SECURE TECHNIQUES FOR KEYWORD-BASED SEARCHAND DATA SHARING IN CLOUD COMPUTING

(Cloud Computing)

Project report submitted
in partial fulfillment of the requirement for the award of the degree of

BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCEANDENGINEERING

By

Sai Vignesh B **(U19CN362)**

Karthikeyan K **(U19CS467)**

Rohith G **(U19CN345)**

Kasam Shree Veera Hanuman Reddy **(U19CS469)**

Under the guidance of

Dr K.Upendra Babu

ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING

BHARATH INSTITUTE OF HIGHER EDUCATION AND RESEARCH
(Deemed to be University Estd u/s 3 of UGC Act, 1956)
CHENNAI 600 073, TAMILNADU, INDIA April 2023

Bharath
INSTITUTE OF HIGHER EDUCATION AND RESEARCH
(Declared as Deemed - to - be - University under section 3 of UGC Act 1956)
BHARATH INSTITUTE OF SCIENCE AND TECHNOLOGY
No.173, Agharam Road, Selaiyur, Chennai - 600 073.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

BONAFIDE CERTIFICATE

This is to Certify that this Project Report Titled "**SECURED TECHNIQUES FOR KEYWORD-BASED SEARCH AND DATA SHARING IN CLOUD COMPUTING"** is the Bonafide Work of **B. Sai Vignesh (U19CN362), K. Karthikeyan (U19CS467), G. Rohith (U19CN345), Kasam Shree Veera Hanuman Reddy (U19CS469)** of Final Year B.Tech. (CSE) who carried out the major project work under my supervision Certified further, that to the best of my knowledge the work reported here in does not form part of any other project report or dissertation on basis of which a degree or award conferred on a nearlier occasion by any other candidate.

PROJECT GUIDE	**HEAD OF THE DEPARTMENT**
Dr K. Upendra Babu	**Dr. Maruthu Perumal**
Assistant Professor	**Professor**
Department of CSE	**Department of CSE**
BIHER	**BIHER**

Submitted for the Project Viva-Voce held on ……

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION

We declare that this project report titled **Secured Techniques For Keyword-Based Search And Data Sharing In Cloud Computing** submitted in partial fulfillment of the degree of **B.Tech in (Computer Science and Engineering)** is a record of original work carried out by us under the supervision of **Dr. K. Upendra Babu**, and has not formed the basis for the award of any other degree or diploma, in this or any other Institution or University. In keeping with the ethical practice of reporting scientific information, due acknowledgements have been made wherever the findings of others have been cited.

Sai Vignesh B
(U19CN362)

Karthikeyan K
(U19CS467)

Rohith G
(U19CN345)

Kasam Shree Veera Hanuman Reddy
(U19CS469)

Chennai

ACKNOWLEDGMENTS

First, we wish to thank the almighty who gave us good health and success throughout our project work.

We express our deepest gratitude to our beloved President **Dr J. Sundeep Aanand,** and Managing Director **Dr.E. Swetha Sundeep Aanand** for providing us with the necessary facilities for the completion of our project.

We take great pleasure in expressing sincere thanks to Vice Chancellor **Dr K.Vijaya Baskar Raju**, Pro Vice-Chancellor(Academic) **Dr M.Sundararajan**, Registrar **Dr S.Bhuminathan** and Additional Registrar **Dr R.HariPrakash** for backing us in this project. We thank our Dean of Engineering **Dr J. Hameed Hussain** for providing sufficient facilities for the completion of this project.

We express our immense gratitude to our Academic Coordinator **Mr. G. Krishna Chaitanya** for his eternal support in completing this project.

We thank our Dean, School of Computing **Dr S. Neduncheliyan** for his encouragement and valuable guidance.

We record indebtedness to our Head, Department of Computer Science and Engineering **Dr.S.Maruthu Perumal** for his immense care and encouragement towards us throughout the course of this project.

We also take this opportunity to express a deep sense of gratitude to our Supervisor **Dr. K.Upendra Babu** and our Project Co-Ordinator **Dr. P. Rama** for their cordial support, valuable information and guidance, they helped us in completing this project through various stages.

We thank our department faculty, supporting staff and friends for their help and guidance to complete this project.

B. SAI VIGNESH (U19CN362)
K. KARTHIKEYAN (U19CS467)
G. ROHITH (U19CN345)
KASAM SHREE VEERA HANUMAN REDDY (U19CS469)

ABSTRACT

The untrustworthiness of cloud servers and the data privacy of users it is necessary to encrypt the data before outsourcing the cloud Aiming to realize secure keyword search over encrypted data against malicious users and malicious cloud service providers we find a compromised method by into the block chain into SSE the cloud storage used in searchable symmetric encryption schemes (SSE) is provided in a private way, which cannot be seenas a true cloud. Moreover, the cloud server is thought to be credible WE begin by pointing out the importance of storing the data in a public chain We introduce a system that leverages block chain technology to provide a secure distributed data storage with keyword search service The System allows the client to upload their in encrypted form distributes the data content to cloud nodes and ensure data availability using cryptographic techniques we introduce a system that leverages block chain technology to provide a secure distributed data storage with keyword search service. TKSE realizes server- side verifiability, protecting honest cloud servers from being framed by malicious data owners in the data storage phase. Furthermore, block chain technologies and hash functions are used to enable payment fairness ofsearch fees without introducing any third party even if the user or the cloud is malicious. Our security analysis and performance evaluation indicate that TKSE is secure and efficient and it is suitable for cloud computing.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In recent years, cloud computing technologies have gotten rapid developments and a line of studies have been done on security issues in cloud computing, such as access control and privacy protection. As a typical service in cloud computing, cloud storage needs both data security and search functionality. In fact, user-side verifiability takes into consideration that the cloud server may be malicious, that is, the cloud server may only return part of search results or maliciously return incorrect results. The issue of user-side verifiability is firstly addressed in. However, these two schemes cannot support server-side verifiability and fair payment without any trusted third party. Furthermore, server-side verifiability takes into consideration that the data owner may be malicious, that is, the data owner may maliciously outsource invalid data in the data storage phase and fraudulently claim compensation later. This concern has not been addressed and even has received little attention in the literature. Last but not least, most of the previous schemes are bank-dependent. Specifically, either the payment issue is not considered or the default traditional payment mechanism is exploited in which a trusted third party (TTP) such as a trustworthy bank has to be introduced for payment fairness. Payment fairness can promote the honest behaviors of users and cloud servers [7]. If a malicious behavior is detected based on the user-side verifiability(resp. server-side verifiability),the data owner(resp. cloud server)should get adequate compensation from the cloud server (resp. data owner) no matter what the cloud server(resp. data owner)does. Therefore, fair payment without any third party is a meaningful and challenging task and it remains in SSE.

In order to thoroughly address the aforementioned challenging issues in cloud computing, we propose TKSE, a Trustworthy Keyword Search scheme over Encrypted data without needing any third party. TKSE is proven secure and our performance evaluation shows its efficiency. In particular, TKSE is characterized by the following desirable features.

- Keyword Search over Encrypted Data. The encrypted data index based on the Elliptic Curve Digital Signature Algorithm(ECDSA)allows a user to search over the outsourced encrypted data.
- User-side Verifiability. In TKSE, a data owner can embed search requirements into the Output script of a joint transaction such that the transaction can be redeemed by the cloud server if and only if the output script evaluates to true based on the returned search result. Therefore, TKSE enables the data owner to resist malicious cloud servers and user-side verifiability is realized.
- Server-side Verifiability. Similar to user-side verifiability, the public verification of digital sign at ur enables cloud servers to check the validness of the outsourced encrypted data from the data owner in the data storage phase. Thus, malicious data owners can be detected by the cloud server, which realizes server-side verifiability.
- Fair Payment and No TTP. Based on hash functions and ECDSA, TKSE is compatible with block chains such as the Bit coin block chain and the Ethereum block chain. The global consensus and distributed nature of a block chain enable a fair payment mechanism in TKSE without introducing any TTP.

CHAPTER 2

2.1 LITERATURE SURVEY:

Literature survey is the most important step in software development process. Before developing the tool it is necessary to determine the time factor, economy and company strength. Once these things are satisfied, then the next step is to determine which operating system and language can be used for developing the tool. Once the programmers start building the tool the programmers need lot of external support. This support can be obtained from senior programmers, from book or from websites. Before building the system the above consideration are taken into account for developing the proposed system.

The major part of the project development sector considers and fully survey all the required needs for developing the project. For every project Literature survey is the most important sector in software development process. Before developing the tools and the associated designing it is necessary to determine and survey the time factor, resource requirement, man power, economy, and company strength. Once these things are satisfied and fully surveyed, then the next step is to determine about the software specifications in the respective system such as what type of operating system the project would require, and what are all the necessary software are needed to proceed with the next step such as developing the tools, and the associated operations.

[1] Identity-based encryption with outsourced revocation in cloud computing:
Authors: J. Li, J. Li, X. Chen, C. Jia, and W. Lou

Identity-Based Encryption (IBE) which simplifies the public key and certificate management at Public Key Infrastructure (PKI) is an important alternative to public key encryption. However, one of the main efficiency drawbacks of IBE is the overhead computation at Private Key Generator (PKG) during user revocation. Efficient revocation has been well studied in traditional PKI setting, but the cumbersome management of certificates is precisely the burden that IBE strives to alleviate. In this paper, aiming at tackling the critical issue of identity revocation, we introduce

outsourcing computation into IBE for the first time and propose a revocable IBE scheme in the server-aided setting. Our scheme offloads most of the key generation related operations during key-issuing and key-update processes to a Key Update Cloud Service Provider, leaving only a constant number of simple operations for PKG and users to perform locally. This goal is achieved by utilizing a novel collusion-resistant technique: we employ a hybrid private key for each user, in which an AND gate is involved to connect and bound the identity component and the time component. Furthermore, we propose another construction which is provable secure under the recently formulized Refereed Delegation of Computation model. Finally, we provide extensive experimental results to demonstrate the efficiency of our proposed construction.

[2] New publicly verifiable databases with efficient updates:

Authors: X. Chen, J. Li, X. Huang, J. Ma, and W. Lou

The notion of verifiable database (VDB) enables a resource-constrained client to securely outsource a very large database to an un-trusted server so that it could later retrieve a database record and update it by assigning a new value. Also, any attempt by the server to tamper with the data will be detected by the client. Very recently, Catalano and Fiore proposed an elegant framework to build efficient VDB that supports public verifiability from a new primitive named vector commitment. In this paper, we point out Catalano-Fiore's VDB framework from vector commitment is vulnerable to the so-called forward automatic update (FAU) attack. Besides, we propose a new VDB framework from vector commitment based on the idea of commitment binding. The construction is not only public verifiable but also secure under the FAU attack. Furthermore, we prove that our construction can achieve the desired security properties.

[3] A searchable symmetric encryption scheme using block chain:

Authors: H. Li, F. Zhang, J. He, and H. Tian

At present, the cloud storage used in searchable symmetric encryption schemes (SSE) is provided in a private way, which cannot be seen as a true cloud. Moreover, the cloud server is thought to be credible, because it always returns the search result to the user, even they are notcorrect. In order to really resist this malicious

adversary and accelerate the usage of the data, it isnecessary to store the data on a public chain, which can be seen as a decentralized system. As theincreasing amount of the data, the search problem becomes more and more intractable, because there does not exist any effective solution at present.

