium-250.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
<style type="text/css">
<!--
.style21 {font-size: 14px}
.style24 {
    font-size: 16px;
    font-weight: bold;
}
.style31 {color: #FFFF00}
.style32 {font-size: 14px; color: #FFFF00; }
-->
</style>
</head>
<body>
<div class="main">
    <div class="header">
        <div class="header_resize">
            <div class="logo">
                <h1><a href="index.html"><span>Efficient Traceable Authorization
Search<br />
System for Secure Cloud Storage</span></a><a
href="index.html"><span></span></a></h1>
            </div>
            <div class="menu_nav">
                <ul>
                    <li class="active"><a href="CS_Main.jsp"><span>Home
</span></a></li>
                    <li><a href="CS_Login.jsp"><span>Logout</span></a></li>

```

```

        <li><a href="#"></a></li>
        <li><a href="#"></a></li>
        <li><a href="#"></a></li>
    </ul>
</div>
<div class="clr"></div>
<div class="slider">
    <div id="coin-slider"><a href="#"></a> <a href="#"></a> <a
href="#"></a></div>
    <div class="clr"></div>
</div>
<div class="clr"></div>
</div>
<div class="content">
    <div class="content_resize">
        <div class="mainbar">
            <div class="article">
                <h2><span> Files</span> Details </h2>
                <p class="infopost">&nbsp;</p>
                <div class="clr"></div>
                <div class="article">
                    <p class="style24">&nbsp;</p>
                    <p class="style24">
                        <%

```

```

String
s1="",s2="",s3="",s4="",s5="",s6="",s7="",s8="",s9="",s10,s11,s12,s13;
int i=0,j=0,k=0;
String fname=request.getParameter("fname");
try
{
    String query="Select *from cloudserver where
fname='"+fname+"'";
    Statement st=connection.createStatement();

```

```
ResultSet rs=st.executeQuery(query);
```

```
while ( rs.next() )
```

```
{
```

```
    i=rs.getInt(1);
```

```
    s2=rs.getString(2);
```

```
    s7=rs.getString(3);
```

```
    s3=rs.getString(4);
```

```
    s4=rs.getString(6);
```

```
    s5=rs.getString(7);
```

```
    s6=rs.getString(8);
```

```
}
```

```
%>
```

```
</p>
```

```
<table width="568" style="border-collapse:collapse" border="2"
cellpadding="0" cellspacing="0" align="center">
```

```
<tr>
```

```
<td width="159" height="40" bgcolor="#FF0000"><div
align="center" class="style21 style31">
```

```
<div align="left"><strong> ID :</strong></div>
```

```
</div></td>
```

```
<td width="393"><label>
```

```
<input name="textfield" type="text" value="<%=i%>" size="49"
```

```
/>
```

```
</label>
```

```
</td>
```

```
</tr>
```

```
<tr>
```

```
<td height="40" bgcolor="#FF0000"><div align="center"
class="style32">
```

```
<div align="left"><strong>File Name : </strong></div>
```

```
</div></td>
```

```
<td><input name="textfield2" type="text" size="49"
value="<%=s2%>" /> </td>
```

```
</tr>
```

```
<tr>
```

```
<td height="182" bgcolor="#FF0000"><div align="center"
class="style32">
```

```
<div align="left"><strong>Contents :</strong></div>
```

```

        </div></td>
        <td><label>
            <textarea rows="10" cols="50"
name="textarea"><%=s7%></textarea>
        </label>        </td>
    </tr>
    <tr>
        <td height="41" bgcolor="#FF0000"><div align="center"
class="style32">
            <div align="left"><strong>Trapdoor :</strong></div>
        </div></td>
        <td><input name="textfield3" type="text" value="<%=s3%>"
size="49" /></td>
    </tr>
    <tr>
        <td height="40" bgcolor="#FF0000"><div align="center"
class="style32">
            <div align="left"><strong>Rank :</strong></div>
        </div></td>
        <td><input name="textfield32" type="text" value="<%=s4%>"
size="49" /></td>
    </tr>
    <tr>
        <td height="38" bgcolor="#FF0000"><div align="center"
class="style32">
            <div align="left"><strong>Date & Time</strong></div>
        </div></td>
        <td><input name="textfield33" type="text" value="<%=s5%>"
size="49" /></td>
    </tr>
    <tr>
        <td height="43" bgcolor="#FF0000"><div align="center"
class="style32">
            <div align="left"><strong>Data Owner</strong></div>
        </div></td>
        <td><input name="textfield34" type="text" value="<%=s6%>"
size="49" /></td>
    </tr>

```

```

connection.close();

```

```

    }

    catch(Exception e)
    {
        out.println(e.getMessage());
    }
%>

</tr>
</table>
<p align="right"><a href="d_DataMain.jsp"></a></p>
<p align="right"><a href="CS_ViewFiles.jsp">Back</a></p>
</div><div class="clr"></div>
</div>
</div>
<div class="sidebar">
    <div class="searchform">
        <form id="formsearch" name="formsearch" method="post"
action="#">
            <span>
                <input name="editbox_search" class="editbox_search"
id="editbox_search" maxlength="80" value="Search :" type="text" />
            </span>
            <input name="button_search" src="images/search.gif"
class="button_search" type="image" />
        </form>
    </div>
<div class="clr"></div>
<div class="g1">
    <div class="g2">
        <div class="gadget">
            <h2 class="star">Menu</h2>
            <div class="clr"></div>
            <ul class="sb_menu">
                <li><a href="CS_Main.jsp">Home </a></li>
                <li><a href="CS_ViewAttacker.jsp">Attackers</a></li>
                <li><a href="CS_ViewTransaction.jsp">Transactions</a></li>
                <li><a href="CS_ViewGC.jsp">Group Certificates</a></li>
                <li><a href="CS_DUAuthorize.jsp">User Authorization </a></li>
                <li><a href="CS_ViewResults.jsp">Group Users Chart </a></li>
            </ul>
        </div>
    </div>
</div>

```

```

        <li><a href="CS_DecryptPer.jsp">Decrypt Permission </a></li>
        <li><a href="CS_Login.jsp">Logout</a></li>
    </ul>
</div>
</div>
</div>
</div>
<div class="clr"></div>
</div>
</div>
<div class="footer"></div>
</div>
<div align=center></div>
</body>
</html>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Cloud Login</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-titillium-250.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
<style type="text/css">
<!--
.style1 {color: #FF0000}
.style3 {color: #FF0000; font-weight: bold; }
-->
</style>
</head>
<body>
<div class="main">
    <div class="header">

```



```

<div class="header_resize">
  <div class="logo">
    <h1><a href="index.html"><span>Efficient Traceable Authorization
Search<br />
System for Secure Cloud Storage</span></a><a
href="index.html"><span></span></a></h1>
  </div>
  <div class="menu_nav">
    <ul>
      <li><a href="index.html"><span>Home </span></a></li>
      <li><a href="DU_Login.jsp"><span>Data User </span></a></li>
      <li><a href="DO_Login.jsp"><span>Data Owner </span></a></li>

      <li><a href="TA_Login.jsp"><span>KGC</span></a></li>
      <li class="active"><a href="CS_Login.jsp"><span>Cloud Server
</span></a></li>
    </ul>
  </div>
  <div class="clr"></div>
  <div class="slider">
    <div id="coin-slider"><a href="#"></a> <a href="#"></a> <a
href="#"></a></div>
    <div class="clr"></div>
  </div>
  <div class="clr"></div>
</div>
<div class="content">
  <div class="content_resize">
    <div class="mainbar">
      <div class="article">
        <h2><span>Cloud Login </span></h2>
        <p class="infopost">&nbsp;</p>
        <div class="clr"></div>
        <div class="article">
          <form action="CS_Authentication.jsp" method="post"
id="leavereply">

```

[illegible]

```

        </form>
    </div>
    <div class="clr"></div>
    <div class="g1">
        <div class="g2">
            <div class="gadget">
                <h2 class="star">Menu</h2>
                <div class="clr"></div>
                <ul class="sb_menu">
                    <li><a href="DU_Login.jsp">Data User</a></li>
                    <li><a href="DO_Login.jsp">Data Owner </a></li>
                    <li><a href="CS_Login.jsp">Cloud Server</a></li>

                    <li><a href="TA_Login.jsp">KGC</a></li>
                </ul>
            </div>
        </div>
    </div>
    <div class="clr"></div>
</div>
<div class="footer"></div>
</div>
<div align=center></div>
</body>
</html>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Cloud Main</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-titillium-250.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>

```

```

<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
<style type="text/css">
<!--
.style1 {
    color: #FF0000;
    font-weight: bold;
}
-->
</style>
</head>
<body>
<div class="main">
    <div class="header">
        <div class="header_resize">
            <div class="logo">
                <h1><a href="index.html"><span>Efficient Traceable Authorization
Search<br />
System for Secure Cloud Storage</span></a><a
href="index.html"><span></span></a></h1>
            </div>
            <div class="menu_nav">
                <ul>
                    <li class="active"><a href="CS_Main.jsp"><span>Home
</span></a></li>
                    <li><a href="CS_Login.jsp"><span>Logout</span></a></li>
                    <li><a href="#"></a></li>
                    <li><a href="#"></a></li>
                    <li><a href="#"></a></li>
                </ul>
            </div>
            <div class="clr"></div>
            <div class="slider">
                <div id="coin-slider"><a href="#"></a> <a href="#"></a> <a
href="#"></a></div>
                <div class="clr"></div>
            </div>
        </div>
    </div>

```

```

    <div class="clr"></div>
  </div>
</div>
<div class="content">
  <div class="content_resize">
    <div class="mainbar">
      <div class="article">
        <h2><span>Welcome Cloud Server </span></h2>
        <p class="infopost">&nbsp;</p>
        <div class="clr"></div>
        <div class="img"></div>
        <div class="post_content">
          <p align="justify" class="style1">Secure search over encrypted
remote data is crucial in cloud computing to guarantee the data privacy
and usability. To prevent unauthorized data usage, fine-grained access
control is necessary in multi-user system. However, authorized user may
intentionally leak the secret key for financial benefit. Thus, tracing and
revoking the malicious user who abuses secret key needs to be solved
imminently. In this paper, we propose an escrow free traceable attribute
based multiple keywords subset search system with verifiable outsourced
decryption (EF-TAMKS-VOD). The key escrow free mechanism could
effectively prevent the key generation centre (KGC) from unscrupulously
searching and decrypting all encrypted files of users. Also, the decryption
process only requires ultra lightweight computation, which is a desirable
feature for energy-limited devices. In addition, efficient user revocation is
enabled after the malicious user is figured out. Moreover, the proposed
system is able to support flexible number of attributes rather than
polynomial bounded. Flexible multiple keyword subset search pattern is
realized, and the change of the query keywords order does not affect the
search result. Security analysis indicates that EF-TAMKS-VOD is provably
secure. Efficiency analysis and experimental results show that EF-
TAMKS-VOD improves the efficiency and greatly reduces the computation
overhead of users' terminals.</p>
          <p class="spec">&nbsp;</p>
        </div>
      </div>
    <div class="clr"></div>
  </div>
</div>
<div class="sidebar">
  <div class="searchform">
    <form id="formsearch" name="formsearch" method="post"
action="#">

```

```

    <span>
        <input name="editbox_search" class="editbox_search"
id="editbox_search" maxlength="80" value="Search :" type="text" />
    </span>
    <input name="button_search" src="images/search.gif"
class="button_search" type="image" />
</form>
</div>
<div class="clr"></div>
<div class="g1">
    <div class="g2">
        <div class="gadget">
            <h2 class="star">Menu</h2>
            <div class="clr"></div>
            <ul class="sb_menu">
                <li><a href="CS_Main.jsp">Home </a></li>
                <li><a href="CS_DUAuthorize.jsp">View Users and
Authorization </a></li>
                <li><a href="CS_DOAuthorize.jsp">View Data
Owners and Authorization </a></li>
                <li><a href="CS_ViewAttacker.jsp">View Traitors</a></li>
                <li><a href="CS_ViewTransaction.jsp">View
Transactions</a></li>
                <li><a href="CS_ViewFiles.jsp">View Files &
Videos</a></li>
                <li><a href="CS_ViewGC.jsp">View Group Certificates</a></li>
                <li><a href="CS_DecryptPer.jsp">View Decrypt
Permission </a></li>

                <li>-----GROUP ACTIVITIES -----</li>

                <li><a href="GM_ViewCerti.jsp">View Certificates
</a></li>
                <li><a href="GM_UserGCPer.jsp">User Requests </a></li>
                <li><a href="GM_OwnerGCPer.jsp">Owner Requests</a></li>
                <li><a href="GM_DecryptPer.jsp">Decryption Requests </a></li>
                <li><a href="GM_GenCerti.jsp">Generate Certificate </a></li>