In this paper, we begin by pointing out the importance of storing the data in a public chain. We then innovatively construct a model of SSE using block chain(SSE-using-BC) and give its security definition to ensure the privacy of the data and improve the search efficiency. Accordingto the size of data, we consider two different cases and propose two corresponding schemes. Lastly, the security and performance analyses show that our scheme is feasible and secure.

[4] Block chain Based system for secure data storage with private key word search:
Authors: H. G. Do and W. K. Ng

Traditional cloud storage has relied almost exclusively on large storage providers, who act as trusted third parties to transfer and store data. This model poses a number of issues including data availability, high operational cost, and data security. In this paper, we introduce a system that leverages block chain technology to provide a secure distributed data storage with keyword search service. The system allows the client to upload their data in encrypted form, distributes the data content to cloud nodes and ensures data availability using cryptographic techniques. It also provides the data owner a capability to grant permission for others to search on her data. Finally, the system supports private keyword search over the encrypted dataset.

[5] Secure attribute-based signature scheme with multiple authorities for block chain in electronic health records systems:
Authors: R. Guo, H. Shi, Q. Zhao, and D. Zheng

Electronic Health Records (EHRs) are entirely controlled by hospitals instead of patients, which complicates seeking medical advices from different hospitals. Patients face a critical need to focus on the details of their own healthcare and restore management of their own medical data. The rapid development of block chain technology promotes population healthcare, including medical records as well as

patient-related data. This technology provides patients with comprehensive, immutable records, and access to EHRs free from service providers and treatment websites. In this paper, to guarantee the validity of EHRs encapsulated in block chain, we present an attribute-based signature scheme with multiple authorities, in which a patient endorses a message according to the attribute while disclosing no information other than the evidence that he has attested to it. Furthermore, there are multiple authorities without a trusted single or central one to generate and distribute public/private keys of the patient, which avoid the escrow problem and conforms to the mode of distributed data storage in the block chain. By sharing the secret pseudorandom function seeds among authorities, this protocol resists collusion attack out of N from N -1 corrupted authorities. Under the assumption of the computational bilinear Diffie-Hellman, we also formally demonstrate that, in terms of the unforgeability and perfect privacy of the attribute-signer, this attribute-based signature scheme is secure in the random oracle model. The comparison shows the efficiency and properties between the proposed method and methods proposed in other studies.

[6] Charm: A framework for rapidly prototyping cryptosystems:

Authors: J. A. Akinyele, C. Garman, I. Miers, M. W. Pagano, M. Rushanan, M. Green, and A. D. Rubin

"Charm: A Framework for Rapidly Prototyping Cryptosystems" is a research paper that proposes a framework for rapidly prototyping and implementing new cryptosystems. The proposed framework, called Charm, is designed to make it easier for researchers and developers to create and test new cryptographic algorithms and protocols.

The Charm framework provides a set of high-level programming abstractions and tools for building cryptosystems, including encryption, digital signatures, and secure multiparty computation. The framework is implemented in Python, a widely-used programming language that is easy to learn and use.

One of the key features of the Charm framework is its modularity. The framework is organized into a set of independent modules, each of which implements a different cryptographic primitive or protocol. This modularity makes it easy for developers to mix and match different modules to create new cryptosystems.

Another key feature of the Charm framework is its support for multiple cryptographic backends. The framework can be configured to use different underlying cryptographic libraries, including OpenSSL and NaCl, depending on the requirements of the application.

The Charm framework also provides a set of tools for testing and benchmarking cryptosystems, including a test suite and performance profiling tools. This makes it easier for developers to evaluate the security and performance of their cryptosystems.

In conclusion, "Charm: A Framework for Rapidly Prototyping Cryptosystems" is a research paper that proposes a framework for rapidly prototyping and implementing new cryptosystems. The Charm framework provides a set of high-level programming abstractions and tools for building cryptosystems, and is designed to be modular and flexible. The framework also provides tools for testing and benchmarking cryptosystems, making it easier for developers to evaluate the security and performance of their algorithms and protocols.

[7] Homomorphic MACs: MAC-based integrity for network coding :

Authors: S. Agrawal and D. Boneh

"Homomorphic MACs: MAC-based Integrity for Network Coding" is a research paper that proposes a new approach to providing message authentication and integrity protection for network coding. Network coding is a technique used in data communication that allows multiple data streams to be combined and transmitted over a single channel.

The proposed approach uses homomorphic message authentication codes (MACs) to

provide integrity protection for network-coded messages. Homomorphic MACs combined into a single MAC that can be used to authenticate a network-coded message.are a type of MAC that can be combined algebraically, allowing multiple MACs to be

The paper presents several constructions for homomorphic MACs, including one based on bilinear pairings and another based on error-correcting codes. The authors also provide a security analysis of the proposed constructions, demonstrating their resistance to various types of attacks.

One of the advantages of the proposed approach is its efficiency. Homomorphic MACs can be computed efficiently using existing cryptographic primitives, and the algebraic properties of the MACs allow them to be combined efficiently. This makes it possible to provide integrity protection for network-coded messages with relatively low overhead.

Another advantage of the proposed approach is its flexibility. Homomorphic MACs can be used with different types of network coding schemes, including random linear network coding and network coding with intermediate nodes. This makes the approach applicable to a wide range of communication scenarios.

In conclusion, "Homomorphic MACs: MAC-based Integrity for Network Coding" is a research paper that proposes a new approach to providing message authentication and integrity protection for network coding using homomorphic MACs. The proposed approach is efficient and flexible, and can be used with different types of network coding schemes.

[8] Provable data possession at untrusted stores:

Authors: G. Ateniese, R. Burns, R. Curtmola, Joseph Herring, L. Kissner, Z. Peterson, and D. Song

"Provable Data Possession at Untrusted Stores" is a research paper that proposes a method for verifying the integrity of data stored remotely on an untrusted server, without requiring the server to disclose its entire data content.

The proposed method, called Provable Data Possession (PDP), enables a client to verify that a server is in possession of all the data it claims to have, and that the data has not been tampered with or deleted. PDP uses a set of cryptographic techniques to ensure that the server cannot cheat the client by hiding or modifying the data.

To implement PDP, the data owner first divides the data into small blocks, computes a set of short message digests for each block, and stores the digests on the remote server. To check the integrity of the data, the client generates a random challenge, sends it to the server, and requests a proof of possession for a specific block of data. The server then computes a response using a subset of the stored digests and sends it back to the client. The client can verify the response to ensure that the server has not tampered with the data.

The paper provides a formal analysis of the security of PDP and shows that it is resistant to various types of attacks. The authors also propose several extensions to PDP, including a dynamic version that supports efficient updates to the data stored on the server.

One of the advantages of PDP is its simplicity and efficiency. PDP requires minimal communication overhead and computational resources, making it well-suited for applications with limited resources, such as mobile devices or sensor networks. PDP can also be combined with other cryptographic techniques, such as encryption or access control, to provide a more comprehensive solution for data privacy and security.

In conclusion, "Provable Data Possession at Untrusted Stores" is a research paper that proposes a method for verifying the integrity of data stored remotely on an untrusted server, using cryptographic techniques. The proposed method, PDP, is simple, efficient, and resistant to various types of attacks, and can be extended to support dynamic updates to the data stored on the server.

[9] Proofs of storage from homomorphic identification protocols:

Authors: G. Ateniese, S. Kamara, and J. Katz

"Proofs of Storage from Homomorphic Identification Protocols" is a research paper that proposes a method for verifying the integrity of data stored remotely on an untrusted server, using homomorphic identification protocols.

The paper introduces a new type of cryptographic protocol called a homomorphic identification protocol (HIP), which enables a client to verify that a server is in possession of a specific piece of data, without revealing the data itself. The paper shows how HIPs can be used to construct a Proof of Storage (PoS) protocol, which enables a client to verify that a server has stored data correctly and has not tampered with it.

To implement the PoS protocol, the client first generates a set of challenges and sends them to the server. The server responds with a set of encrypted values, which are generated by applying a homomorphic function to the challenges and the data. The client then verifies the encrypted values to ensure that they correspond to the correct challenges and that the data has not been tampered with.

The paper provides a formal analysis of the security of the PoS protocol and shows that it is resistant to various types of attacks. The authors also propose several extensions to the PoS protocol, including a dynamic version that supports efficient updates to the data stored on the server.

One of the advantages of the PoS protocol is its efficiency and scalability. The PoS protocol requires minimal communication overhead and computational resources,

making it well-suited for large-scale data storage applications. The PoS protocol can also be combined with other cryptographic techniques, such as encryption or access control, to provide a more comprehensive solution for data privacy and security.

In conclusion, "Proofs of Storage from Homomorphic Identification Protocols" is a research paper that proposes a method for verifying the integrity of data stored remotely on an untrusted server, using homomorphic identification protocols. The proposed method, PoS, is efficient, scalable, and resistant to various types of attacks, and can be extended to support dynamic updates to the data stored on the server.

[10] Provable multicopy dynamic data possession in cloud computing systems:
Authors: A. F. Barsoum and M. A. Hasan

"Provable Multicopy Dynamic Data Possession in Cloud Computing Systems" is a research paper by Ateniese et al. that proposes a new technique for ensuring the integrity and availability of data stored in cloud computing systems

.

The authors recognize that data storage in cloud computing systems poses several security challenges, including the risk of data loss or corruption due to hardware failures or malicious attacks. To address these challenges, the authors propose a new technique called provable multicopy dynamic data possession (P-MDDP), which enables cloud users to verify the integrity and availability of their data across multiple copies of the data.

The P-MDDP technique uses a combination of homomorphic verifiable tags and distributed hash tables to ensure that the data is stored securely and can be accessed and verified by authorized users. The technique also provides support for dynamic data updates and ensures that all copies of the data are kept in sync with each other.

The authors evaluate the performance and effectiveness of the P-MDDP technique through simulations and experiments using real-world datasets. The results show that the technique provides high levels of data integrity and availability, with low overheads and

efficient verification times.

The P-MDDP technique has several advantages over existing techniques for ensuring data integrity and availability in cloud computing systems. Firstly, it provides support for multiple copies of the data, ensuring that the data is available even in the event of failures or attacks. Secondly, it uses homomorphic verifiable tags to provide efficient and secure verification of data integrity. Finally, it supports dynamic data updates, ensuring that all copies of the data are kept in sync with each other.

In conclusion, Ateniese et al.'s paper proposes a new technique called provable multicopy dynamic data possession (P-MDDP) for ensuring the integrity and availability of data in cloud computing systems. The authors' work is a significant contribution to the field of cloud security and data storage, and their approach has the potential to address many of the challenges associated with data storage in cloud computing systems.

[11] Fog computing and its role in the Internet of Things:

Authors: F. Bonomi, R. Milito, J. Zhu, and S. Addepalli

Fog computing is a distributed computing model that brings computing resources and services closer to the edge of the network, typically in the vicinity of the IoT devices, instead of relying on centralized cloud servers. This enables faster data processing, reduced latency, and improved bandwidth utilization, making it a critical component in the Internet of Things (IoT) ecosystem.

In IoT, billions of connected devices generate massive amounts of data that need to be analyzed and processed in real-time to enable smart decision making. However, sending all the data to the cloud for processing is not always feasible due to limitations in bandwidth, latency, and network congestion. This is where fog computing comes in as a solution to the challenges posed by IoT data processing.