                <li><a href="CS_ViewResults.jsp">View Group

```

Users Result

```

        <li><a href="CS_Login.jsp">Logout</a></li>
    </ul>
</div>
</div>
</div>
</div>
<div class="clr"></div>
</div>
</div>
<div class="fbg"></div>
<div class="footer"></div>
</div>
<div align=center></div>
</body>
</html>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page language="java" contentType="text/html; charset=ISO-8859-1 "
    pageEncoding="ISO-8859-1"%>
    <%@page import ="java.util.*"%>
<%@ include file="connect.jsp" %>
    <%@page import
="java.util.*,java.security.Key,java.util.Random,javax.crypto.Cipher,javax.c
rypto.spec.SecretKeySpec,org.bouncycastle.util.encoders.Base64"%>
    <%@ page
import="java.sql.*,java.util.Random,java.io.PrintStream,java.io.FileOutputS
tream,java.io.FileInputStream,java.security.DigestInputStream,java.math.B
igInteger,java.security.MessageDigest,java.io.BufferedInputStream" %>
<%@ page import
="java.security.Key,java.security.KeyPair,java.security.KeyPairGenerator,j
avax.crypto.Cipher"%>
    <%@page import
="java.util.*,java.text.SimpleDateFormat,java.util.Date,java.io.FileInputStre
am,java.io.FileOutputStream,java.io.PrintStream"%>

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Cloud </title>
```

```

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-titillium-250.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
<style type="text/css">
<!--
.style21 {font-size: 14px}
.style23 {color: #666666; font-weight: bold;}
.style24 {
    font-size: 16px;
    font-weight: bold;
}
.style29 {color: #6d6d6d}
.style30 {font-size: 14px; color: #6d6d6d; }
-->
</style>
</head>
<body>
<div class="main">
    <div class="header">
        <div class="header_resize">
            <div class="logo">
                <h1><a href="index.html"><span>Efficient Traceable Authorization
Search<br />
System for Secure Cloud Storage</span></a><a
href="index.html"><span></span></a></h1>
            </div>
            <div class="menu_nav">
                <ul>
                    <li class="active"><a href="CS_Main.jsp"><span>Home
</span></a></li>
                    <li><a href="CS_Login.jsp"><span>Logout</span></a></li>
                    <li><a href="#"></a></li>
                    <li><a href="#"></a></li>
                    <li><a href="#"></a></li>
                </ul>
            </div>
        </div>
    </div>

```



```

        </ul>
    </div>
    <div class="clr"></div>
    <div class="slider">
        <div id="coin-slider"><a href="#"></a> <a href="#"></a> <a
href="#"></a></div>
        <div class="clr"></div>
    </div>
    <div class="clr"></div>
</div>
<div class="content">
    <div class="content_resize">
        <div class="mainbar">
            <div class="article">
                <h2><span> Play </span> Video </h2>
                <p class="infopost">&nbsp;</p>
                <div class="clr"></div>
                <div class="article">
                    <p class="style24">&nbsp;</p>
                    <div class="article">
                        <div align="center">
                            <%
int j=0;

String fn=request.getParameter("fname");
String title=request.getParameter("title");

%>
                            <video controls preload="auto" src="./<%=fn%>" width="400"
height="300"></video>
                        </div>
                        <div align="center">

```

```

        <% out.println(fn);%>
    </div>
</div>
<p class="style24">&nbsp;</p>
<p align="right"><a href="CS_ViewFiles.jsp">Back</a></p>
</div><div class="clr"></div>
</div>
</div>
<div class="sidebar">
    <div class="searchform">
        <form id="formsearch" name="formsearch" method="post"
action="#">
            <span>
                <input name="editbox_search" class="editbox_search"
id="editbox_search" maxlength="80" value="Search :" type="text" />
            </span>
            <input name="button_search" src="images/search.gif"
class="button_search" type="image" />
        </form>
    </div>
<div class="clr"></div>
<div class="g1">
    <div class="g2">
        <div class="gadget">
            <h2 class="star">Menu</h2>
            <div class="clr"></div>
            <ul class="sb_menu">
                <li><a href="CS_Main.jsp">Home </a></li>
                <li><a href="CS_ViewAttacker.jsp">Attackers</a></li>
                <li><a href="CS_ViewTransaction.jsp">Transactions</a></li>
                <li><a href="CS_ViewGC.jsp">Group Certificates</a></li>
                <li><a href="CS_DUAuthorize.jsp">User Authorization </a></li>
                <li><a href="CS_ViewResults.jsp">Group Users Chart </a></li>
                <li><a href="CS_DecryptPer.jsp">Decrypt Permission </a></li>
                <li><a href="CS_Login.jsp">Logout</a></li>
            </ul>
        </div>
    </div>
</div>
</div>

```

```

    </div>
    <div class="clr"></div>
  </div>
</div>
<div class="fbg"></div>
<div class="footer"></div>
</div>
<div align=center></div>
</body>
</html>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
    <%@ page import ="java.util.*"%>
<%@ include file="connect.jsp" %>
    <%@ page import
="java.util.*,java.security.Key,java.util.Random,javax.crypto.Cipher,javax.c
rypto.spec.SecretKeySpec,org.bouncycastle.util.encoders.Base64"%>
    <%@ page
import="java.sql.*,java.util.Random,java.io.PrintStream,java.io.FileOutputS
tream,java.io.FileInputStream,java.security.DigestInputStream,java.math.B
igInteger,java.security.MessageDigest,java.io.BufferedInputStream" %>
<%@ page import
="java.security.Key,java.security.KeyPair,java.security.KeyPairGenerator,j
avax.crypto.Cipher"%>
    <%@ page import
="java.util.*,java.text.SimpleDateFormat,java.util.Date,java.io.FileInputStre
am,java.io.FileOutputStream,java.io.PrintStream"%>

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Cloud </title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-titillium-250.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>

```

```

<script type="text/javascript" src="js/coin-slider.min.js"></script>
<style type="text/css">
<!--
.style23 {font-size: 14px; font-weight: bold; }
.style24 {color: #FFFF00}
.style25 {font-size: 14px; font-weight: bold; color: #FFFF00; }
-->
</style>
</head>
<body>
<div class="main">
  <div class="header">
    <div class="header_resize">
      <div class="logo">
        <h1><a href="index.html"><span>Efficient Traceable Authorization
Search<br />
System for Secure Cloud Storage</span></a><a
href="index.html"><span></span></a></h1>
      </div>
      <div class="menu_nav">
        <ul>
          <li class="active"><a href="CS_Main.jsp"><span>Home
</span></a></li>
          <li><a href="CS_Login.jsp"><span>Logout</span></a></li>
          <li><a href="#"></a></li>
          <li><a href="#"></a></li>
          <li><a href="#"></a></li>
        </ul>
      </div>
      <div class="clr"></div>
      <div class="slider">
        <div id="coin-slider"><a href="#"></a> <a href="#"></a> <a
href="#"></a></div>
        <div class="clr"></div>
      </div>
      <div class="clr"></div>
    </div>
  </div>

```

```

</div>
<div class="content">
  <div class="content_resize">
    <div class="mainbar">
      <div class="article">
        <h2><span>View All Traitors </span></h2>
        <p class="infopost">&nbsp;</p>
        <div class="clr"></div>
        <div class="article">
          <p align="justify">&nbsp;</p>
          <table width="600" border="2" style="border-collapse:collapse"
cellpadding="0" cellspacing="0" align="center">
            <tr>
              <td width="118" height="31" bgcolor="#FF0000"><div
align="center" class="style23 style24">User Name </div></td>
              <td width="149" bgcolor="#FF0000"><div align="center"
class="style25">File Name </div></td>
              <td width="165" bgcolor="#FF0000"><div align="center"
class="style25">Secret Key </div></td>
              <td width="165" bgcolor="#FF0000"><div align="center"
class="style25">Date & Time</div></td>
              <td width="167" bgcolor="#FF0000"><div align="center"
class="style25">Attack Type </div></td>
            </tr>
          <%

```

```

String
s1="",s2="",s3="",s4="",s5="",s6="",s7="",s8,s9="",s10,s11,s12,s13;
int i=0,j=0,k=0;

try
{

```

```

String query="select * from Attacker ";
Statement st=connection.createStatement();

```

```

        ResultSet rs=st.executeQuery(query);
while ( rs.next() )
{
    i=rs.getInt(1);
    s2=rs.getString(2);
    s3=rs.getString(3);
    s4=rs.getString(4);
    s5=rs.getString(5);
    s6=rs.getString(6);

    %>
    <tr>
        <td height="29"><div align="center"><%=s2%></div></td>
        <td><div align="center"><%=s3%></div></td>
        <td><div align="center"><%=s4%></div></td>
        <td><div align="center"><%=s5%></div></td>
        <td><div align="center"><%=s6%></div></td>
    </tr>
    <%
}

        connection.close();
    }

    catch(Exception e)
    {
        out.println(e.getMessage());
    }
%>

    </table>
    <p align="right">&nbsp;</p>
    <p align="right"><a href="CS_Main.jsp">Back</a></p>
</div>
<div class="clr"></div>
</div>
</div>
<div class="sidebar">
    <div class="searchform">

```

```

        <form id="formsearch" name="formsearch" method="post"
action="#">
        <span>
        <input name="editbox_search" class="editbox_search"
id="editbox_search" maxlength="80" value="Search :" type="text" />
        </span>
        <input name="button_search" src="images/search.gif"
class="button_search" type="image" />
        </form>
</div>
<div class="clr"></div>
<div class="g1">
    <div class="g2">
        <div class="gadget">
            <h2 class="star">Menu</h2>
            <div class="clr"></div>
            <ul class="sb_menu">
                <li><a href="CS_Main.jsp">Home </a></li>
                <li><a href="CS_ViewTransaction.jsp">Transactions</a></li>
                <li><a href="CS_ViewFiles.jsp">Files & Videos</a></li>
                <li><a href="CS_ViewGC.jsp">Group Certificates</a></li>
                <li><a href="CS_DUAuthorize.jsp">User Authorization </a></li>
                <li><a href="CS_ViewResults.jsp">Group Users Chart </a></li>
                <li><a href="CS_DecryptPer.jsp">Decrypt Permission </a></li>
                <li><a href="CS_Login.jsp">Logout</a></li>
            </ul>
        </div>
    </div>
</div>
</div>
<div class="clr"></div>
</div>
</div>
<div class="fbg"></div>
<div class="footer"></div>
</div>
<div align=center></div>
</body>
</html>

```

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
    <%@page import ="java.util.*"%>
<%@ include file="connect.jsp" %>
    <%@page import
="java.util.*,java.security.Key,java.util.Random,javax.crypto.Cipher,javax.c
rypto.spec.SecretKeySpec,org.bouncycastle.util.encoders.Base64"%>
    <%@ page
import="java.sql.*,java.util.Random,java.io.PrintStream,java.io.FileOutputS
tream,java.io.FileInputStream,java.security.DigestInputStream,java.math.B
igInteger,java.security.MessageDigest,java.io.BufferedInputStream" %>
    <%@ page import
="java.security.Key,java.security.KeyPair,java.security.KeyPairGenerator,j
avax.crypto.Cipher"%>
    <%@page import
="java.util.*,java.text.SimpleDateFormat,java.util.Date,java.io.FileInputStre
am,java.io.FileOutputStream,java.io.PrintStream"%>

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Cloud </title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-titillium-250.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
<style type="text/css">
<!--
.style21 {font-size: 14px}
.style24 {
    font-size: 16px;
    font-weight: bold;
}
.style25 {color: #FFFF00}
.style27 {font-size: 14px; color: #FFFF00; font-weight: bold; }

```



```

-->
</style>
</head>
<body>
<div class="main">
  <div class="header">
    <div class="header_resize">
      <div class="logo">
        <h1><a href="index.html"><span>Efficient Traceable Authorization
Search<br />
System for Secure Cloud Storage</span></a><a
href="index.html"><span></span></a></h1>
      </div>
      <div class="menu_nav">
        <ul>
          <li class="active"><a href="CS_Main.jsp"><span>Home
</span></a></li>
          <li><a href="CS_Login.jsp"><span>Logout</span></a></li>
          <li><a href="#"></a></li>
          <li><a href="#"></a></li>
          <li><a href="#"></a></li>
        </ul>
      </div>
      <div class="clr"></div>
      <div class="slider">
        <div id="coin-slider"><a href="#"></a> <a href="#"></a> <a
href="#"></a></div>
        <div class="clr"></div>
      </div>
      <div class="clr"></div>
    </div>
  </div>
  <div class="content">
    <div class="content_resize">
      <div class="mainbar">
        <div class="article">
          <h2><span>All Files</span> & Videos </h2>