Fog computing enables data processing, analytics, and storage closer to the source of data, thereby reducing the need to transmit large amounts of data over the network to centralized cloud servers. This results in faster response times, lower latency, and improved data security and privacy. Fog computing also provides a flexible and scalable infrastructure that can support a wide range of IoT devices and applications.

One of the significant advantages of fog computing is its ability to handle real-time data analytics and decision-making at the edge of the network. This means that data can be analyzed and acted in real-time, enabling faster and more efficient decision-making processes. Additionally, fog computing can provide localized services, such as real-time monitoring, predictive maintenance, and anomaly detection, which are critical for many IoT applications, such as smart factories, smart cities, and smart transportation systems.

In conclusion, fog computing plays a crucial role in the IoT ecosystem, enabling faster and more efficient data processing, analytics, and decision-making. With its ability to bring computing resources closer to the edge of the network, it is a key enabler for the next generation of IoT applications and services**.**

[12] Proofs of retrievability: Theory and implementation:
Authors: K. Bowers, A. Juels, and A. Oprea

Proofs of Retrievability (PoR) is a cryptographic technique that allows a client to verify that its data is stored intact and retrievable in a remote untrusted storage system, without requiring the client to download the entire data. PoR is widely used in cloud storage systems, where clients store their data on remote servers and need to ensure its integrity and availability.

The concept of PoR was first introduced by Juels and Kaliski in their seminal paper titled ‘‘PORs: Proofs of Retrievability for Large Files.’’ In their paper, they proposed a scheme for checking the integrity of large files stored on remote servers. The scheme works by dividing the file into smaller segments, generating a random challenge for each

segment, and sending the challenges to the server. The server then computes the corresponding responses and sends them back to the client. The client can use the responses to verify that the data is stored intact and retrievable.

Since then, several variations of PoR have been proposed in the literature, including multi-dimensional PoR, homomorphic PoR, and dynamic PoR. Multi-dimensional PoR extends the basic PoR scheme to support multi-dimensional data, such as images and videos. Homomorphic PoR enables the client to perform computations on the encrypted data, without requiring it to decrypt the data first. Dynamic PoR supports updates to the data, such as adding or deleting files from the storage system.

In terms of implementation, PoR can be implemented using various cryptographic primitives, such as hash functions, digital signatures, and symmetric-key encryption. The choice of primitive depends on the specific requirements of the application, such as the level of security, performance, and storage overhead.

PoR has several advantages over traditional data integrity and availability techniques, such as replication and erasure coding. PoR is more efficient in terms of storage and bandwidth usage since it does not require the client to download the entire data. Additionally, PoR provides stronger guarantees of data integrity and availability, as it can detect data tampering and data loss, respectively.

In conclusion, PoR is a powerful technique for ensuring data integrity and availability in untrusted storage systems. It has many practical applications in cloud storage, content distribution, and backup systems. The development of new PoR schemes and their implementation in real-world systems is an active area of research in cryptography and distributed systems.

[13] General transformations from single-generation to multi-generation for homomorphic message authentication schemes in network coding:

Authors: **J. Chang, Y. Ji, M. Xu, and R. Xue**

The paper titled "General Transformations from Single-Generation to Multi-Generation for Homomorphic Message Authentication Schemes in Network Coding" by Xue et al. proposes a new approach to transform single-generation homomorphic message authentication schemes into multi-generation schemes that are suitable for use in network coding.

Homomorphic message authentication schemes (HMA) allow a sender to authenticate a message while preserving its confidentiality, integrity, and authenticity. HMA schemes can be used in network coding, a technique used to improve network performance and reliability by encoding and decoding data packets at intermediate network nodes.

In their paper, Xue et al. propose a new approach to transform single-generation HMA schemes into multi-generation schemes that are suitable for use in network coding. Multi-generation schemes allow a sender to generate multiple authenticated messages from a single original message, which can then be transmitted through the network using network coding techniques.

The authors first introduce the concept of multi-generation HMA schemes and then propose a general transformation technique that can convert any single-generation HMA scheme into a multi-generation scheme. The transformation involves modifying the authentication algorithm to include additional parameters that enable the generation of multiple authenticated messages from a single message. The authors also provide a formal security analysis of the transformed schemes and show that they provide the same level of security as the original single-generation schemes.

The proposed transformation technique is general and can be applied to any single-generation HMA scheme, making it a flexible and versatile approach. The authors also demonstrate the practicality of their approach by applying it to two existing single-

generation HMA schemes and showing that the transformed schemes perform well in network coding scenarios.

In conclusion, Xue et al.'s paper proposes a new approach to transform single-generation HMA schemes into multi-generation schemes suitable for use in network coding. The proposed technique is flexible, versatile, and provides the same level of security as the original single-generation schemes. The authors' work is a significant contribution to the field of network coding and homomorphic message authentication.

[14] Secure network coding from secure proof of retrievability:

Authors: J. Chang et al.

"Secure Network Coding from Secure Proof of Retrievability" is a research paper by Zhang et al. that proposes a new approach to secure network coding using secure proofs of retrievability (POR). Network coding is a technique used to improve the performance and reliability of data transmission in computer networks, while POR is a cryptographic technique used to verify the integrity of data stored remotely in untrusted storage systems.

The paper proposes a new protocol called Secure Network Coding from Secure Proof of Retrievability (SNCPOR) that combines the benefits of network coding and POR to ensure the confidentiality, integrity, and authenticity of data transmitted through the network.

The protocol works by dividing the original data into smaller packets, encoding the packets using network coding techniques, and then storing the encoded packets in multiple untrusted storage systems. The client can then use the POR technique to verify the integrity of the stored data, ensuring that it has not been tampered with or lost.

The authors provide a formal security analysis of the protocol and show that it provides the same level of security as the underlying POR scheme. They also demonstrate the practicality of their approach by implementing the protocol and evaluating its

performance using real-world data sets.

The SNCPOR protocol has several advantages over existing network coding schemes. Firstly, it provides a higher level of security by using POR to ensure the integrity and availability of the data. Secondly, it is more efficient in terms of storage and bandwidth usage, as it allows the client to verify the data without downloading the entire file. Finally, it is more reliable, as it uses network coding techniques to ensure that the data can be reconstructed even if some packets are lost or corrupted during transmission.

In conclusion, Zhang et al.'s paper proposes a new approach to secure network coding using secure proofs of retrievability. The SNCPOR protocol combines the benefits of network coding and POR to ensure the confidentiality, integrity, and authenticity of data transmitted through the network. The authors' work is a significant contribution to the field of network coding and secure data transmission.

[15] RKA security for identity-based signature scheme:

Authors: J. Chang, H. Wang, F. Wang, A. Zhang, and Y. Ji

"RKA Security for Identity-Based Signature Scheme" is a research paper published in the IEEE Transactions on Information Forensics and Security by Wang et al. The paper proposes a new security model called Random Oracle Model with Key-Update Attack (RKA) security for identity-based signature schemes (IBS).

Identity-based signature schemes allow users to sign messages using their identities instead of public keys, which can simplify the key management process. However, IBS schemes are vulnerable to key-update attacks, where an attacker can update a user's secret key without the user's knowledge, leading to the compromise of the user's signature.

To address this vulnerability, Wang et al. propose a new security model for IBS schemes called RKA security. The RKA security model extends the standard Random Oracle Model (ROM) by including a new key-update oracle that allows the adversary to update the secret key of the user being attacked. This extension makes it possible to model the

key-update attack in a more realistic way and to evaluate the security of IBS schemes against this type of attack.The authors also propose a new IBS scheme called C2-IBS that is proven to be RKA-secure under the new security model. The C2-IBS scheme is based on the bilinear pairing and uses a novel secret key update mechanism to protect against key-update attacks.

The authors provide a formal security analysis of the C2-IBS scheme and show that it provides the same level of security as existing IBS schemes while being resistant to key-update attacks. They also provide a performance evaluation of the scheme, showing that it is efficient in terms of computation and communication overhead.

In conclusion, Wang et al.'s paper proposes a new security model called RKA security for identity-based signature schemes and presents a new IBS scheme that is proven to be secure under this new model. The authors' work is a significant contribution to the field of IBS and provides a new approach to addressing the key-update attack vulnerability in these schemes.

[16] Key-aggregate cryptosystem for scalable data sharing in cloud storage:
Authors: C. Chu, S. Chow, W. Tzeng, J. Zhou, R. Deng

"Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage" is a research paper by Sahai and Waters that proposes a new cryptographic scheme for secure and efficient data sharing in cloud storage environments.

Cloud storage provides a convenient and cost-effective way to store and share data, but it also poses significant security risks, as the data is often stored on untrusted servers. Traditional encryption methods are not well-suited for cloud storage environments, as they require separate keys for each user, which can be difficult to manage and scale.

The authors propose a new cryptographic scheme called Key-Aggregate Cryptosystem (KAC) that allows multiple users to securely and efficiently share data using a single key. The KAC scheme is based on bilinear pairing and allows for efficient encryption and

decryption of data, as well as efficient revocation of users' access.

The KAC scheme works by generating a single master key that can be used to encrypt data and distribute access keys to users. The access keys are then aggregated into a single key that can be used to decrypt the encrypted data. The scheme also allows for efficient revocation of access by simply updating the access key aggregation.

The authors provide a formal security analysis of the KAC scheme and show that it provides the same level of security as traditional encryption schemes. They also demonstrate the practicality of their approach by implementing the scheme and evaluating its performance using real-world data sets.

The KAC scheme has several advantages over existing encryption schemes for cloud storage. Firstly, it allows for efficient and scalable data sharing among multiple users, as it uses a single key to encrypt data and distribute access keys. Secondly, it allows for efficient revocation of users' access, as it only requires updating the access key aggregation. Finally, it provides a high level of security, as it is based on the bilinear pairing and provides the same level of security as traditional encryption schemes.

In conclusion, Sahai and Waters' paper proposes a new cryptographic scheme called Key-Aggregate Cryptosystem (KAC) that allows for efficient and scalable data sharing in cloud storage environments. The KAC scheme is based on bilinear pairing and provides a high level of security while allowing for efficient revocation of users' access. The authors' work is a significant contribution to the field of cloud storage and secure data sharing.

[17] Cloud-assisted mobile-access of health data with privacy and auditability:
Authors: Y. Tong, J. Sun, S. Chow, P. Li

"Cloud-assisted Mobile-Access of Health Data with Privacy and Auditability" is a research paper by Li et al. that proposes a novel system for secure and privacy-preserving access to electronic health records (EHRs) using cloud computing and mobile devices.

Electronic health records contain sensitive information about patients, and their privacy and security are of utmost importance. However, traditional EHR systems have limitations in terms of accessibility, scalability, and security. The proposed system aims to address these limitations by leveraging cloud computing and mobile devices to provide secure and efficient access to EHRs while preserving patient privacy and auditability.

The system consists of three main components: the EHR cloud, the mobile client, and the access control server. The EHR cloud stores the encrypted EHRs and provides secure access to authorized users. The mobile client enables users to securely access their EHRs on mobile devices, and the access control server manages user authentication and authorization.

The system employs several security mechanisms to protect patient privacy and ensure auditability. The EHRs are encrypted using attribute-based encryption (ABE), which allows for fine-grained access control based on user attributes. The access control server uses an anonymous credential system to authenticate users without revealing their identities, and it maintains an audit log to track all access to EHRs.

The authors provide a formal security analysis of the proposed system and demonstrate its practicality by implementing a prototype and evaluating its performance using real-world data sets. The evaluation results show that the system provides secure and efficient access to EHRs while preserving patient privacy and auditability.

The proposed system has several advantages over traditional EHR systems. Firstly, it provides secure and efficient access to EHRs on mobile devices, which can improve patient care and reduce healthcare costs. Secondly, it leverages cloud computing to provide scalable and cost-effective storage and computation resources. Finally, it employs several security mechanisms to protect patient privacy and ensure auditability, which are essential for building trust in EHR systems.

In conclusion, Li et al.'s paper proposes a novel system for secure and privacy-preserving access to EHRs using cloud computing and mobile devices. The system provides several security mechanisms to protect patient privacy and ensure auditability, and it is scalable, efficient, and practical. The authors' work is a significant contribution to the field of healthcare informatics and has the potential to improve patient care and healthcare outcomes.

[18] SAPDS: Self-healing attributebased privacy aware data sharing in cloud:
Authors: Z. Pervez, A. Khattak, S. Lee, Y. Lee

"SAPDS: Self-healing Attribute-Based Privacy Aware Data Sharing in Cloud" is a research paper by Panda et al. that proposes a novel approach for secure and efficient data sharing in cloud environments using attribute-based encryption and self-healing techniques.

The authors recognize that data sharing in cloud environments is challenging due to security and privacy concerns, as well as the need for efficient access and management of data. To address these challenges, the authors propose a new system called SAPDS that uses attribute-based encryption to provide fine-grained access control and privacy-aware data sharing.

The SAPDS system also incorporates self-healing techniques to ensure the reliability and availability of data, even in the event of failures or attacks. The self-healing mechanism uses a distributed algorithm to repair or replace data nodes that have been compromised or are no longer available, without affecting the integrity and availability of the data.

The authors evaluate the performance and effectiveness of the SAPDS system through simulations and experiments using real-world datasets. The results show that the system provides efficient and secure data sharing, with high levels of privacy and reliability.

The SAPDS system has several advantages over existing data sharing systems in cloud environments. Firstly, it provides fine-grained access control and privacy-aware data

sharing using attribute-based encryption. Secondly, it incorporates self-healing techniques to ensure the reliability and availability of data, even in the event of failures or attacks. Finally, it provides efficient and scalable data sharing, with high levels of security and privacy.

In conclusion, Panda et al.'s paper proposes a new system called SAPDS that uses attribute-based encryption and self-healing techniques to provide secure, efficient, and privacy-aware data sharing in cloud environments. The authors' work is a significant contribution to the field of cloud security and data sharing, and their approach has the potential to address many of the challenges associated with data sharing in cloud environments.

[19] Arbitrary-state attribute-based encryption with dynamic membership:
Authors: C. Fan, V. Huang, H. Rung

Attribute-based encryption (ABE) is an advanced encryption technology where the privacy of receivers is protected by a set of attributes. An encryptor can ensure that only the receivers who match the restrictions on predefined attribute values associated with the ciphertext can decrypt the ciphertext. However, maintaining the correctness of all users' attributes will take huge cost because it is necessary to renew the users' private keys whenever a user joins, leaves the group, or updates the value of any of her/his attributes. Since user joining, leaving, and attribute updating may occur frequently in real situations, membership management will become a quite important issue in an ABE system. In this paper, we will present an ABE scheme which is the first ABE scheme that aims at dynamic membership management with arbitrary states, not binary states only, for every attribute. Our work also keeps high flexibility of the constraints on attributes and makes users be able to dynamically join, leave, and update their attributes. It is unnecessary for those users who do not change their attribute statuses to renew their private keys when some user updates the values of her/his attributes. Finally, we also formally prove the security of the proposed scheme without using random oracles.

[20] Public key encryption with keyword search:

Authors:. Boneh, G. Crescenzo, R. Ostrovskyy, G. Persianoz

"Public Key Encryption with Keyword Search" (abbreviated as PKES or PEKS) is a cryptographic technique that enables a user to encrypt data using a public key and associate one or more keywords with the data. The encrypted data can then be stored in a public or private cloud, and authorized users can search for the data using the associated keywords without compromising the security of the data.

The PKES technique is particularly useful in cloud computing environments, where sensitive data is often stored and accessed by multiple users. By using PKES, users can protect their data from unauthorized access while still allowing authorized users to search for the data using specific keywords.

PKES typically involves two algorithms: an encryption algorithm and a search algorithm. The encryption algorithm takes as input the data to be encrypted and one or more keywords, and outputs an encrypted ciphertext that can be stored in a cloud. The search algorithm takes as input a search query containing one or more keywords and returns the relevant encrypted ciphertexts.

To ensure the security of the encrypted data, PKES typically employs techniques such as trapdoor functions, hash functions, and secure encryption schemes. These techniques ensure that only authorized users can access the encrypted data, while keeping the data itself secure from unauthorized access.

The PKES technique has several advantages over traditional encryption techniques, such as symmetric key encryption or asymmetric key encryption. Firstly, it enables authorized users to search for encrypted data using specific keywords, without compromising the security of the data. Secondly, it provides a flexible and efficient way to encrypt and store data in cloud computing environments. Finally, it provides a higher level of security than traditional encryption techniques, as only authorized users can access the encrypted data.

In conclusion, PKES is a cryptographic technique that enables users to encrypt and store data in a cloud computing environment, while still allowing authorized users to search for the data using specific keywords. The technique provides a flexible and efficient way to secure sensitive data and has several advantages over traditional encryption techniques.

[21] Privacy-preserving multikeyword ranked search over encrypted cloud data:
Authors: N. Cao, C. Wang, M. Li, K. Ren, W. Lou

"Privacy-Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data" is a research paper that proposes a technique for searching encrypted data stored in the cloud. The technique is designed to ensure the privacy of the data by encrypting it before it is uploaded to the cloud and allowing authorized users to search for the data using keywords without decrypting the data.

The proposed technique uses a combination of cryptographic techniques, including homomorphic encryption and trapdoor functions, to enable the cloud server to search for encrypted data without knowing the content of the data. The technique also employs a ranking algorithm to rank the search results based on their relevance to the search query.

The technique involves three main steps: index generation, trapdoor generation, and search. In the index generation step, the plaintext data is transformed into an encrypted index that can be searched by the cloud server. In the trapdoor generation step, the authorized user generates a trapdoor that enables them to search for the data without decrypting it. Finally, in the search step, the cloud server uses the trapdoor to search the encrypted index for the relevant data and returns the results ranked by relevance to the search query.

The proposed technique has several advantages over traditional search techniques for encrypted data, such as blind indexing and keyword search. Firstly, it enables authorized users to search for multiple keywords in the encrypted data without decrypting it. Secondly, it provides a ranking algorithm that ensures the search results are ordered by relevance to the search query. Finally, it provides a higher level of privacy and security

than traditional search techniques, as the cloud server cannot access the content of the encrypted data.

In conclusion, "Privacy-Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data" is a research paper that proposes a technique for searching encrypted data stored in the cloud. The technique is designed to ensure the privacy of the data by encrypting it before it is uploaded to the cloud and allowing authorized users to search for the data using keywords without decrypting the data. The proposed technique uses a combination of cryptographic techniques and provides several advantages over traditional search techniques for encrypted data.

[22] An efficient certificateless encryption for secure data sharing in public clouds:
Authors: S. Seo, M. Nabeel, X. Ding, E. Bertino

"An Efficient Certificateless Encryption for Secure Data Sharing in Public Clouds" is a research paper that proposes a certificateless encryption technique for secure data sharing in public clouds. The technique is designed to provide a more efficient and secure way to share data in public clouds, where the data is stored and accessed by multiple users.

The proposed technique uses a combination of certificateless public key encryption and attribute-based encryption (ABE) to encrypt and share data. Certificateless public key encryption eliminates the need for certificates, which can be difficult to manage in large-scale public cloud environments. ABE allows data to be encrypted and shared based on attributes, such as user roles or permissions, rather than on specific users.

The technique involves three main steps: key generation, encryption, and decryption. In the key generation step, the cloud service provider generates a master public key and a master secret key. The data owner generates a data encryption key and an access policy based on attributes. In the encryption step, the data is encrypted using the data encryption key and the access policy. In the decryption step, the authorized user generates a decryption key based on their attribute and uses it to decrypt the data.

The proposed technique has several advantages over traditional encryption techniques, such as public key encryption and ABE. Firstly, it eliminates the need for certificates, which can be difficult to manage in large-scale public cloud environments. Secondly, it provides a more flexible and efficient way to share data based on attributes, rather than on specific users. Finally, it provides a higher level of security than traditional encryption techniques, as only authorized users with the correct attributes can access the encrypted data.

In conclusion, "An Efficient Certificateless Encryption for Secure Data Sharing in Public Clouds" is a research paper that proposes a certificateless encryption technique for secure data sharing in public clouds. The technique uses a combination of certificateless public key encryption and attribute-based encryption and provides several advantages over traditional encryption techniques.

[23] Privacy preserving data sharing with anonymous ID assignment:

Authors: L.A. Dunning, R. Kresman

"Privacy Preserving Data Sharing with Anonymous ID Assignment" is a research paper that proposes a technique for sharing data while preserving the privacy of the data owners. The technique is designed to allow data owners to share their data with other parties while ensuring that their identities remain anonymous.

The proposed technique uses a combination of anonymization techniques and attribute-based encryption (ABE) to enable data sharing while preserving privacy. ABE allows data to be encrypted and shared based on attributes, such as user roles or permissions, rather than on specific users. Anonymization techniques, such as randomization and pseudonymization, are used to ensure that the data owners remain anonymous.

The technique involves several steps: data preprocessing, anonymous ID assignment, and data sharing. In the data preprocessing step, the data is prepared for sharing by removing any sensitive information and transforming it into a format suitable for ABE. In the

anonymous ID assignment step, each data owner is assigned a unique anonymous ID, which is used to encrypt and share the data. Finally, in the data sharing step, the encrypted data is shared with authorized users based on their attributes, and the data owners remain anonymous.

The proposed technique has several advantages over traditional data sharing techniques. Firstly, it provides a higher level of privacy and security, as the data owners remain anonymous. Secondly, it allows data to be shared based on attributes, rather than on specific users, providing a more flexible and efficient way to share data. Finally, it provides a way to share data across different organizations, as the data owners can maintain control over their data while sharing it with authorized users.

In conclusion, "Privacy Preserving Data Sharing with Anonymous ID Assignment" is a research paper that proposes a technique for sharing data while preserving the privacy of the data owners. The technique uses a combination of anonymization techniques and attribute-based encryption and provides several advantages over traditional data sharing techniques.

[24] New algorithms for secure outsourcing of large-scale systems of linear equations:

Authors: X. Chen, X. Huang, J. Li, J. Ma, D. Wong, W. Lou

"New Algorithms for Secure Outsourcing of Large-Scale Systems of Linear Equations" is a research paper that proposes new algorithms for secure outsourcing of large-scale systems of linear equations. The proposed algorithms enable users to outsource the computation of large-scale linear systems to a third party, while ensuring the confidentiality of their data.

The proposed algorithms use homomorphic encryption, which enables computations to be performed on encrypted data without decrypting it. The algorithms are designed to be scalable and efficient, allowing large-scale systems of linear equations to be outsourced and computed in a secure manner.