```

```

<p class="infopost">&nbsp;</p>
<div class="clr"></div>
<div class="article">
  <p class="style24">Files </p>
  <p class="style24">&nbsp;</p>
  <table width="600" border="2" cellpadding="0" cellspacing="0"
style="border-collapse:collapse" align="center">
    <tr>
      <td width="51" height="30" bgcolor="#FF0000"><div
align="center" class="style21 style25"><strong> ID </strong></div></td>
      <td width="130" bgcolor="#FF0000"><div align="center"
class="style27">File Name </div></td>
      <td width="120" bgcolor="#FF0000"><div align="center"
class="style27">Data Owner </div></td>
      <td width="120" bgcolor="#FF0000"><div align="center"
class="style27">Secret Key </div></td>
      <td width="140" bgcolor="#FF0000"><div align="center"
class="style27">Date & Time </div></td>
    </tr>
  </table>

```

```

String
s1="",s2="",s3="",s4="",s5="",s6="",s7="",s8,s9="",s10,s11,s12,s13;
int i=0,j=0,k=0;

```

```

try
{
    String query1="Select *from cloudserver ";
    Statement st1=connection.createStatement();
    ResultSet rs1=st1.executeQuery(query1);
    while ( rs1.next() )
    {
        i=rs1.getInt(1);
        s2=rs1.getString(2);
        s3=rs1.getString(7);
        s8=rs1.getString(5);
        s6=rs1.getString(8);

        String keys = "ef50a0ef2c3e3a5f";
    }
}

```

```

byte[] keyValue = keys.getBytes();
Key key = new SecretKeySpec(keyValue,
"AES");

Cipher c = Cipher.getInstance("AES");
c.init(Cipher.DECRYPT_MODE, key);
String dt = new
String(Base64.decode(s3.getBytes()));

        %>

    </tr>
    <tr>
        <td height="28"><div align="center"><%=i%></div></td>
        <td><div align="center"><a
href="CS_filedetails.jsp?fname=<%=s2%>"><%=s2%></div></td>
        <td><div align="center"><%=s6%></div></td>
        <td><div align="center"><%=s8%></div></td>
        <td><div align="center"><%=dt%></div></td>
        <%=
    }

%>

    </tr>
</table>
<p align="justify">&nbsp;</p>
<p align="justify">&nbsp;</p>
<p class="style24">Videos    </p>
<p class="style24">&nbsp;</p>
<table width="600" border="2" cellpadding="0" cellspacing="0"
style="border-collapse:collapse" align="center">
    <tr>
        <td width="51" height="30" bgcolor="#FF0000"><div
align="center" class="style27"> ID </div></td>
        <td width="130" bgcolor="#FF0000"><div align="center"
class="style27">File Name </div></td>
        <td width="120" bgcolor="#FF0000"><div align="center"
class="style27">Data Owner </div></td>
        <td width="120" bgcolor="#FF0000"><div align="center"
class="style27">Secret Key </div></td>
        <td width="140" bgcolor="#FF0000"><div align="center"

```

```
class="style27">Date & Time </div></td>
<%
```

```
String query="Select *from cloud_videos ";
Statement st=connection.createStatement();
ResultSet rs=st.executeQuery(query);
while ( rs.next() )
```

```
{
    i=rs.getInt(1);
    s4=rs.getString(2);
    s5=rs.getString(4);
    s7=rs.getString(6);
    s9=rs.getString(8);
```

```
String keys = "ef50a0ef2c3e3a5f";
byte[] keyValue = keys.getBytes();
Key key = new SecretKeySpec(keyValue, "AES");
Cipher c = Cipher.getInstance("AES");
c.init(Cipher.DECRYPT_MODE, key);
```

```
String s41 = new
String(Base64.decode(s4.getBytes()));
String s51 = new
String(Base64.decode(s5.getBytes()));
String s71 = new
String(Base64.decode(s7.getBytes()));
```

```
%>
```

```
</tr>
```

```
<tr>
```

```

        <td height="28"><div align="center"><%=i%></div></td>
        <td><div align="center"><a
href="CS_playvideo.jsp?fname=<%=s41%>"><%=s41%></a></div></td>
        <td><div align="center"><%=s51%></div></td>
        <td><div align="center"><%=s9%></div></td>
        <td><div align="center"><%=s71%></div></td>
        <%=

    }

    connection.close();
}

catch(Exception e)
{
    out.println(e.getMessage());
}
%>

</tr>
</table>
<p>&nbsp;</p>
<p align="right"><a href="CS_Main.jsp">Back</a></p>
</div><div class="clr"></div>
</div>
</div>
<div class="sidebar">
    <div class="searchform">
        <form id="formsearch" name="formsearch" method="post"
action="#">
            <span>
                <input name="editbox_search" class="editbox_search"
id="editbox_search" maxlength="80" value="Search :" type="text" />
            </span>
            <input name="button_search" src="images/search.gif"
class="button_search" type="image" />
        </form>
    </div>
    <div class="clr"></div>
    <div class="g1">
        <div class="g2">

```

```

<div class="gadget">
  <h2 class="star">Menu</h2>
  <div class="clr"></div>
  <ul class="sb_menu">
    <li><a href="CS_Main.jsp">Home </a></li>
    <li><a href="CS_ViewAttacker.jsp">Attackers</a></li>
    <li><a href="CS_ViewTransaction.jsp">Transactions</a></li>
    <li><a href="CS_ViewGC.jsp">Group Certificates</a></li>
    <li><a href="CS_DUAuthorize.jsp">User Authorization </a></li>
    <li><a href="CS_ViewResults.jsp">Group Users Chart </a></li>
    <li><a href="CS_DecryptPer.jsp">Decrypt Permission </a></li>
    <li><a href="CS_Login.jsp">Logout</a></li>
  </ul>
</div>
</div>
</div>
</div>
<div class="clr"></div>
</div>
</div>
<div class="fbg"></div>
<div class="footer"></div>
</div>
<div align=center></div>
</body>
</html>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page import="java.sql.*"%>
<%@ include file="connect.jsp" %>
<%@ page
import="com.oreilly.servlet.*,java.lang.*,java.text.SimpleDateFormat,java.i
o.*,javax.servlet.*, javax.servlet.http.*" %>
<%@page import
="java.util.*,java.security.Key,java.util.Random,javax.crypto.Cipher,javax.c
rypto.spec.SecretKeySpec"%>
<%@ page import="org.bouncycastle.util.encoders.Base64"%>
<%@ page import="java.util.Random,java.io.PrintStream,
java.io.FileOutputStream, java.io.FileInputStream,
java.security.DigestInputStream, java.math.BigInteger,

```

java.security.MessageDigest, java.io.BufferedInputStream" %>

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Cloud </title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-titillium-250.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
<style type="text/css">
<!--
.style24 {color: #CC3300}
.style25 {color: #FF0000}
.style26 {color: #2A3F55}
-->
</style>
</head>
<body>
<div class="main">
  <div class="header">
    <div class="header_resize">
      <div class="logo">
        <h1><a href="index.html"><span>Efficient Traceable Authorization
Search<br />
System for Secure Cloud Storage</span></a><a
href="index.html"><span></span></a></h1>
      </div>
      <div class="menu_nav">
        <ul>
          <li class="active"><a href="CS_Main.jsp"><span>Home
</span></a></li>
          <li><a href="CS_Login.jsp"><span>Logout</span></a></li>
          <li><a href="#"></a></li>
          <li><a href="#"></a></li>
          <li><a href="#"></a></li>
        </ul>
      </div>
    </div>
  </div>
</body>
</html>
```

```

        </ul>
    </div>
    <div class="clr"></div>
    <div class="slider">
        <div id="coin-slider"><a href="#"></a> <a href="#"></a> <a
href="#"></a></div>
        <div class="clr"></div>
    </div>
    <div class="clr"></div>
</div>
<div class="content">
    <div class="content_resize">
        <div class="mainbar">
            <div class="article">
                <h2><span>Group Certificates </span></h2>
                <p class="infopost">&nbsp;</p>
                <div class="clr"></div>
                <div class="article">
                    <p align="justify">
                        <%

```

```

String
s1="",s2="",s3="",s4="",s5="",s6="",s7="",s8,s9="",s10,s11,s12,s13;
int i=0,j=0,k=0;
String downer=(String)application.getAttribute("dname");

```

```

try
{
    String query="Select *from certificate ";
    Statement st=connection.createStatement();
    ResultSet rs=st.executeQuery(query);
    while ( rs.next() )
    {

```



```

        i=rs.getInt(1);
        s2=rs.getString(2);
        s3=rs.getString(3);
        s6=rs.getString(4);
        s4=rs.getString(5);

        %>

    </p>
    <table width="437" height="134" border="2" align="center"
    style="border-collapse:collapse" cellpadding="0" cellspacing="0"
    bgcolor="#FFBF0">
        <form id="form3" name="form3" method="post"
        action="GM_Certificate.jsp">
            <tr>
                <th width="120" height="38" scope="row"><span
                class="style24">Certificate ID : </span></th>
                <td width="386"><div align="center"><strong><span
                class="style25"><%=s6%></span></strong></div></td>
            </tr>
            <tr>
                <th height="38" scope="row"><span class="style24 ">Group :
                </span></th>
                <td><div align="center" class="style26"><%=s2%></div></td>
            </tr>
            <tr>
                <th height="56" scope="row"><span class="style24 ">Date
                & Time: </span></th>
                <td><div align="center" class="style26"><%=s4%></div>
                <span class="style26">
                <label></label>
                </span></td>
            </tr>
        </form>
    </table>
    <%

        }
        connection.close();
    }

```

```

        catch(Exception e)
        {
            out.println(e.getMessage());
        }
    %>
        <p align="right"><a href="CS_Main.jsp">Back</a></p>
    </div>
    <div class="clr"></div>
</div>
<div class="sidebar">
    <div class="searchform">
        <form id="formsearch" name="formsearch" method="post"
action="#">
            <span>
                <input name="editbox_search" class="editbox_search"
id="editbox_search" maxlength="80" value="Search :" type="text" />
            </span>
            <input name="button_search" src="images/search.gif"
class="button_search" type="image" />
        </form>
    </div>
    <div class="clr"></div>
    <div class="g1">
        <div class="g2">
            <div class="gadget">
                <h2 class="star">Menu</h2>
                <div class="clr"></div>
                <ul class="sb_menu">
                    <li><a href="CS_Main.jsp">Home </a></li>
                    <li><a href="CS_ViewAttacker.jsp">Attackers</a></li>
                    <li><a href="CS_ViewTransaction.jsp">Transactions</a></li>
                    <li><a href="CS_ViewFiles.jsp">Files & Videos</a></li>
                    <li><a href="CS_DUAuthorize.jsp">User Authorization </a></li>
                    <li><a href="CS_ViewResults.jsp">Group Users Chart </a></li>
                    <li><a href="CS_DecryptPer.jsp">Decrypt Permission </a></li>
                    <li><a href="CS_Login.jsp">Logout</a></li>
                </ul>
            </div>
        </div>
    </div>

```

```

        </div>
    </div>
</div>
</div>
<div class="clr"></div>
</div>
</div>
<div class="fbg"></div>
<div class="footer"></div>
</div>
<div align=center></div>
</body>
</html>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Cloud </title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-titillium-250.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
</head>
<body>
<div class="main">
    <div class="header">
        <div class="header_resize">
            <div class="logo">
                <h1><a href="index.html"><span>Efficient Traceable Authorization
Search<br />
System for Secure Cloud Storage</span></a><a
href="index.html"><span></span></a></h1>
            </div>
            <div class="menu_nav">
                <ul>

```

```

        <li class="active"><a href="CS_Main.jsp"><span>Home
</span></a></li>
        <li><a href="CS_Login.jsp"><span>Logout</span></a></li>
        <li><a href="#"></a></li>
        <li><a href="#"></a></li>
        <li><a href="#"></a></li>
    </ul>
</div>
<div class="clr"></div>
<div class="slider">
    <div id="coin-slider"><a href="#"></a> <a href="#"></a> <a
href="#"></a></div>
    <div class="clr"></div>
</div>
<div class="clr"></div>
</div>
<div class="content">
    <div class="content_resize">
        <div class="mainbar">
            <div class="article">
                <h2><span>Number Of Group Users </span></h2>
                <p class="infopost">&nbsp;</p>
                <div class="clr"></div>
                <div class="article">
                    <p align="justify">
                        <iframe width="600" height="350" src="Results.jsp"></iframe>
                    </p>
                    <p align="right"><a href="CS_Main.jsp">Back</a></p>
                </div>
            <div class="clr"></div>
        </div>
    </div>
</div>
<div class="sidebar">
    <div class="searchform">
        <form id="formsearch" name="formsearch" method="post"

```