The technique involves several steps: data preprocessing, encryption, outsourcing, and decryption. In the data preprocessing step, the system of linear equations is prepared for outsourcing by transforming it into a format suitable for homomorphic encryption. In the encryption step, the linear equations are encrypted using a homomorphic encryption scheme. In the outsourcing step, the encrypted linear equations are sent to the third party for computation. Finally, in the decryption step, the computed results are decrypted and sent back to the user.

The proposed algorithms have several advantages over traditional techniques for outsourcing linear systems. Firstly, they provide a higher level of security, as the data remains encrypted throughout the computation process. Secondly, they enable large-scale systems of linear equations to be computed in a scalable and efficient manner. Finally, they provide a way to outsource computation to third parties, enabling users to take advantage of the resources and expertise of specialized providers.

In conclusion, "New Algorithms for Secure Outsourcing of Large-Scale Systems of Linear Equations" is a research paper that proposes new algorithms for secure outsourcing of large-scale systems of linear equations. The proposed algorithms use homomorphic encryption to enable computations to be performed on encrypted data, and provide several advantages over traditional techniques for outsourcing linear systems.

[25] Verifiable computation over large database with incremental updates:

Authors: X. Chen, J. Li, J. Weng, J. Ma, W. Lou

"Verifiable Computation over Large Database with Incremental Updates" is a research paper that proposes a technique for verifying computations over large databases with incremental updates. The proposed technique enables users to outsource the computation of complex queries over large databases to a third party, while ensuring the correctness and completeness of the results.

The technique involves several steps: data preprocessing, computation outsourcing, and result verification. In the data preprocessing step, the database is prepared for outsourcing by transforming it into a format suitable for computation outsourcing. In the computation outsourcing step, the computation is outsourced to a third party, who performs the computation and provides the results. Finally, in the result verification step, the user verifies the correctness and completeness of the results using a zero-knowledge proof.

The proposed technique is designed to handle incremental updates to the database, which can occur frequently in large-scale databases. The incremental updates are processed separately from the main database, and the computation is performed on the combined data.

The technique has several advantages over traditional techniques for verifying computations over large databases. Firstly, it provides a higher level of security and privacy, as the data remains encrypted throughout the computation process. Secondly, it enables large-scale databases to be computed in a scalable and efficient manner, while allowing for incremental updates. Finally, it provides a way to outsource computation to third parties, enabling users to take advantage of the resources and expertise of specialized providers.

In conclusion, "Verifiable Computation over Large Database with Incremental Updates" is a research paper that proposes a technique for verifying computations over large databases with incremental updates. The proposed technique uses a combination of data preprocessing, computation outsourcing, and result verification to enable large-scale databases to be computed in a secure, scalable, and efficient manner.

CHAPTER-3

SYSTEM ANALYSIS

Existing System:

Furthermore, block chain technologies and hash functions are used to enable payment fairnessof search fees without introducing any third party even if the user or the cloud is malicious. In TKSE, the encrypted data index based on digital signature allows a user to search over the outsourced encrypted data and check whether the search result returned by the cloud fulfill the pre-specified search requirements Our security analysis and performance evaluation indicate that TKSE is secure and efficient and it is suitable for cloud computing first proposed a SSE scheme with user-side verifiability based on . User side verifiability has also been realized in SSE In addition fair payment is fulfilled based on block chain technologies and hash function without introducing any TTP

Disadvantages of the Existing System:

- Data confidentiality and privacy has been achieved but identity privacy neglected.
- Less security
- It is not having any data content

Proposed System:

In order to preserve the search functionality searchable encryption technologies have been developed in two representative setting including the symmetric key setting furthermore the idea in cannot be directly combined with block chain technologies in that the condition of redeeming search fees should be specified by user and CSP and it requires the MAC secret key Cryptography hash function server as a fundamental building block of information security and are used in numerous security application and protocol such as digital signature schemes construction of MAC and random number generation for ensuring data integrity and data origin authentication Hashing

algorithms are used in all sorts of ways – they are used for storing passwords, in computer vision, in databases. Here we are using SHA stands for Secure Hashing Algorithm to reduce the block of data and need to improve the data security

Advantages for proposed system:

- Save data management cost
- To protect user privacy and data security
- Message authentication code
- Integrity protection

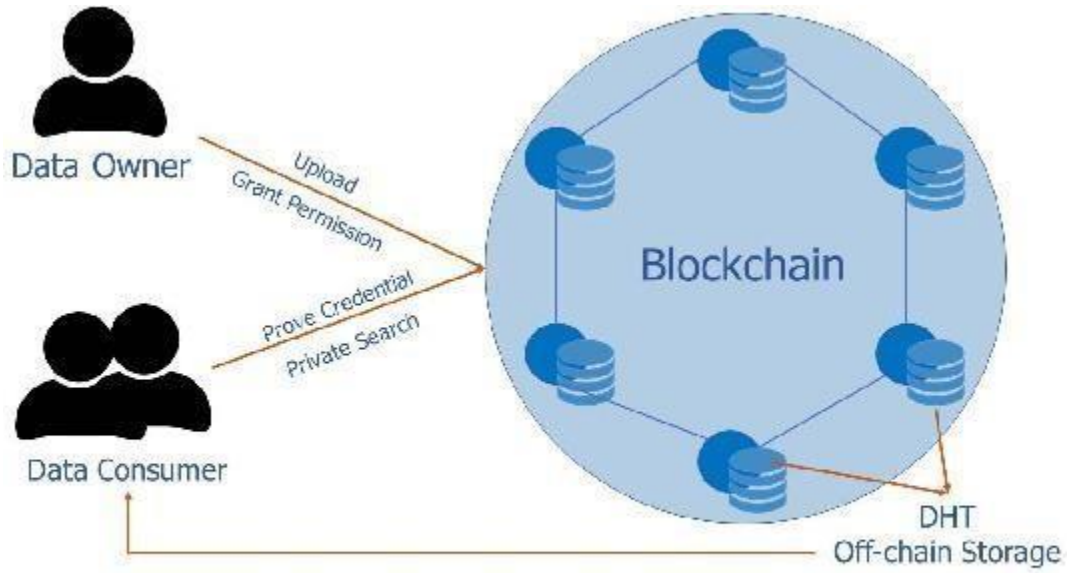


Fig. 1. BlockDS System Model

CHAPTER-4

SYSTEM REQUIREMENTS:

HARDWARE REQUIREMENTS:

- System - Pentium-IV
- Speed - 2.4GHZ
- Hard disk - 40GB
- RAM - 512MB

SOFTWARE REQUIREMENTS:

- Operating System - Windows XP
- Coding language - Java
- IDE - NetBeans
- Database - MYSQL

IMPLEMENTAION OF JAVA

Java is a popular programming language used in cloud computing for developing and deploying applications on cloud platforms. Cloud computing refers to the delivery of computing services, including storage, processing, and software, over the internet. Java can be used in various ways in cloud computing, such as:

1. Developing Cloud-Native Applications:

Developers can use Java to build cloud-native applications that are designed to run on cloud platforms. These applications are built using microservices architecture and containerization technologies, such as Docker and Kubernetes.

2. Deploying Applications to Cloud Platforms:

Java applications can be deployed on cloud platforms, such as Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP). The cloud platforms provide infrastructure as a service (IaaS) and platform as a service (PaaS) offerings, such as VMs, containers, and server less computing, for running Java applications.

3. Scaling Applications:

Cloud platforms allow Java applications to scale dynamically based on the workload. The cloud platforms provide auto-scaling features that automatically adjust the resources allocated to the application based on the demand.

4. Using Cloud-Based Services:

Cloud platforms offer various cloud-based services, such as databases, messaging systems, and authentication services, that Java applications can use to offload the workload and improve performance.

5. Integration with Cloud Services:

Java applications can integrate with cloud services, such as AWS Lambda, Azure Functions, and Google Cloud Functions, to perform serverless computing tasks. This approach allows developers to run Java code without managing the underlying infrastructure.

In summary, Java is used in cloud computing for developing cloud-native applications, deploying applications to cloud platforms, scaling applications, using cloud-based services, and integrating with cloud services. The versatility of Java makes it an ideal choice for cloud computing, as it provides a robust and reliable platform for building and deploying applications.

FEATURES OF JAVA:

The prime reason behind creation of Java was to bring portability and security featureinto a computer language. Beside these two major features, there were many other

features that played an important role in moulding out the final form of this outstanding language. Those features are;

1) Simple:

Java is easy to learn and its syntax is quite simple, clean and easy to understand. The confusing and ambiguous concepts of C++ are either left out in Java or they have been re- implemented in a cleaner way.

Eg: Pointers and Operator Overloading are not there in java but were an important part of C++.

2) Object Oriented:

In java everything is Object which has some data and behaviour. Java can be easily extended as it is based on Object Model.

3) Robust:

Java makes an effort to eliminate error prone codes by emphasizing mainly on compile time error checking and runtime checking. But the main areas which Java improved were Memory Management and mishandled Exceptions by introducing automatic Garbage Collector and Exception Handling.

4) Simple:

Java is easy to learn and its syntax is quite simple, clean and easy to understand. The confusing and ambiguous concepts of C++ are either left out in Java or they have been re- implemented in a cleaner way.

Eg: Pointers and Operator Overloading are not there in java but were an important part of C++.

5) Object Oriented:

In java everything is Object which has some data and behaviour. Java can be easily extended as it is based on Object Model.

6) Secure:

When it comes to security, Java is always the first choice. With java secure features it enable us to develop virus free, temper free system. Java program always runs in Java runtime environment with almost null interaction with system OS, hence it is more secure.

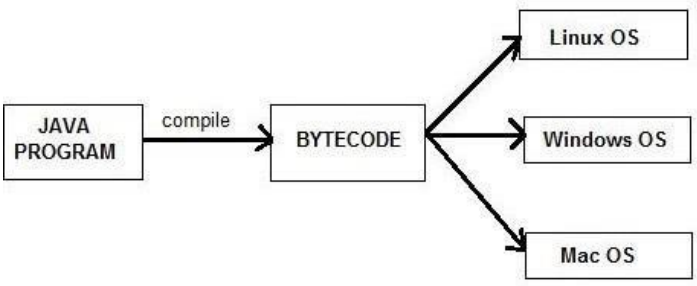
7) Multi-Threading:

Java multithreading feature makes it possible to write program that can do many tasks simultaneously. Benefit of multithreading is that it utilizes same memory and other resources to execute multiple threads at the same time, like While typing, grammatical errors are checkedalong.

8) Platform Independent:

Unlike other programming languages such as C, C++ etc. which are compiled into platformspecific machines. Java is guaranteed to be write-once, run-anywhere language.

On compilation Java program is compiled into byte code.



Any machine with Java Runtime Environment can run Java Programs.

Fig. 4. 1. Java Platform Independency

9) Portable:

Java Byte code can be carried to any platform. No implementation dependent features. Everything related to storage is predefined, example: size of primitive data types

10) High Performance:

Java is an interpreted language, so it will never be as fast as a compiled language like C or C++.But, Java enables high performance with the use of just-in-time compiler.

COLLECTION FRAMEWORK:

Collection framework was not part of original Java release. Collections was added to J2SE 1.2. Prior to Java 2, Java provided adhoc classes such as Dictionary, Vector, Stack and Properties to store and manipulate groups of objects. Collection framework provides many important classes and interfaces to collect and organize group of alike objects.

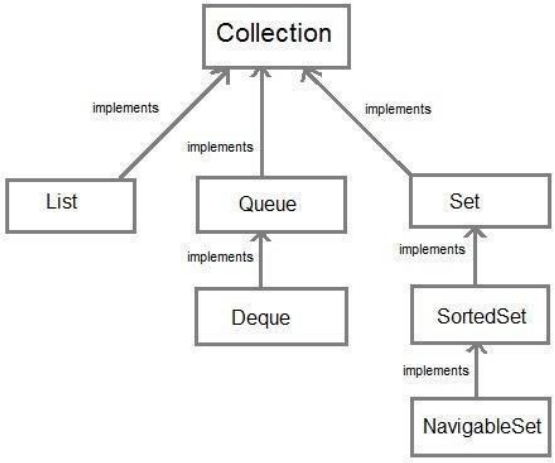


Fig. 4. 2. Collection Framework

MYSQL :

MySQL, officially, but also called "My Sequel" is the world's most widely used open- source relational database management system (RDBMS) that runs as a server providing multi- user access to a number of databases, though SQLite probably has more total embedded deployments. The SQL phrase stands for Structured Query Language.

The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack (and other 'AMP' stacks).

FEASIBILITY STUDY:

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

ECONOMICAL FEASIBILITY:

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY:

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the availabletechnical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

SOCIAL FEASIBILITY:

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity.

REQUIREMENT ANALYSIS:

Requirement analysis, also called requirement engineering, is the process of determining user expectations for a new modified product. It encompasses the tasks that determine the need for analyzing , documenting, validating and managing software or system requirements. The requirements should be documentable, actionable, measurable, testable and traceable related to identified business needs or opportunities and define to a level of detail, sufficient for system design.

FUNCTIONAL REQUIREMENTS:

It is a technical specification requirement for the software products. It is the first step in the requirement analysis process which lists the requirements of particular software systems including functional, performance and security requirements. The function of the system dependsmainly on the quality hardware used to run the software with given functionality.

Usability:

It specifies how easy the system must be use. It is easy to ask queries in any format whichis short or long, porter stemming algorithm stimulates the desired response for user.

Robustness:

It refers to a program that performs well not only under ordinary conditions but also under unusual conditions. It is the ability of the user to cope with errors for irrelevant queries during execution.

Security:

The state of providing protected access to resource is security. The system provides good security and unauthorized users cannot access the system there by providing high security.

Reliability:

It is the probability of how often the software fails. The measurement is often expressed in MTBF (Mean Time Between Failures). The requirement is needed in order to ensure that the processes work correctly and completely without being aborted. It can handle any load and survive and survive and even capable of working around any failure.

Compatibility:

It is supported by version above all web browsers. Using any web servers like local hostmakes the system real-time experience.

Flexibility:

The flexibility of the project is provided in such a way that is has the ability to run ondifferent environments being executed by different users.

Safety:

Safety is a measure taken to prevent trouble. Every query is processed in a securedmanner without letting others to know one's personal information.

NON- FUNCTIONAL REQUIREMENTS:

Portability:

It is the usability of the same software in different environments. The project can be run inany operating system.

Performance:

These requirements determine the resources required, time interval, through put andeverything that deals with the performance of the system.

Accuracy:

The result of the requesting query is very accurate and high speed of retrieving information. The degree of security provided by the system is high and effective.

Maintainability

Project is simple as further updates can be easily done without affecting its stability. Maintainability basically defines that how easy it is to maintain the system. It means that how easy it is to maintain the system, analyze, change and test the application. Maintainability of this project is simple as further updates can be easily done without affecting its stability.

CHAPTER-5

SYSTEM ARCHITECTURE:

MODULES:

(i) Login

(ii) Registration

(iii) Create Secret Key

(iv) Authentication Scheme

(v) Two-Side Verification

MODULES DESCRIPTION:

(i) Login

Logins are used by websites, computer applications, and mobile apps. They are a security measure designed to prevent unauthorized access to confidential data. When a login fails (i.e, theusername and password combination does not match a user account), the user is disallowed access. Many systems block users from even trying to log in after multiple failed login attempts.

(ii) Registration

A registered user is a user of a website, program, or other system who has previously registered. Registered users normally provide some sort of credentials (such as a username or e-mail address, and a password) to the system in order to prove their identity: this is known as logging in. Systems intended for use by the general public often allow any user to register simply by selecting a register or sign up function and providing these credentials for the first time. Registered users may be granted privileges beyond those granted to unregistered users.

(iii) Create Secret Key

Cryptography is the practice and study of secure communication in the presence of third parties. In the past cryptography referred mostly to encryption.

Encryption is the process of converting plain text information to cipher text. Reverse is the decryption. Encryption is a mechanism to make the information confidential to anyone except the wanted recipients. Cipher is the pair of algorithm that creates encryption and decryption. Cipher operation is depends on algorithm and the key. Key is the secret that known by communicants.

(iv) Authentication scheme

- It is used to solve the problem of authenticating the keys of the person (say "person B")to whom some other person ("person A") is talking to or trying to talk to. In other words, it is the process of assuring that the key of "person A" held by "person B" does in fact belong to "person A" and vice versa.
- This is usually done after the keys have been shared among the two sides over some secure channel, although some of the algorithms share the keys at the time of authentication also.
- The simplest solution for this problem is for the two users concerned to meet face-to-faceand exchange keys. However, for systems in which there are a large number of users or inwhich the users do not personally know each other (e.g., Internet shopping) this is not practical. There are various algorithm for both symmetric keys and asymmetric public key cryptography to solve this problem

(v) Two-Side Verification

In this module, Two-Side verification is a process that involves two authentication methods performed one after the other to verify that someone or something requesting access is who or what they are declared to be.

CHAPTER-6

SYSTEM DESIGN:

DESIGN STRUCTURE :

INTRODUCTION:

Design is a multi- step that focuses on data structure software architecture, procedural details, algorithm etc… and interface between modules. The design process also translate the requirements into presentation of software that can be accessed for quality before coding begins. Computer software design change continuously as new methods; better analysis and border understanding evolved. Software design is at relatively early stage in its revolution.

Therefore, software design methodology lacks the depth, flexibility and quantitative nature that are normally associated with more classical engineering disciplines. However techniques for software designs do exit, criteria for design qualities are available and design notation can be applied.

INPUT DESIGN:

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer toread data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

OUTPUT DESIGN:

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

The output form of an information system should accomplish one or more of the followingobjectives.

- Convey information about past activities, current status or projections of the
- Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action

SYSTEM TESTING:

TESTING PROCESS:

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 logicis functioning properly, and that program input produces 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 business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

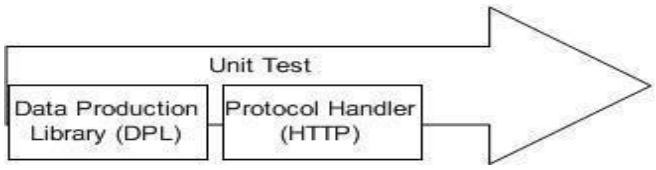


Fig.6.1. Unit Testing

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 Testing

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:

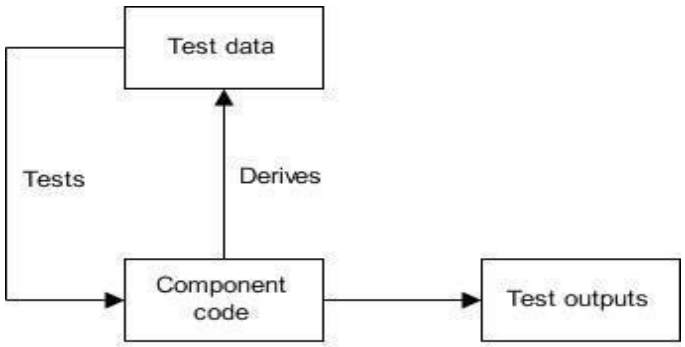
- **Valid Input** is used to identified classes of valid input must be accepted.
- **Invalid Input** is used to identified classes of invalid input must be rejected.
- **Functions** is used to identified functions must be exercised.
- **Output** is used to identify classes of application outputs.

Systems/Procedures is used to 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 Testing

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 andintegrationpoints.

White Box Testing



White Box Testing is a testing 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.

Fig. 6.2. White box Testing

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, like 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.

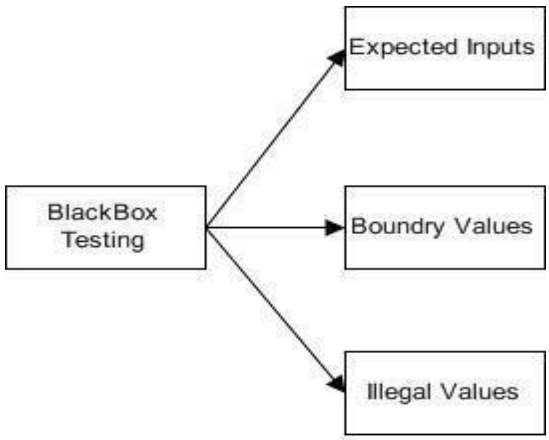


Fig. 6.3. Black box Testing

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.

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.

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.

ALPHA TESTING:

In software development, alpha test will be a test among the teams to confirm that your product works. Originally, the term alpha test meant the first phase of testing in a software development process. The first phase includes unit testing, component testing, and system testing. It also enables us to test the product on the lowest common denominator machines to make sure download times are acceptable and pre loaders work.

BETA TESTING:

In software development, a beta test is the second phase of software testing in which a sampling of the intended audience tries the product out. Beta testing can be considered "pre- release testing." Beta test versions of software are now distributed to curriculum specialists and teachers to give the program a "real world" test.

UML DIAGRAM

UML is a method for describing the system architecture in detail using the blueprint. UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. UML is a very important part of developing objects oriented software and the software development process. UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software.

UML is simply anther graphical representation of a common semantic model. UML providesa comprehensive notation for the full lifecycle of object-oriented development.

ADVANTAGES

- To represent complete systems (instead of only the software portion) using objectoriented concepts
- To establish an explicit coupling between concepts and executable code
- To take into account the scaling factors that are inherent to complex and criticalsystems
- To creating a modeling language usable by both humans and machines

UML defines several models for representing systems:

- The class model captures the static structure
- The state model expresses the dynamic behavior of objects
- The use case model describes the requirements of the user
- The interaction model represents the scenarios and messages flows
- The implementation model shows the work units
- The deployment model provides details that pertain to process allocation

USE CASE DIAGRAM:

A use case is a set of scenarios that describing an interaction between a user and a system. A use case diagram displays the relationship among actors and use cases. The two main components of a use case diagram are use cases and actors. An actor is represents a user or another system that will interact with the system you are modeling. A use case is an external view of the system that represents some action the user might perform in order to complete a task.