```

action="#">
    <span>
        <input name="editbox_search" class="editbox_search"
id="editbox_search" maxlength="80" value="Search :" type="text" />
    </span>
    <input name="button_search" src="images/search.gif"
class="button_search" type="image" />
    </form>
</div>
<div class="clr"></div>
<div class="g1">
    <div class="g2">
        <div class="gadget">
            <h2 class="star">Menu</h2>
            <div class="clr"></div>
            <ul class="sb_menu">
                <li><a href="CS_Main.jsp">Home </a></li>
                <li><a href="CS_ViewAttacker.jsp">Attackers</a></li>
                <li><a href="CS_ViewTransaction.jsp">Transactions</a></li>
                <li><a href="CS_ViewFiles.jsp">Files & Videos</a></li>
                <li><a href="CS_ViewGC.jsp">Group Certificates</a></li>
                <li><a href="CS_DUAuthorize.jsp">User Authorization </a></li>
                <li><a href="CS_DecryptPer.jsp">Decrypt Permission </a></li>
                <li><a href="CS_Login.jsp">Logout</a></li>
            </ul>
        </div>
    </div>
</div>
</div>
<div class="clr"></div>
</div>
</div>
<div class="fbg"></div>
<div class="footer"></div>
</div>
<div align="center">
</div>
</body>
</html>

```

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page import="java.sql.*"%>
<%@ include file="connect.jsp" %>
<%@ page
import="com.oreilly.servlet.*,java.lang.*,java.text.SimpleDateFormat,java.i
o.*,javax.servlet.*, javax.servlet.http.*" %>
<%@page import
="java.util.*,java.security.Key,java.util.Random,javax.crypto.Cipher,javax.c
rypto.spec.SecretKeySpec"%>
<%@ page import="org.bouncycastle.util.encoders.Base64"%>
<%@ page import="java.util.Random,java.io.PrintStream,
java.io.FileOutputStream, java.io.FileInputStream,
java.security.DigestInputStream, java.math.BigInteger,
java.security.MessageDigest, java.io.BufferedInputStream" %>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Data Owner </title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-titillium-250.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
<style type="text/css">
<!--
.style24 {color: #CC3300}
.style25 {color: #FF0000}
.style26 {color: #2A3F55}
-->
</style>
</head>
<body>
<div class="main">
<div class="header">
<div class="header_resize">
<div class="logo">
<h1><a href="index.html"><span>Efficient Traceable Authorization

```

```

Search<br />
System for Secure Cloud Storage</span></a><a
href="index.html"><span></span></a></h1>
</div>
<div class="menu_nav">
<ul>
<li class="active"><a href="DO_Main.jsp"><span>Home
</span></a></li>
<li><a href="DO_Login.jsp">Logout</a></li>
<li><a href="#"><span> </span></a></li>
<li><a href="#"><span> </span></a></li>
<li><a href="#"><span></span></a></li>
<li><a href="#"><span> </span></a></li>
</ul>
</div>
<div class="clr"></div>
<div class="slider">
<div id="coin-slider"><a href="#"></a> <a href="#"></a> <a
href="#"></a></div>
<div class="clr"></div>
</div>
<div class="clr"></div>
</div>
<div class="content">
<div class="content_resize">
<div class="mainbar">
<div class="article">
<h2><span>View Group Certificate</span></h2>
<p class="infopost">&nbsp;</p>
<div class="clr"></div>
<div class="article">
<p align="justify">
<%

```

```

String
s1="",s2="",s3="",s4="",s5="",s6="",s7="",s8,s9="",s10,s11,s12,s13;
int i=0,j=0,k=0;
String downer=(String)application.getAttribute("doname");

try
{

String query1="Select *from certificaterequest where
user='"+downer+"' ";
Statement st1=connection.createStatement();
ResultSet rs1=st1.executeQuery(query1);
if (rs1.next())
{

String cerid=rs1.getString(4);

if(cerid.equalsIgnoreCase("Requested"))
{

%>
</p>
<h2 class="style27">Wait For Group Manager To
Issue Certificate !! </h2>
<p><a href="DO_ReqGC.jsp"
class="style28">Back</a></p>
<p>
<%

}
else
{

String query="Select *from certificate where
cerid='"+cerid+"' ";
Statement st=connection.createStatement();
ResultSet rs=st.executeQuery(query);

```



```

while ( rs.next() )
{
    i=rs.getInt(1);
    s2=rs.getString(2);
    s3=rs.getString(3);
    s6=rs.getString(4);
    s4=rs.getString(5);

    %>
</p>
<p>&nbsp;</p>
<table width="437" height="136" border="2" align="center"
cellpadding="0" cellspacing="0" style="border-collapse:collapse"
bgcolor="#FFFBF0">
    <form id="form3" name="form3" method="post"
action="GM_Certificate.jsp">
        <tr>
            <th width="120" height="38" scope="row"><span
class="style24">Certificate ID : </span></th>
            <td width="386"><div align="center"><strong><span
class="style25"><%=s6%></span></strong></div></td>
        </tr>
        <tr>
            <th height="38" scope="row"><span class="style24 ">Group :
</span></th>
            <td><div align="center" class="style26"><%=s2%></div></td>
        </tr>
        <tr>
            <th height="56" scope="row"><span class="style24 ">Date
&amp; Time: </span></th>
            <td><div align="center" class="style26"><%=s4%></div>
                <span class="style26">
                    <label></label>
                </span></td>
        </tr>
    </form>
</table>
<%

```

```

        }
    }
}
else
{
    %>
    </p>
    <h2 class="style27">You Have Not Requested For Group
Certificate !! </h2>
    <p><a href="DO_ReqGC.jsp"
class="style28">Back</a></p>
    <p>
    <%

    }
    connection.close();
}

catch(Exception e)
{
    out.println(e.getMessage());
}
%>
<p align="right"><a href="DO_ReqGC.jsp">Back</a></p>
</div>
<div class="clr"></div>
</div>
</div>
<div class="sidebar">
    <div class="searchform">
        <form id="formsearch" name="formsearch" method="post"
action="#">
            <span>
                <input name="editbox_search" class="editbox_search"
id="editbox_search" maxlength="80" value="Search :" type="text" />
            </span>
            <input name="button_search" src="images/search.gif"
class="button_search" type="image" />

```

```

        </form>
    </div>
    <div class="clr"></div>
    <div class="g1">
        <div class="g2">
            <div class="gadget">
                <h2 class="star">Menu</h2>
                <div class="clr"></div>
                <ul class="sb_menu">
                    <li><a href="DO_Main.jsp">Home</a></li>
                    <li><a href="DO_Files.jsp">Files</a></li>
                    <li><a href="DO_Videos.jsp">Videos</a></li>
                    <li><a href="DO_Login.jsp">Logout</a></li>
                </ul>
            </div>
        </div>
    </div>
    <div class="clr"></div>
</div>
<div class="fbg"></div>
<div class="footer"></div>
</div>
<div align=center></div>
</body>
</html>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
    <%@ page import ="java.util.*"%>
    <%@ include file="connect.jsp" %>
    <%@ page import
="java.util.*,java.security.Key,java.util.Random,javax.crypto.Cipher,javax.c
rypto.spec.SecretKeySpec,org.bouncycastle.util.encoders.Base64"%>
    <%@ page
import="java.sql.*,java.util.Random,java.io.PrintStream,java.io.FileOutputS
tream,java.io.FileInputStream,java.security.DigestInputStream,java.math.B