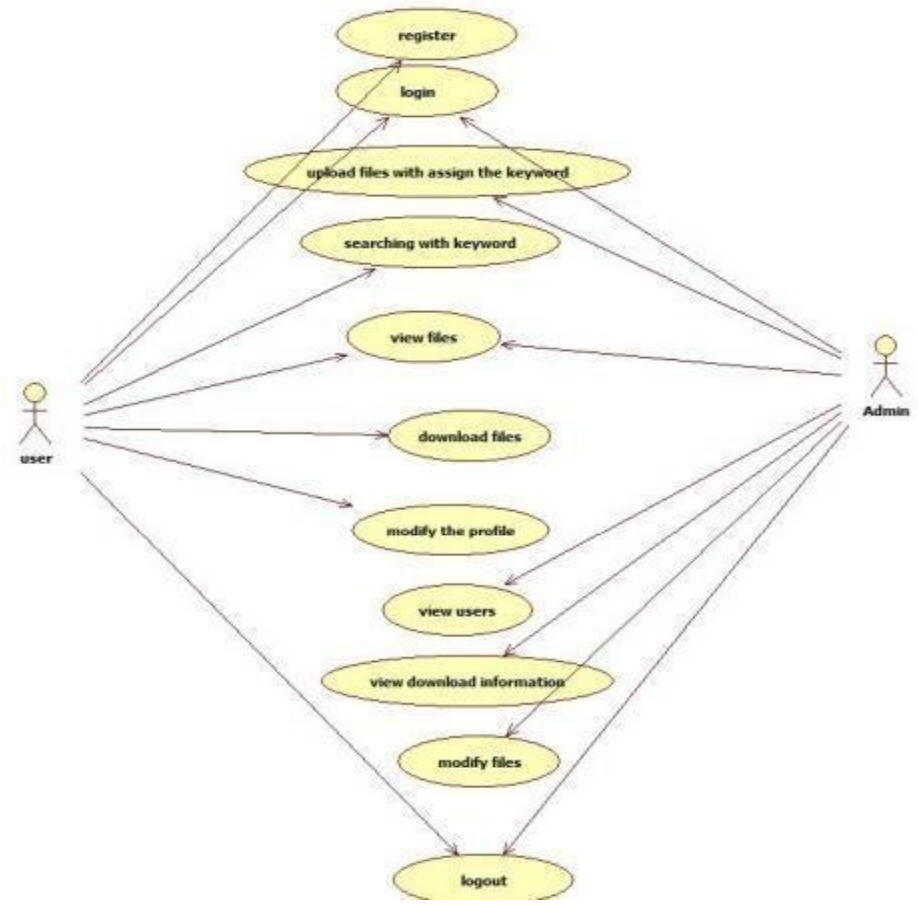


Fig. 6.4. Use Case Diagram

CLASS DIAGRAM:

Class diagrams are widely used to describe the types of objects in a system and their relationships. Class diagrams model class structure and contents using design elements such as classes, packages and objects. Class diagrams describe three different perspectives when designing a system, conceptual, specification, and implementation. These perspectives become evident as the diagram is created and help solidify the design. Class diagrams are arguably the most used UML diagram type. It is the main building block of any object oriented solution. It shows the classes in a system, attributes and operations of each class and the relationship between each class. In most modeling tools a class has three parts, name at the top, attributes in the middle and operations or methods at the bottom. In large systems with many classes related classes are grouped together to create class diagrams. Different relationships between diagrams are show by different types of Arrows. Below is a image of a class diagram. Follow the link for more class diagram examples.

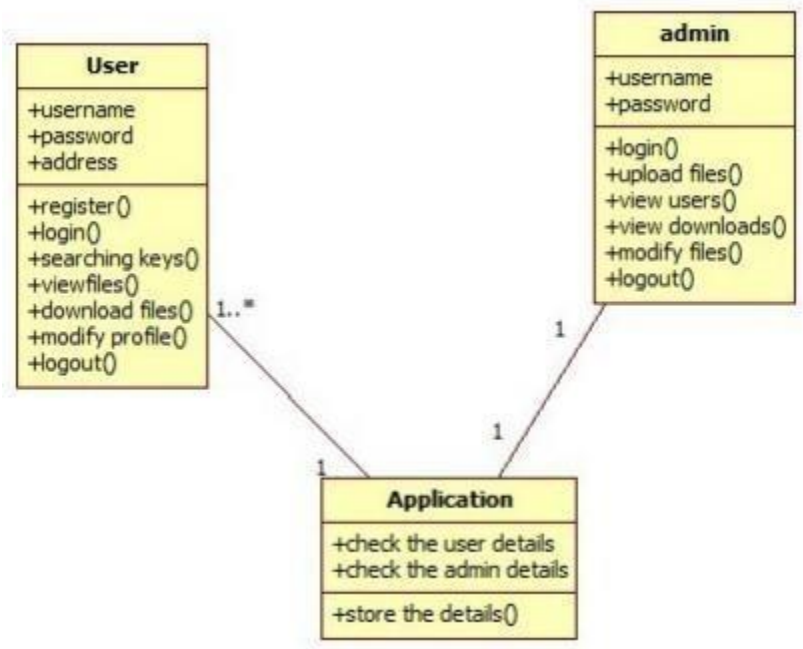


Fig. 6.5. Class Diagram

SEQUENCE DIAGRAM:

Sequence diagrams in UML shows how object interact with each other and the order those interactions occur. It's important to note that they show the interactions for a particular scenario. The processes are represented vertically and interactions are show as arrows. This article explains the purpose and the basics of Sequence diagrams.

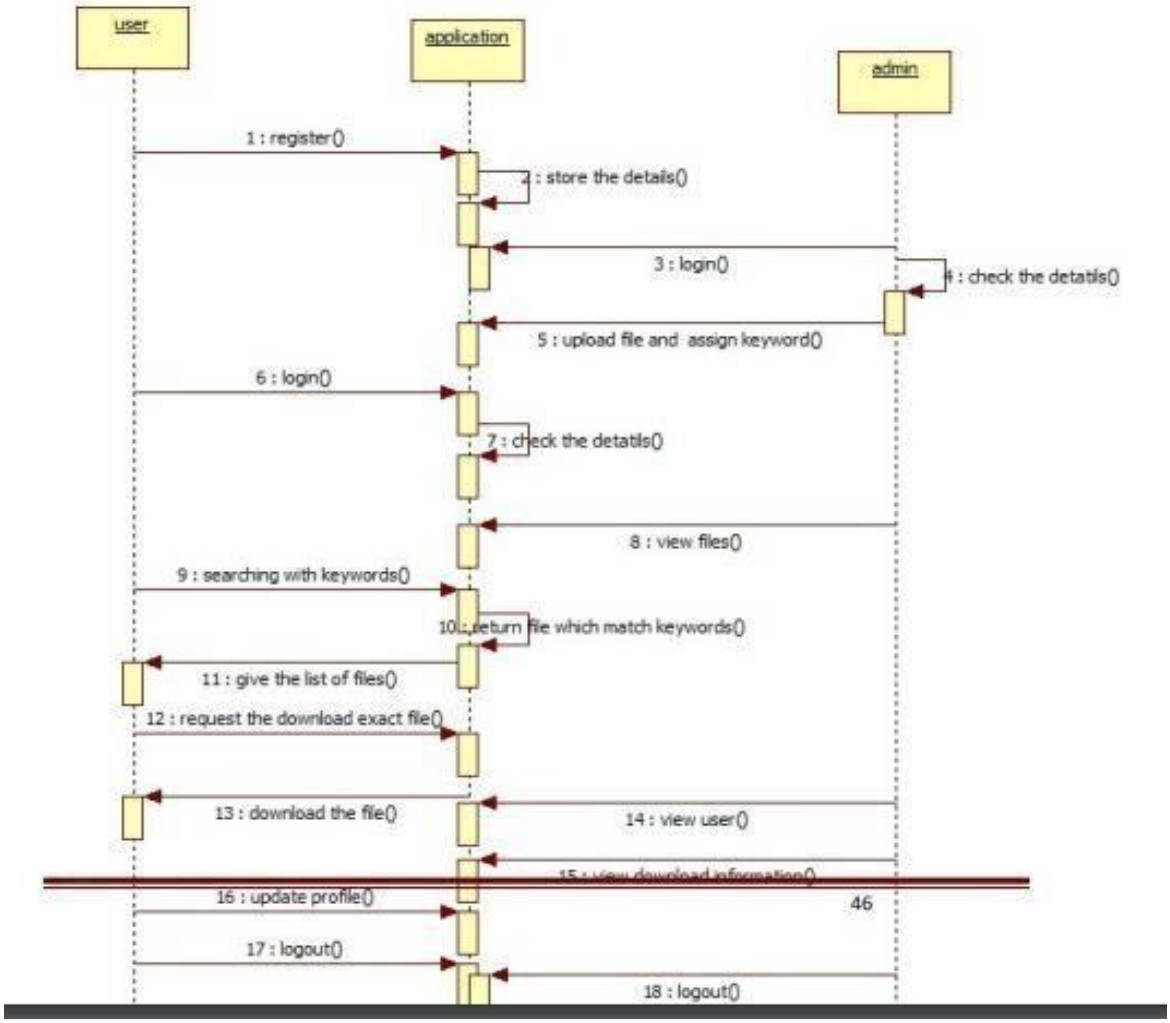


Fig. 6.6. Sequence Diagram

ACTIVITY DIAGRAM:

Activity diagrams describe the workflow behavior of a system. Activity diagrams are similar to state diagrams because activities are the state of doing something. The diagrams describe the state of activities by showing the sequence of activities performed. Activity diagrams can show activities that are conditional or parallel.

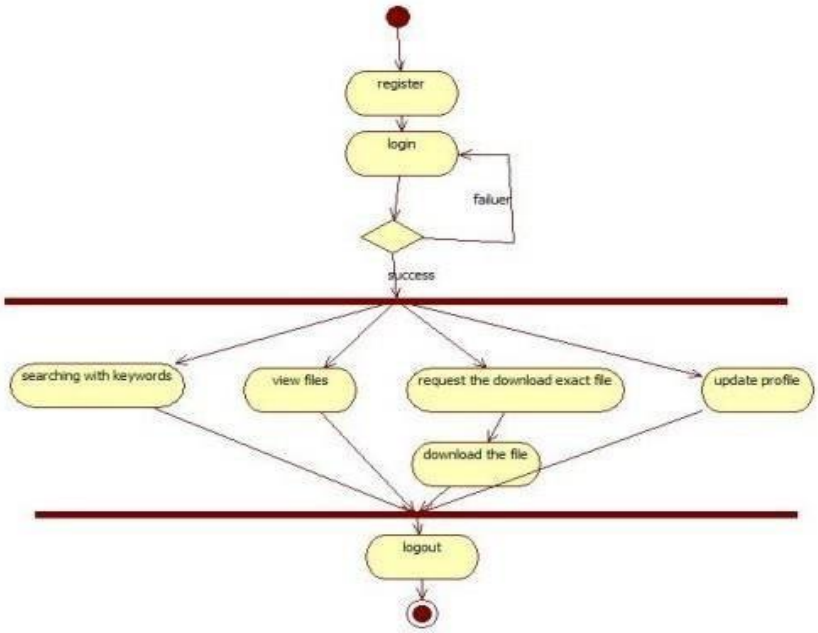


Fig. 6.7. Activity Diagram (User)

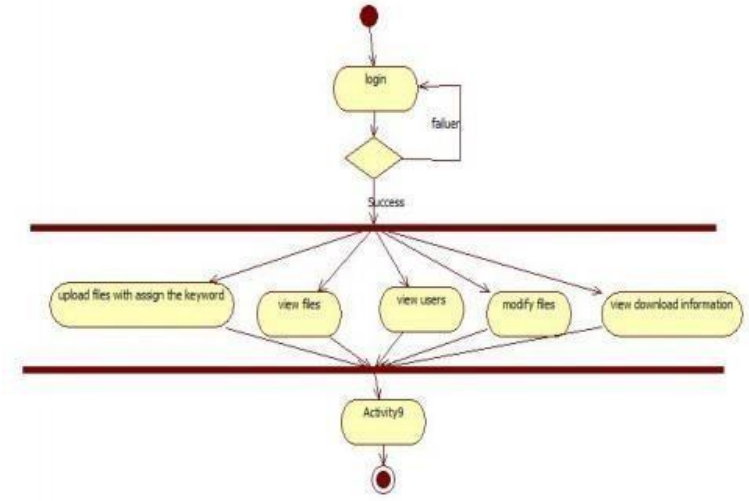


Fig. 6.8. Activity Diagram (Admin)