```

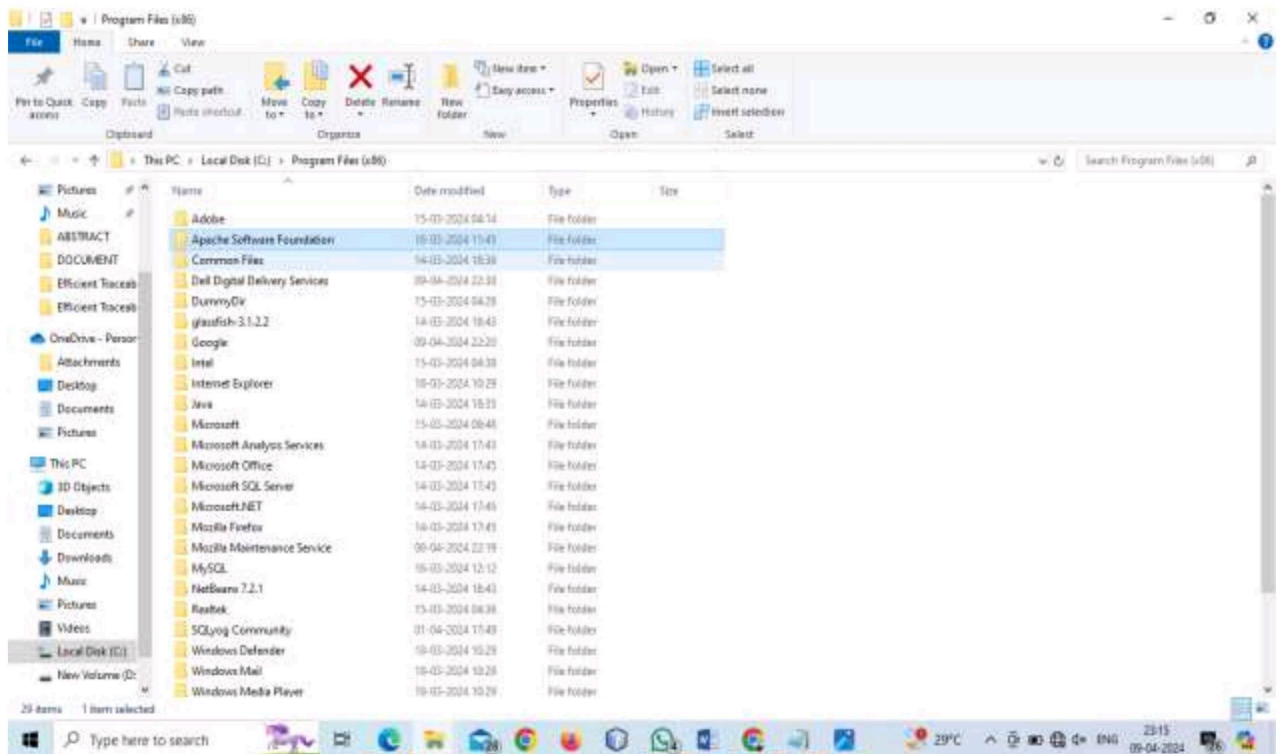
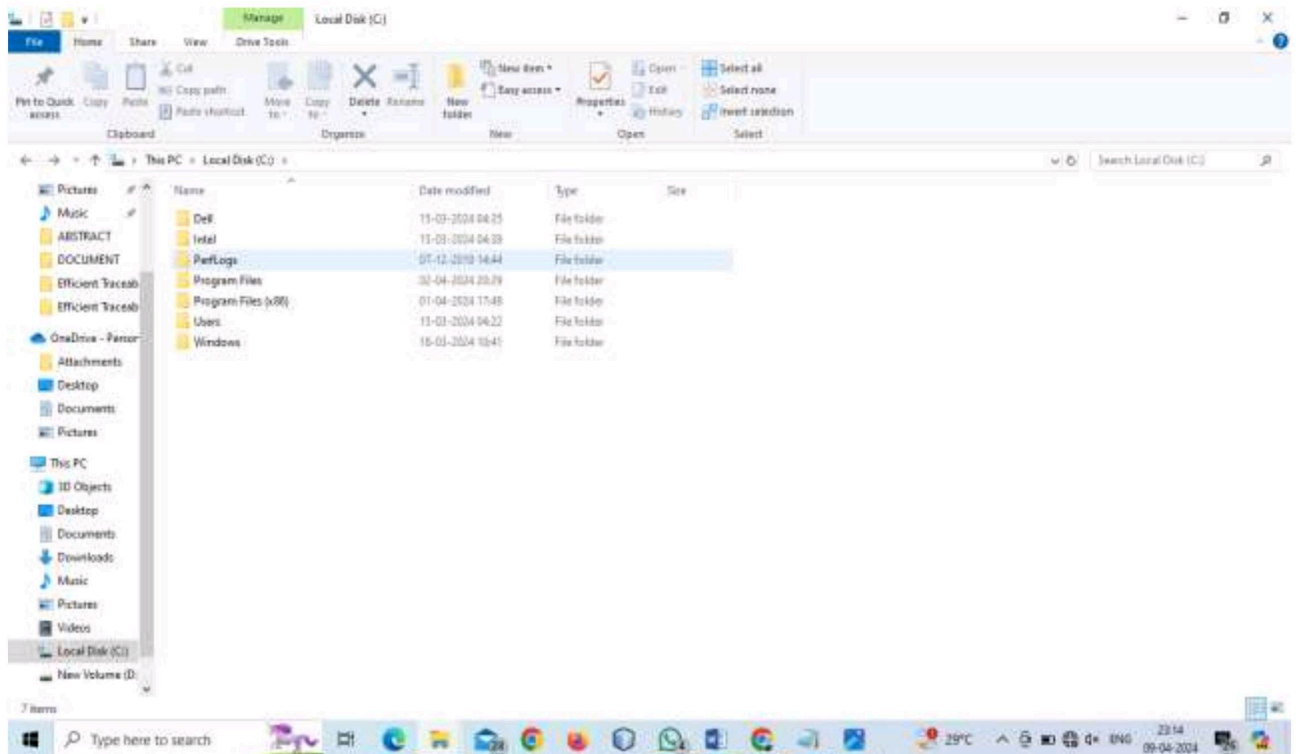
```

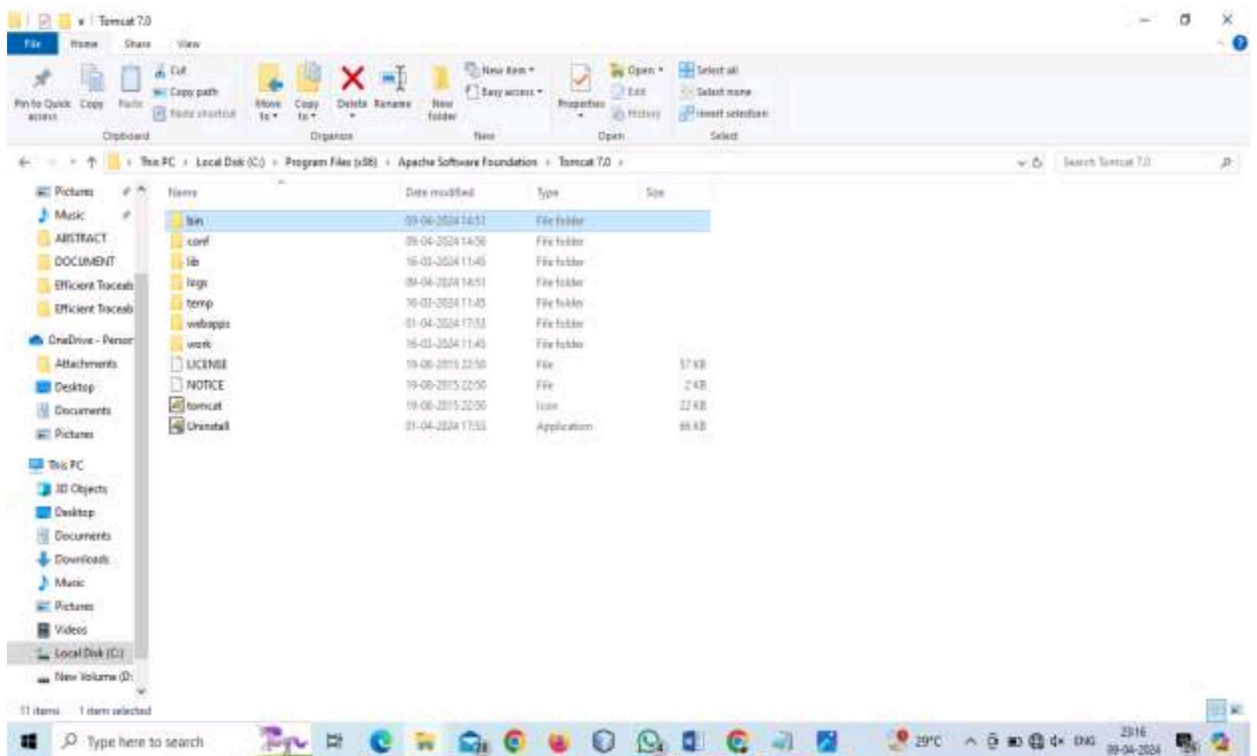
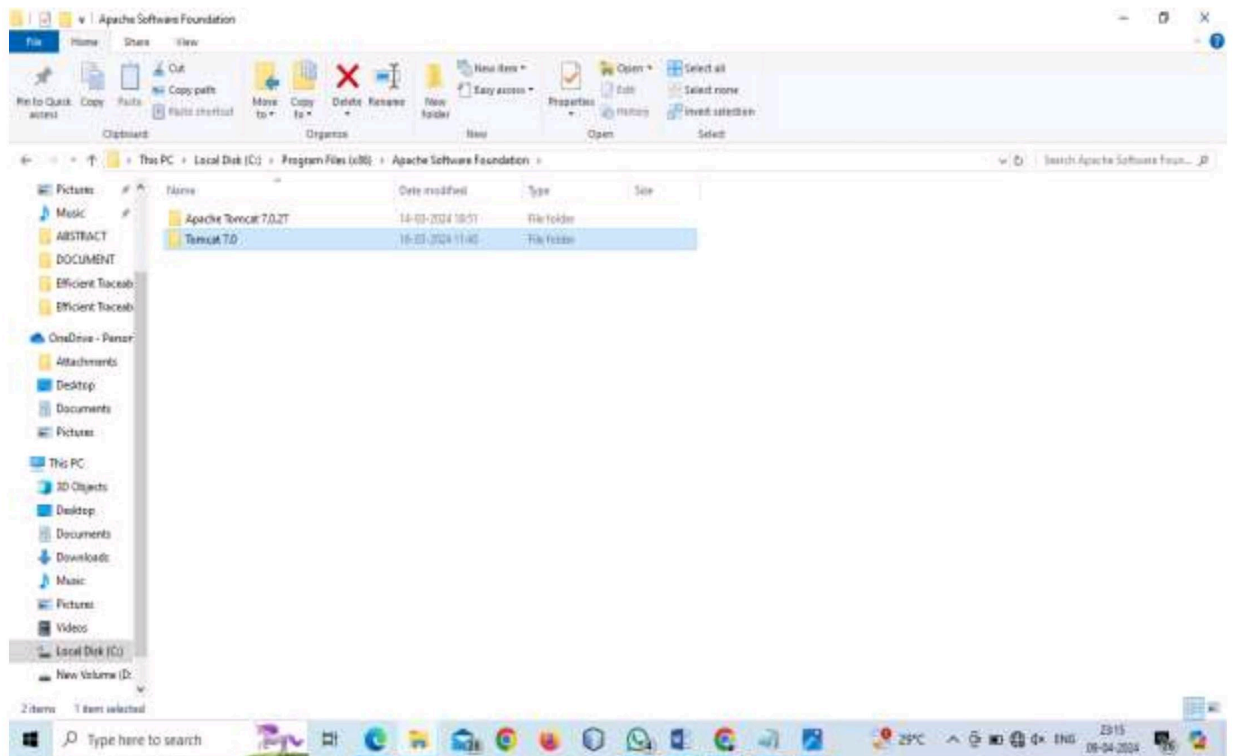
Integer,java.security.MessageDigest,java.io.BufferedInputStream" %>
<%@ page import
="java.security.Key,java.security.KeyPair,java.security.KeyPairGenerator,j
avax.crypto.Cipher"%>
<%@ page import
="java.util.*,java.text.SimpleDateFormat,java.util.Date,java.io.FileInputStre
am,java.io.FileOutputStream,java.io.PrintStream"%>

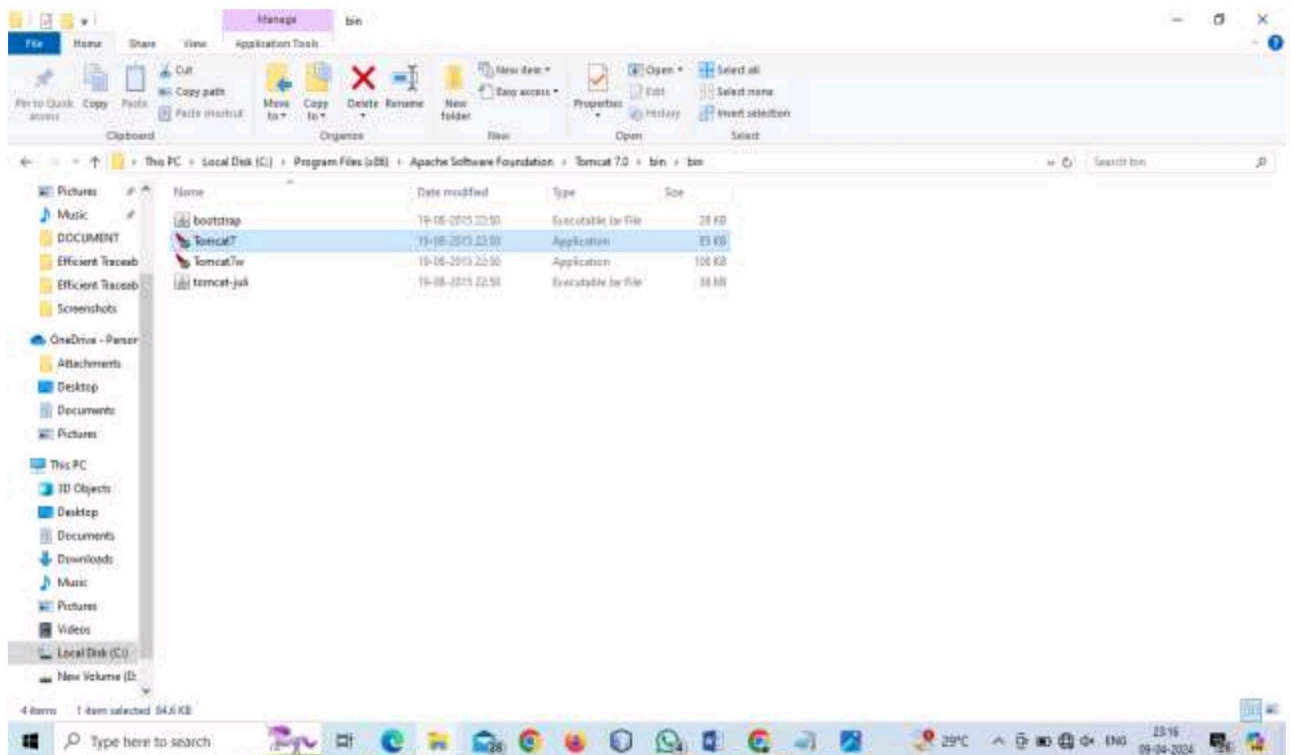
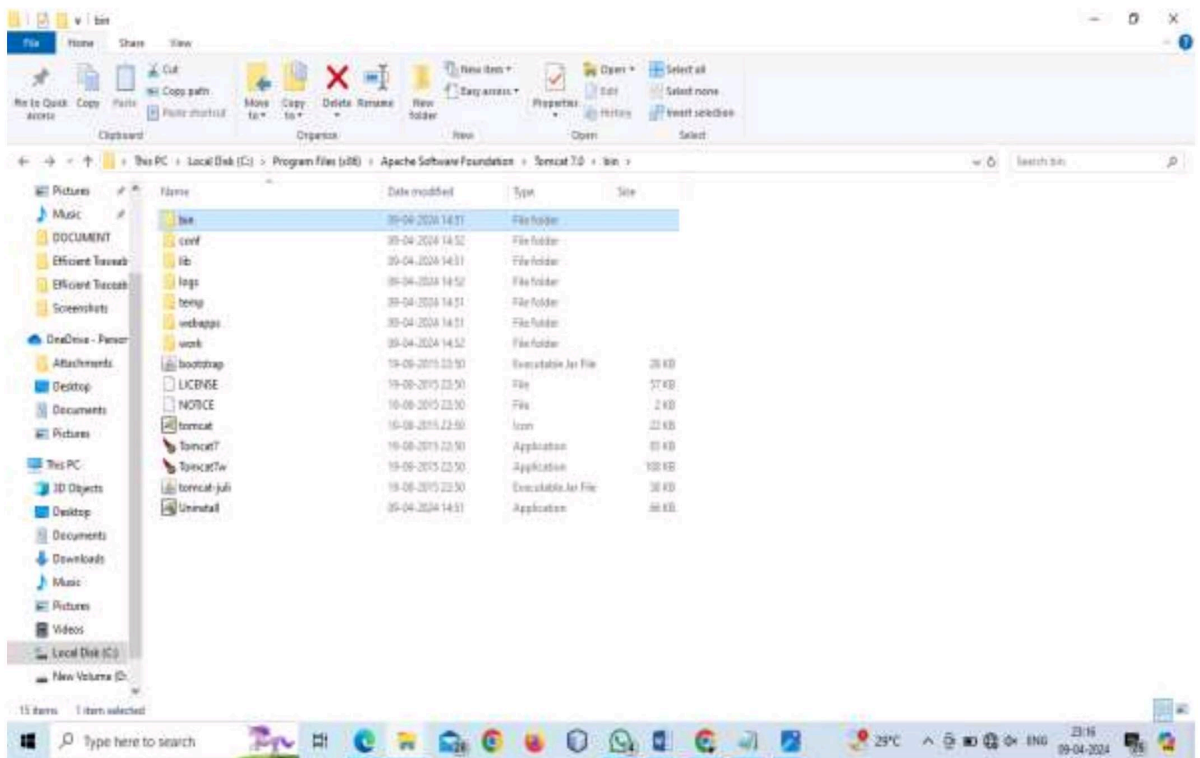
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Data Owner </title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link href="css/style.css" rel="stylesheet" type="text/css" />
<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />
<script type="text/javascript" src="js/cufon-yui.js"></script>
<script type="text/javascript" src="js/cufon-titillium-250.js"></script>
<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="js/script.js"></script>
<script type="text/javascript" src="js/coin-slider.min.js"></script>
<style type="text/css">
<!--
.style21 {font-size: 14px}
.style24 {color: #FFFF00}
.style25 {
    font-size: 14px;
    color: #FFFF00;
    font-weight: bold;
}

```

B. SCREENSHOTS:





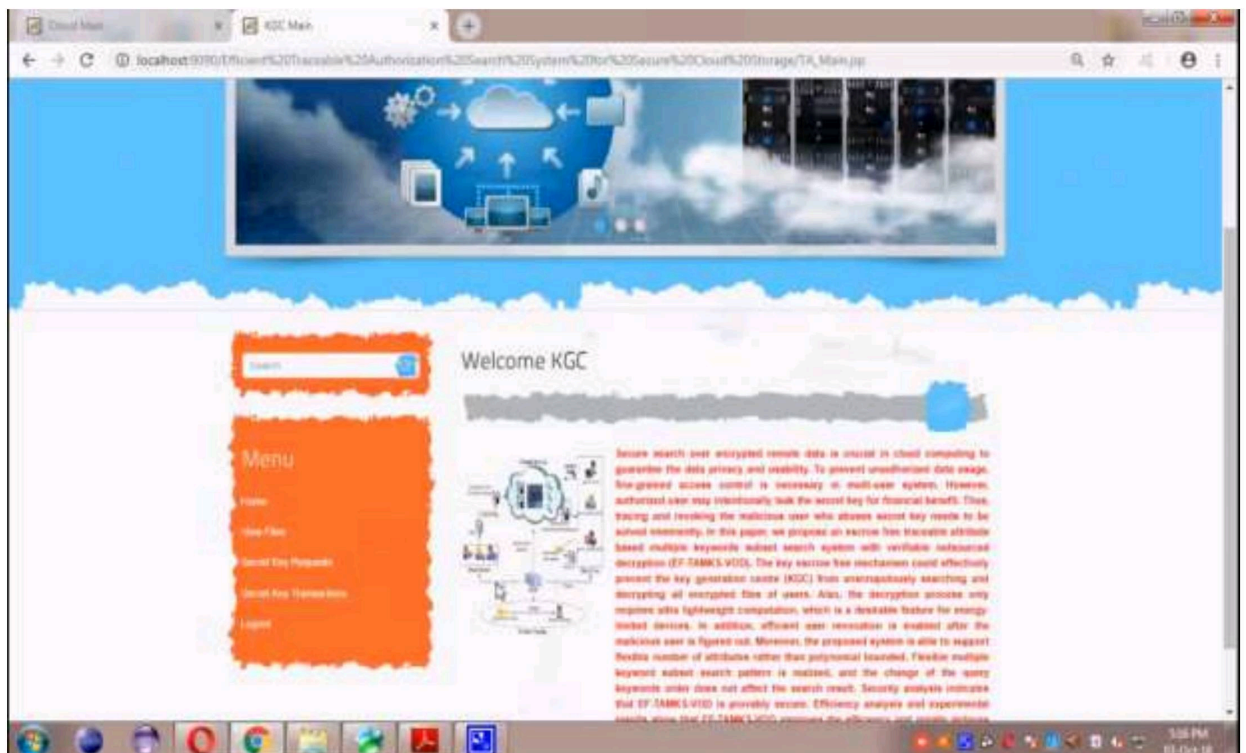
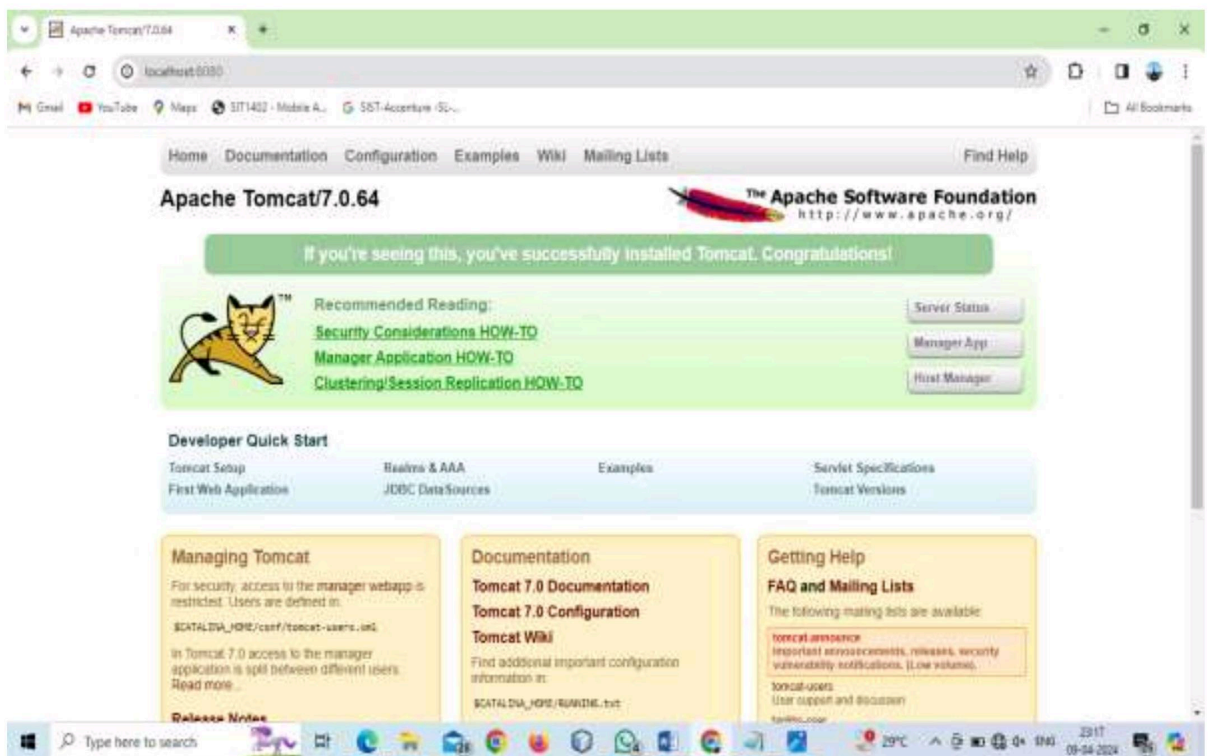


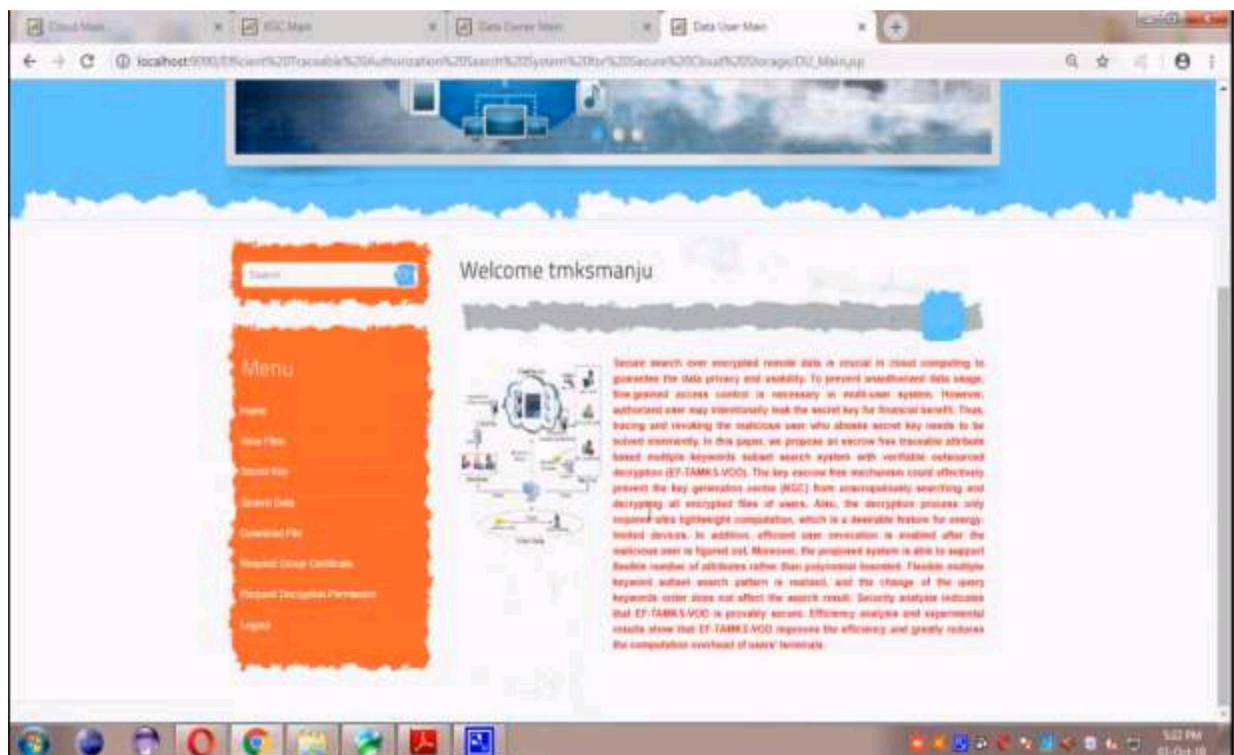
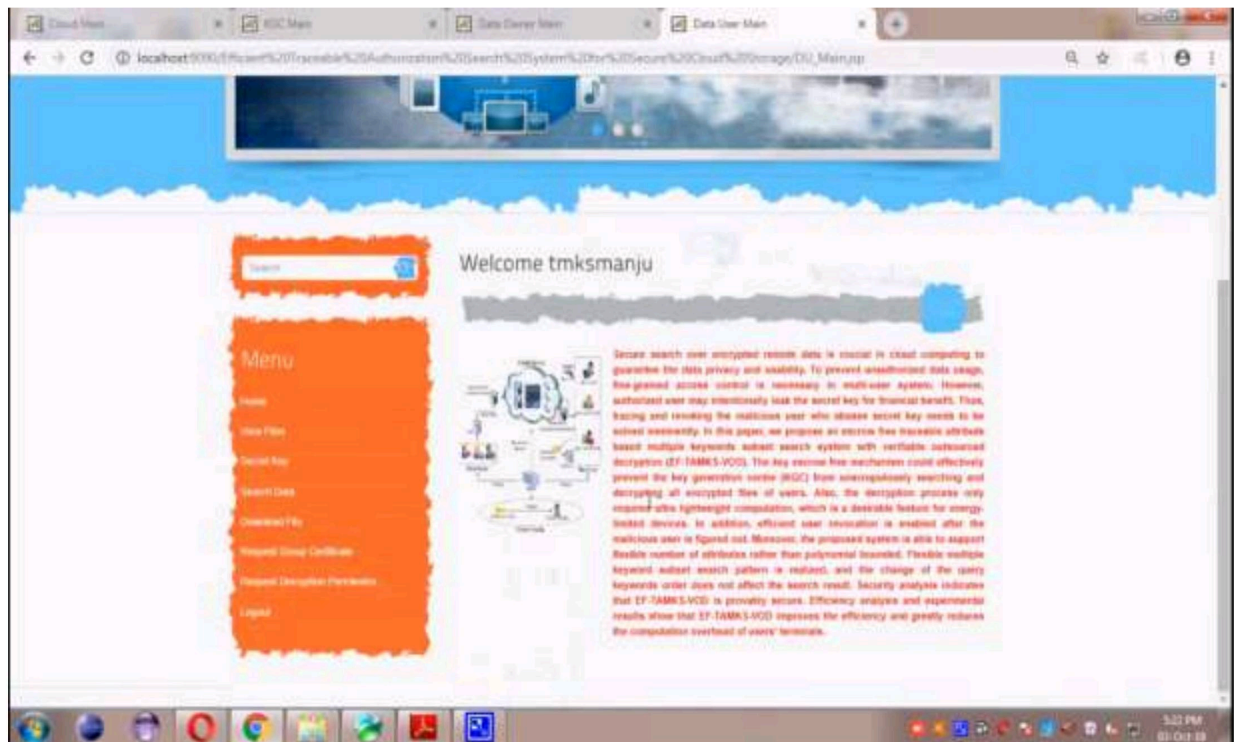

```
C:\Program Files (x86)\Apache Software Foundation\Tomcat 7.0\bin\bin\Tomcat7.exe
at org.apache.catalina.util.LifecycleBase.init(LifecycleBase.java:102)
at org.apache.catalina.startup.Catalina.load(Catalina.java:638)
at org.apache.catalina.startup.Catalina.load(Catalina.java:663)
at sun.reflect.NativeMethodAccessorImpl.invoke(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(Unknown Source)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)
at java.lang.reflect.Method.invoke(Unknown Source)
at org.apache.catalina.startup.Bootstrap.load(Bootstrap.java:288)
at org.apache.catalina.startup.Bootstrap.main(Bootstrap.java:454)
Caused by: org.apache.catalina.LifecycleException: Protocol handler initialization failed
at org.apache.catalina.connector.Connector.initInternal(Connector.java:988)
at org.apache.catalina.util.LifecycleBase.init(LifecycleBase.java:102)
... 12 more
Caused by: java.net.BindException: Address already in use: JVM_Bind <null>:8080
at org.apache.tomcat.util.net.JIoEndpoint.bind(JIoEndpoint.java:413)
at org.apache.tomcat.util.net.AbstractEndpoint.init(AbstractEndpoint.java:651)
at org.apache.coyote.AbstractProtocol.init(AbstractProtocol.java:434)
at org.apache.catalina.connector.Connector.initInternal(Connector.java:978)
... 11 more
Caused by: java.net.BindException: Address already in use: JVM_Bind
at java.net.DualStackPlainSocketImpl.bind0(Native Method)
at java.net.DualStackPlainSocketImpl.socketBind(Unknown Source)
at java.net.AbstractPlainSocketImpl.bind(Unknown Source)
at java.net.PlainSocketImpl.bind(Unknown Source)
at java.net.ServerSocket.bind(Unknown Source)
at java.net.ServerSocket.<init>(Unknown Source)
at java.net.ServerSocket.<init>(Unknown Source)
at org.apache.tomcat.util.net.DefaultServerSocketFactory.createSocket(DefaultServerSocketFactory.java:49)
at org.apache.tomcat.util.net.JIoEndpoint.bind(JIoEndpoint.java:408)
... 10 more

Apr 09, 2024 11:17:08 PM org.apache.catalina.startup.Catalina load
INFO: Initialization processed in 2157 ms
Apr 09, 2024 11:17:08 PM org.apache.catalina.core.StandardService startInternal
INFO: Starting service Catalina
Apr 09, 2024 11:17:08 PM org.apache.catalina.core.StandardEngine startInternal
INFO: Starting Servlet Engine: Apache Tomcat/7.0.64
Apr 09, 2024 11:17:08 PM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory C:\Program Files (x86)\Apache Software Foundation\Tomcat 7.0\bin\webapps\docs
Apr 09, 2024 11:17:08 PM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deployment of web application directory C:\Program Files (x86)\Apache Software Foundation\Tomcat 7.0\bin\webapps\docs has finished in 687 ms
Apr 09, 2024 11:17:08 PM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory C:\Program Files (x86)\Apache Software Foundation\Tomcat 7.0\bin\webapps\examples
```

```
C:\Program Files (x86)\Apache Software Foundation\Tomcat 7.0\bin\bin\Tomcat7.exe
at org.apache.catalina.util.LifecycleBase.init(LifecycleBase.java:102)
at org.apache.catalina.startup.Catalina.load(Catalina.java:638)
at org.apache.catalina.startup.Catalina.load(Catalina.java:663)
at sun.reflect.NativeMethodAccessorImpl.invoke(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(Unknown Source)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)
at java.lang.reflect.Method.invoke(Unknown Source)
at org.apache.catalina.startup.Bootstrap.load(Bootstrap.java:288)
at org.apache.catalina.startup.Bootstrap.main(Bootstrap.java:454)
Caused by: org.apache.catalina.LifecycleException: Protocol handler initialization failed
at org.apache.catalina.connector.Connector.initInternal(Connector.java:988)
at org.apache.catalina.util.LifecycleBase.init(LifecycleBase.java:102)
... 12 more
Caused by: java.net.BindException: Address already in use: JVM_Bind <null>:8080
at org.apache.tomcat.util.net.JIoEndpoint.bind(JIoEndpoint.java:413)
at org.apache.tomcat.util.net.AbstractEndpoint.init(AbstractEndpoint.java:651)
at org.apache.coyote.AbstractProtocol.init(AbstractProtocol.java:434)
at org.apache.catalina.connector.Connector.initInternal(Connector.java:978)
... 11 more
Caused by: java.net.BindException: Address already in use: JVM_Bind
at java.net.DualStackPlainSocketImpl.bind0(Native Method)
at java.net.DualStackPlainSocketImpl.socketBind(Unknown Source)
at java.net.AbstractPlainSocketImpl.bind(Unknown Source)
at java.net.PlainSocketImpl.bind(Unknown Source)
at java.net.ServerSocket.bind(Unknown Source)
at java.net.ServerSocket.<init>(Unknown Source)
at java.net.ServerSocket.<init>(Unknown Source)
at org.apache.tomcat.util.net.DefaultServerSocketFactory.createSocket(DefaultServerSocketFactory.java:49)
at org.apache.tomcat.util.net.JIoEndpoint.bind(JIoEndpoint.java:408)
... 10 more

Apr 09, 2024 11:17:08 PM org.apache.catalina.startup.Catalina load
INFO: Initialization processed in 2157 ms
Apr 09, 2024 11:17:08 PM org.apache.catalina.core.StandardService startInternal
INFO: Starting service Catalina
Apr 09, 2024 11:17:08 PM org.apache.catalina.core.StandardEngine startInternal
INFO: Starting Servlet Engine: Apache Tomcat/7.0.64
Apr 09, 2024 11:17:08 PM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory C:\Program Files (x86)\Apache Software Foundation\Tomcat 7.0\bin\webapps\docs
Apr 09, 2024 11:17:08 PM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deployment of web application directory C:\Program Files (x86)\Apache Software Foundation\Tomcat 7.0\bin\webapps\docs has finished in 687 ms
Apr 09, 2024 11:17:08 PM org.apache.catalina.startup.HostConfig deployDirectory
INFO: Deploying web application directory C:\Program Files (x86)\Apache Software Foundation\Tomcat 7.0\bin\webapps\examples
```



C. RESEARCH PAPER

Effective Detectable Approval Quest Framework FOR SECURE Distributed storage

E.Srividha

*Department of Computer
Science and Engineering,
Sathyabama Institute of
Science and Technology
Chennai, India
srividhya.e@gmail.com*

B.Dharani Kumar Reddy

*Department of Computer Science
and Engineering, Sathyabama
Institute of
Science and Technology
Chennai,
India
bdharanikumarreddy08@gmail.com*

B.Sandeep

*Department of Computer
Science and Engineering,
Sathyabama Institute of
Science and Technology
Chennai, India
sandheepboddu4321@gmail.com*

Abstract - Secure pursuit over scrambled far off information is pivotal in distributed computing to ensure the information protection and convenience. To forestall unapproved information utilization, fine-grained admittance control is fundamental in multi-client framework. In any case, approved client may purposefully release the mystery key for monetary advantage. Subsequently, following and disavowing the malevolent client who manhandles secret key should be addressed unavoidably. In this paper, we propose an escrow free detectable characteristic based numerous watchwords subset search framework with undeniable reevaluated decoding (EF-TAMKS-VOD). The key escrow free mechanism could effectively prevent the key generation centre (KGC) from unscrupulously searching and decrypting all encrypted files of users. Also, the decryption process only requires ultra lightweight computation, which is a desirable feature for energy-limited devices. In addition, efficient user revocation is enabled after the malicious user is figured out. Moreover, the proposed system is able to support flexible number of attributes rather than polynomial bounded. Flexible multiple keyword subset search pattern is realized, and the change of the query keywords order does not affect the

search result. Security analysis indicates that EF-TAMKS-VOD is provably secure. Efficiency analysis and experimental results show that EF-TAMKS-VOD improves the efficiency and greatly reduces the computation overhead of users' terminals.

Key Words: Secure, files, user, system, efficient

1. INTRODUCTION With the development of new computing paradigm, cloud computing becomes the most notable one, which provides convenient, on-demand services from a shared pool of configurable computing resources. Therefore, an increasing number of companies and individuals prefer to outsource their data storage to cloud server. Despite the tremendous economic and technical advantages, unpredictable security and privacy concerns become the most prominent problem that hinders the widespread adoption of data storage in public cloud infrastructure. Encryption is a fundamental method to protect data privacy in remote storage. However, how to effectively execute keyword search for plaintext becomes difficult for encrypted data due to the unreadability of ciphertext. Searchable encryption provides mechanism to enable keyword search over encrypted data. For the file sharing

system, such as multi-owner multiuser scenario, fine-grained search authorization is a desirable function for the data owners to share their private data with other authorized user. However, most of the available systems require the user to perform a large amount of complex bilinear pairing operations. These overwhelmed computations become a heavy burden for user's terminal, which is especially serious for energy constrained devices. The outsourced decryption method allows user to recover the message with ultra-lightweight decryption. However, the cloud server might return wrong halfdecrypted information as a result of malicious attack or system malfunction. Thus, it is an important issue to guarantee the correctness of outsourced decryption in public key encryption with keyword search (PEKS) system. The authorized entities may illegally leak their secret key to a third party for profits. Suppose that a patient someday suddenly finds out that a secret key corresponding his electronic medical data is sold on e-Bay. Such despicable behavior seriously threatens the patient's data privacy. Even worse, if the private electronic health data that contain serious health disease is abused by the insurance company or the patient's employment corporation, the patient would be declined to renew the medical insurance or labor contracts. The intentional secret key leakage seriously undermines the foundation of authorized access control and data privacy protection. Thus, it is extremely urgent to identify the malicious user or even prove it in a court of justice. In attribute based access control system, the secret key of user is associated with a set of attributes rather than individual's identity. As the search and decryption authority can be 2168-7161 (c) 2018

IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information. This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCC.2018.2820714, IEEE Transactions on Cloud Computing
 IEEE TRANSACTIONS ON CLOUD COMPUTING 2 shared by a set of users who own the same set of attributes, it is hard to trace the original key owner. Providing traceability to a fine-grained search authorization system is critical and not considered in previous searchable encryption systems. More importantly, in the original definition of PEKS scheme, key generation centre (KGC) generates all the secret keys in the system, which inevitably leads to the key escrow problem. That is, the KGC knows all the secret keys of the users and thus can unscrupulously search and decrypt on all encrypted files, which is a significant threat to data security and privacy. Beside, the key escrow problem brings another problem when traceability ability is realized in PEKS. If a secret key is found to be sold and the identity of secret key's owner (i.e., the traitor) is identified, the traitor may claim that the secret key is leaked by KGC. There is no technical method to distinguish who is the true traitor if the key escrow problem is not solved.

1.2 Searchable Encryption:

Searchable encryption enables keyword search over encrypted data. The concept of public key encryption with keyword search (PEKS) was proposed by Boneh et al, which is important in protecting the privacy of outsourced data. Data owners in PEKS

schemes store their files in encrypted form in the remote untrusted data server. The data users query to search on the encrypted files by generating a keyword trapdoor, and the data server executes the search operation. Waters et al. showed that PEKS schemes could be utilized to construct searchable audit logs. Later, Xu et al. presented a general framework to combine PEKS and fuzzy keyword search without concrete construction. Tang proposed a multiparty searchable encryption scheme together with a bilinear pairing based scheme. In 2016, Chen et al. introduced the concept “dualserver” into PEKS to resist off-line keyword guessing attack. Yang et al. introduced time-release and proxy reencryption method to PEKS scheme in order to realize time controlled authority delegation. Wang et al. proposed a ranked keyword search scheme for searchable symmetric encryption, in which the order-preserving symmetric encryption is utilized. Cao et al. designed a novel system to realize multiple keyword ranked search. Searchable encryption is also further studied

1.3 ABE:

ABE is an important method to realize fine-grained data sharing. In ABE schemes, descriptive attributes and access policies are associated with attribute secret keys and ciphertexts. A certain secret key can decrypt a cipher text if and only if the associated attributes and the access policy match each other. The notion of ABE was proposed by Sahai et al. in 2005. According to whether the access control policy associates with the ciphertext or the secret key, ABE schemes can be classified into ciphertext-policy ABE (CP-ABE) and key-policy ABE (KP-ABE). Since the Sahai’s seminal work, ABE based access control becomes a research

focus. Considering the challenges in expressing access control policy, ABE scheme with non-monotonic access structure is proposed. ABE systems with constant size cipher text are constructed to reduce the storage overhead. In order to accelerate the decryption, researchers make effort to speed up the decryption algorithm. Decentralized ABE is investigated in which multiple authorities work independently without collaboration.

1.4 Traitor Tracing: Traitor tracing was introduced by Chor et al. to help content distributors identifying pirates. In the digital content distribution system, there is no way to prevent a legitimate user to give (or sell) his decryption key to the others. Traitor tracing mechanism helps the distributor to find out the misbehaved user by running “tracing” algorithm so that he could take legal action against the owner of the leaked secret key. Later, traitor tracing mechanism is introduced to broadcast encryption, where a sender is able to generate ciphertext and only the users in the designated receiver set can decrypt. The traceability function enables the broadcaster to identify the traitor, and prevents the authorized users from leaking their keys. The approach is to give each user a distinct set of keys, which is deemed as “watermark” for tracing. Traceability is further investigated for broadcast encryption in. In CP-ABE scheme, secret keys are not defined over identities. Instead, they are associated with a set of attributes. Multiple users may share the same set of attributes. This brings convenience to expressive access control. However, given a leaked secret key, it is impossible to figure out the original key owner in traditional ABE system. It means that the malicious user, who sells his secret key, almost has little risk of being identified. The traceability problem in

CP-ABE is studied.

2. PROPOSED SYSTEM:

1. Flexible System Extension:

We propose an efficient system which supports flexible system extension, which accommodates flexible number of attributes. The attributes are not fixed in the system initialization phase and the size of attribute set is not restricted to polynomially bound, so that new attribute can be added to the system at any time.

2. Moreover, the size of public parameter does not grow with the number of attributes. No matter how many attributes are supported in the system, no additional communication nor storage costs is brought to the system. This feature is desirable for the cloud system for its ever increasing user volume.

3. **Efficient Verifiable Decryption.** We propose a system which adopts the outsourced decryption mechanism to realize efficient decryption. Most of the decryption computation are outsourced to the cloud server, and the data user is able to complete the final decryption with an ultra lightweight computation. Moreover, the correctness of the cloud server's partial decryption computation can be verified by the user.

4. **Efficient User Revocation.** Once a user is identified as traitor through tracing algorithm, system revokes this malicious user from the authorized group. Compared with the existing scheme this revocation mechanism has much better efficiency.

2.1 ADVANTAGES

1. **Flexible System Extension:**
2. **Efficient Verifiable Decryption.**
3. **Efficient User Revocation.**
4. **Traceability of Abused Secret Key.**

3. GOALS:

The Primary goals in the design of the UML are as Cloud server (CS):

Cloud server has tremendous storage space and powerful computing capability, which provides on-demand service to the system. Cloud server is responsible to store the data owner's encrypted files and respond on data user's search query. Data owner: Follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

3.1 USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system

can be depicted.

4. SYSTEM IMPLEMENTATION

4.1 MODULES:-

Key generation Centre (KGC).

Cloud server (CS).

Data owner.

Data user.

4.2 MODULES DESCRIPTION:-

Key generation Centre (KGC): KGC is responsible to generate the public parameter for the system and the public/secret key pairs for the users. Once the user's secret key is leaked



S.No	User Name	Attributes	Generate Key
1	sal	Delft,Engineering,Programming,Java	Generate Key
2	salil	Mumbai,Accounting,Designing,Java	Generate Key
3	SalilJony	Bangalore,Testing,Designing,Oracle	Generate Key
4	hans	Delft,Engineering,Programming,Java	Generate Key
5	slia	Bangalore,Testing,Designing,Java	Generate Key
6	hansar	Chennai,Engineering,Programming,Java	Generate Key
7	sal	Chennai,Engineering,Programming,Java	Generate Key
8	Salilgadu	Chennai,Engineering,Programming,Java	Generate Key

for profits or other purposes, KGC runstrace algorithm to find the malicious user. After the traitor is traced, KGC sends user revocation request to cloud server to revoke the user's search privilege.

Data owner

Data owner utilizes the cloud storage service to store the files. Before the data outsourcing, the data owner extracts keyword set from the file and encrypts it into secure index. The document is also encrypted to cipher text. During the encryption process, the access policy is specified and embedded into the cipher text to realize fine grained access control.

Data user:

Each data user has attribute set to

describe his characteristics, such as professor, computer Science College, dean, etc. The attribute set is embedded into user's secret key. Using the secret key, data user is able to search on the encrypted files stored in the cloud, i.e., chooses a keyword set that he wants to search. Then, the keyword is encrypted to a trapdoor using user's secret key. If the user's attribute set satisfies the access policy defined in the encrypted files, the cloud server responds on user's search query and finds the match files. Otherwise, the search query is rejected. After the match files are returned, the user runs decryption algorithm to recover the plaintext

User Activation:

Fig -1: User activation

OWNER ACTIVATION:

S.No	Owner Name	Attributes	Generate Key
1	sal	Delft,Engineering,Programming,Java	Generate Key
2	salil	Chennai,Engineering,Programming,Java	Generate Key
3	salil	Chennai,Testing,Designing,Oracle	Generate Key
4	hansar	Chennai,Engineering,Programming,Java	Generate Key
5	Salilgadu	Bangalore,Testing,Designing,Java	Generate Key
6	hans	Chennai,Engineering,Programming,Java	Generate Key
7	hansar	Chennai,Engineering,Programming,Java	Generate Key
8	sal	Chennai,Engineering,Programming,Java	Generate Key
9	Salilgadu	Chennai,Engineering,Programming,Java	Generate Key

Fig -2: Owner activation

SENDING KEYS



Fig -3: Sending keys

FILE INFORMATION:



Fig -4: File Information

MALICIOUS USER BLOCKING:



Fig -5: Malicious User Blocking

MALICIOUS USER UNBLOCKING:



Fig -6: Malicious User

OTP :



Fig -7: OTP

FILE DOWNLOAD:



Fig -8: File Download

1. CONCLUSION

The enforcement of access control and the support of keyword search are important issues in secure cloud storage system. In this work, we defined a new paradigm of searchable encryption system, and proposed a concrete construction. It supports flexible multiple keywords subset search, and solves the key escrow problem during the key generation procedure. Malicious user who sells secret key for benefit can be traced. The decryption operation is partly outsourced to cloud server and the correctness of half- decrypted result can be verified by data user. The performance analysis and simulation show its efficiency in computation and storage overhead.

Experimental results indicate that the computation overhead at user's terminal is significantly reduced, which greatly saves the energy for resource-constrained devices of users.

REFERENCES

- [1] C. Wang, N. Cao, J. Li, K. Ren, W. Lou. "Secure ranked keyword search over encrypted cloud data"[C]//IEEE 30th International Conference on Distributed Computing Systems (ICDCS), IEEE, 2010: 253-262.
- [2] C. Wang, N. Cao, J. Li, K. Ren, W. Lou. "Secure ranked keyword search over encrypted cloud data"[C]//IEEE 30th International Conference on Distributed Computing Systems (ICDCS), IEEE, 2010: 253-262.
- [3] Q. Zhang, L. T. Yang, Z. Chen, P. Li, M. J. Deen. "Privacy- preserving Double-Projection Deep Computation Model with Crowd sourcing on Cloud for Big Data Feature Learning," IEEE Internet of Things Journal, 2017, DOI: 10.1109/JIOT.2017.2732735.
- [4] R. Chen, Y. Mu, G. Yang, F. Guo and X. Wang, "Dual-Server Public-Key Encryption with Keyword Search for Secure Cloud Storage," IEEE Transactions on Information Forensics and Security, 2016, vol.11, no. 4, 789-798.
- [5] X. Liu, R.H. Deng, K.K.R. Choo, J. Weng. "An efficient privacy preserving outsourced calculation toolkit with multiple keys." IEEE Transactions on Information Forensics and Security 11 (2016): 2401-2414.
- [6] B. R. Waters, D. Balfanz, G. Durfee, and D. K. Smetters, "Building an encrypted and searchable audit log," in NDSS, 2004.
- [7] Y. Yang, X. Liu, R.H. Deng, "Multi-user Multi-Keyword Rank Search over Encrypted Data in Arbitrary Language". IEEE Transactions on Dependable and Secure Computing, 2018, publish online, DOI: 10.1109/TDSC.2017.2787588.
- [8] W. Sun, S. Yu, W. Lou, Y. Hou and H. Li, "Protecting Your Right: Verifiable Attribute-based Keyword Search with Finegrained Owner-enforced Search Authorization in the Cloud," IEEE Transactions on Parallel and Distributed Systems, 2016, vol. 27, no.4, pp. 1187-1198.
- [9] K. Liang, W. Susilo, "Searchable Attribute-Based Mechanism with Efficient Data Sharing for Secure Cloud Storage," IEEE Transactions on Information Forensics and Security, 2015, vol. 10, no. 9, pp. 1981-1992.
- [10] M. Green, S. Hohenberger, and B.Waters, "Outsourcing the decryption of ABE ciphertexts," in USENIX Security Symposium, ACM, 2011, pp. 34-34.
- [11] J. Lai, R. H. Deng, C. Guan, and J. Weng, "Attribute-based encryption with verifiable outsourced decryption," IEEE Transactions on Information Forensics and Security, 2013, vol. 8, no. 8, pp. 1343-1354.
- [12] B. Qin, R. H. Deng, S. Liu, and S. Ma, "Attribute-Based Encryption with Efficient Verifiable Outsourced Decryption," IEEE Transactions on Information Forensics and Security, 2015, vol. 10, no. 7, pp. 1384-1394.
- [13] C. Wang, N. Cao, J. Li, K. Ren, W. Lou. "Secure ranked keyword search over encrypted cloud data"[C]//IEEE 30th International Conference on Distributed Computing Systems (ICDCS), IEEE, 2010: 253-262.

- [14] Q. Zhang, L. T. Yang, Z. Chen, P. Li, M. J. Deen. "Privacy-preserving Double-Projection Deep Computation Model with Crowd sourcing on Cloud for Big Data Feature Learning," IEEE Internet of Things Journal, 2017, DOI: 10.1109/JIOT.2017.2732735.
- [15] R. Chen, Y. Mu, G. Yang, F. Guo and X. Wang, "Dual-Server Public- Key Encryption with Keyword Search for Secure Cloud Storage," IEEE Transactions on Information Forensics and Security, 2016, vol.11, no. 4, 789-798.
- [16] X. Liu, R.H. Deng, K.K.R. Choo, J. Weng. "An efficient privacy preserving outsourced calculation toolkit with multiple keys." IEEE Transactions on Information Forensics and Security 11 (2016): 2401-2414.
- [17] B. R. Waters, D. Balfanz, G. Durfee, and D. K. Smetters, "Building an encrypted and searchable audit log," in NDSS, 2004.
- [18] Y. Yang, X. Liu, R.H. Deng, "Multi-user Multi-Keyword Rank Search over Encrypted Data in Arbitrary Language". IEEE Transactions on Dependable and Secure Computing, 2018, publish online, DOI: 10.1109/TDSC.2017.2787588.
- [19] W. Sun, S. Yu, W. Lou, Y. Hou and H. Li, "Protecting Your Right: Verifiable Attribute-based Keyword Search with Finegrained Owner-enforced Search Authorization in the Cloud," IEEE Transactions on Parallel and Distributed Systems, 2016, vol. 27, no.4, pp. 1187-1198.
- [20] K. Liang, W. Susilo, "Searchable Attribute-Based Mechanism with Efficient Data Sharing for Secure Cloud Storage," IEEE Transactions on Information Forensics and Security, 2015, vol. 10, no. 9, pp. 1981-1992.
- [21] M. Green, S. Hohenberger, and B. Waters, "Outsourcing the decryption of ABE ciphertexts," in USENIX Security Symposium, ACM, 2011, pp. 34-34.
- [22] J. Lai, R. H. Deng, C. Guan, and J. Weng, "Attribute-based encryption with verifiable outsourced decryption," IEEE Transactions on Information Forensics and Security, 2013, vol. 8, no. 8, pp. 1343-1354.
- [23] B. Qin, R. H. Deng, S. Liu, and S. Ma, "Attribute-Based Encryption with Efficient Verifiable Outsourced Decryption," IEEE Transactions on Information Forensics and Security, 2015, vol. 10, no. 7, pp. 1384-1394.
- [24] C. Wang, N. Cao, J. Li, K. Ren, W. Lou. "Secure ranked keyword search over encrypted cloud data"[C]//IEEE 30th International Conference on Distributed Computing Systems (ICDCS), IEEE, 2010: 253-262.
- [25] Q. Zhang, L. T. Yang, Z. Chen, P. Li, M. J. Deen. "Privacy-preserving Double-Projection Deep Computation Model with Crowd sourcing on Cloud for Big Data Feature Learning," IEEE Internet of Things Journal, 2017, DOI: 10.1109/JIOT.2017.2732735.
- [26] R. Chen, Y. Mu, G. Yang, F. Guo and X. Wang, "Dual-Server Public- Key Encryption with Keyword Search for Secure Cloud Storage," IEEE Transactions on Information Forensics and Security, 2016, vol.11, no. 4, 789-798.
- [27] X. Liu, R.H. Deng, K.K.R. Choo, J. Weng. "An efficient privacy preserving outsourced calculation

toolkit with multiple keys.” IEEE Transactions on Information Forensics and Security 11 (2016): 2401-2414.

[28] B. R. Waters, D. Balfanz, G. Durfee, and D. K. Smetters, “Building an encrypted and searchable audit log,” in NDSS, 2004.

[29] Y. Yang, X. Liu, R.H. Deng, “Multi-user Multi-Keyword Rank Search over Encrypted Data in Arbitrary Language”. IEEE Transactions on Dependable and Secure Computing, 2018, publish online, DOI: 10.1109/TDSC.2017.2787588.

[30] W. Sun, S. Yu, W. Lou, Y. Hou and H. Li, “Protecting Your Right: Verifiable Attribute-based Keyword Search with Finegrained Owner-enforced Search Authorization in the Cloud,” IEEE Transactions on Parallel and Distributed Systems, 2016, vol. 27, no.4, pp. 1187-1198.

D. CERTIFICATE