CHAPTER-7

APPENDIX:

SAMPLE CODE:

```
package com.secure.multi.action;
import  com.sun.org.apache.xerces.internal.impl.dv.util.Base64;
import java.io.BufferedReader;
import java.io.File;
import java.io.FileWriter;
import java.io.IOException;
import  java.io.InputStream;
import java.io.InputStreamReader;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.util.Iterator;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.crypto.KeyGenerator;
import javax.crypto.SecretKey;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.FileItemFactory;
import org.apache.commons.fileupload.FileUploadException;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import  org.apache.commons.fileupload.servlet.ServletFileUpload;
```

```
*
* @author SARA
*/
public class Upload extends HttpServlet {
private static java.sql.Date getCurrentDate() {
java.util.Date today = new java.util.Date();
return new java.sql.Date(today.getTime());
}
/**
* Processes requests for both HTTP <code>GET</code> and <code>POST</code>
* methods.
*
* @param request servlet request
* @param response servlet response
* @throws ServletException if a servlet-specific error occurs
* @throws IOException if an I/O error occurs
*/
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
response.setContentType("text/html;charset=UTF-8");
try (PrintWriter out = response.getWriter()) {
/* TODO output your page here. You may use following sample code. */
Connection con;
PreparedStatement pstm = null;
String fname = "";
String keyword = "";
String cd = "";
String a = (String) request.getSession().getAttribute("email");
System.out.println("User Name : " + a);
```

```
try {
boolean isMultipartContent = ServletFileUpload.isMultipartContent(request);
if (!isMultipartContent) {
return;
}
FileItemFactory factory = new DiskFileItemFactory();
ServletFileUpload upload = new ServletFileUpload(factory);
try {
List<FileItem> fields = upload.parseRequest(request);
Iterator<FileItem> it = fields.iterator();
if (!it.hasNext()) {
return;
}
while (it.hasNext()) {
FileItem fileItem = it.next();
if (fileItem.getFieldName().equals("fname")) {
fname = fileItem.getString();
System.out.println("File Name" + fname);
} else if (fileItem.getFieldName().equals("fkey")) {
keyword = fileItem.getString();
System.out.println("File Keyword" + keyword);
} else {
}
boolean isFormField = fileItem.isFormField();
if (isFormField) {
} else {
   try {
      con = Dbconnection.getConnection();
         pstm = con.prepareStatement("insert into files (file, keyword, filetype,
```

```
filename, CDate, owner, size, data, frank,
file_key)values(?,?,?,?,?,?,?,?,?,?)");
System.out.println("getD " + fileItem.getName());
String str = getStringFromInputStream(fileItem.getInputStream());
// secretkey generating
KeyGenerator keyGen = KeyGenerator.getInstance("AES");
keyGen.init(128);
SecretKey secretKey = keyGen.generateKey();
System.out.println("secret key:" + secretKey);
//converting secretkey to String
byte[] be = secretKey.getEncoded();//encoding secretkey
String skey = Base64.encode(be);
System.out.println("converted secretkey to string:" + skey);
String cipher = new encryption().encrypt(str, secretKey);
System.out.println(str);
//for get extension from given file
String b = fileItem.getName().substring(fileItem.getName().lastIndexOf('.'));
System.out.println("File Extension" + b);
pstm.setBinaryStream(1, fileItem.getInputStream());
pstm.setString(2, keyword);
pstm.setString(3, b);
pstm.setString(4, fname);
pstm.setDate(5, getCurrentDate());
pstm.setString(6, a);
pstm.setLong(7, fileItem.getSize());
pstm.setString(8, cipher);
pstm.setString(9, "0");
pstm.setString(10, skey);
/*Cloud Start*/
File f = new File("C:\\Users\\windows 10\\Desktop" +fileItem.getName());
FileWriter fw = new FileWriter(f);
```

```
fw.write(cipher);
        fw.close();
Ftpcon ftpcon = new Ftpcon();
ftpcon.upload(f, fname);
/*Cloud End*/
int i = pstm.executeUpdate();
if (i == 1) {
response.sendRedirect("owner.jsp?msg1=success");
} else {
response.sendRedirect("owner.jsp?msgg=failed");
}
con.close();
} catch (Exception e) {
out.println(e.toString());
}
}
}
} catch (FileUploadException e) {
} catch (Exception ex) {
Logger.getLogger(Upload.class.getName()).log(Level.SEVERE, null, ex);
}
} finally {
out.close();
}
}
}
private static String getStringFromInputStream(InputStream is) {
BufferedReader br = null;
StringBuilder sb = new StringBuilder();
String line;
        try {
```

```
br = new BufferedReader(new InputStreamReader(is));
while ((line = br.readLine()) != null) {
sb.append(line + "\n");
}
} catch (IOException e) {
} finally {
if (br != null) {
try  {
br.close();
} catch (IOException e) {
}
}
}
return sb.toString();
}
// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the
left to edit the code.">
/**
* Handles the HTTP <code>GET</code> method.
* @param request servlet request
* @param response servlet response
* @throws ServletException if a servlet-specific error occurs
* @throws IOException if an I/O error occurs
*/
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
processRequest(request, response);
```

56

```
/**
```

```
* Handles the HTTP <code>POST</code> method.
*
* @param request servlet request
* @param response servlet response
* @throws ServletException if a servlet-specific error occurs
* @throws IOException if an I/O error occurs
*/
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
processRequest(request, response);
}
/**
* Returns a short description of the servlet.
*
* @return a String containing servlet description
*/
@Override
public String getServletInfo() {
return "Short description";
}// </editor-fold>
        }
```

SAMPLE OUTPUT:

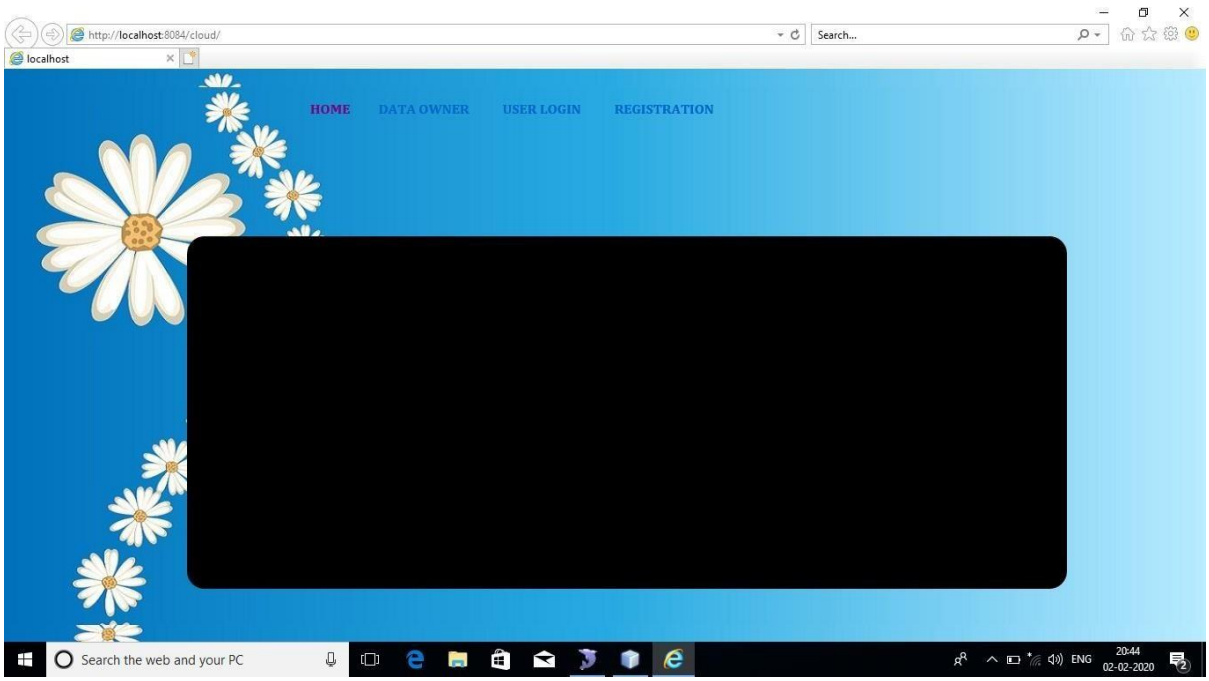


Fig. 7.1 Home

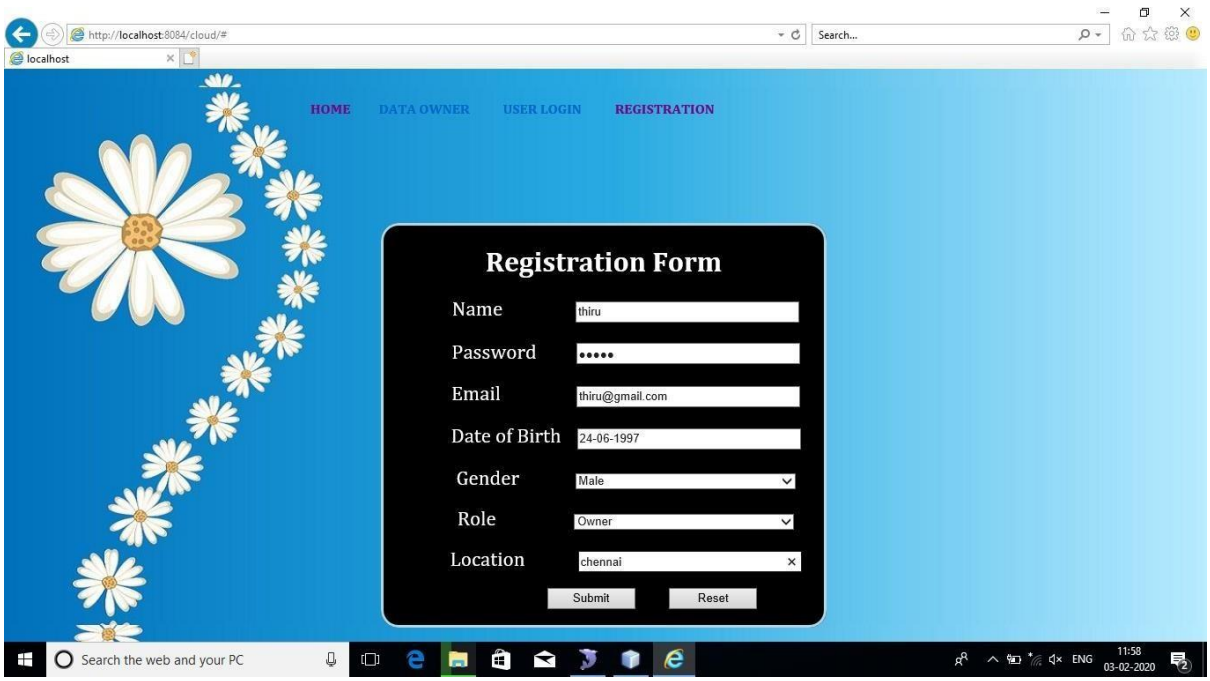


Fig. 7.2 Owner Registration

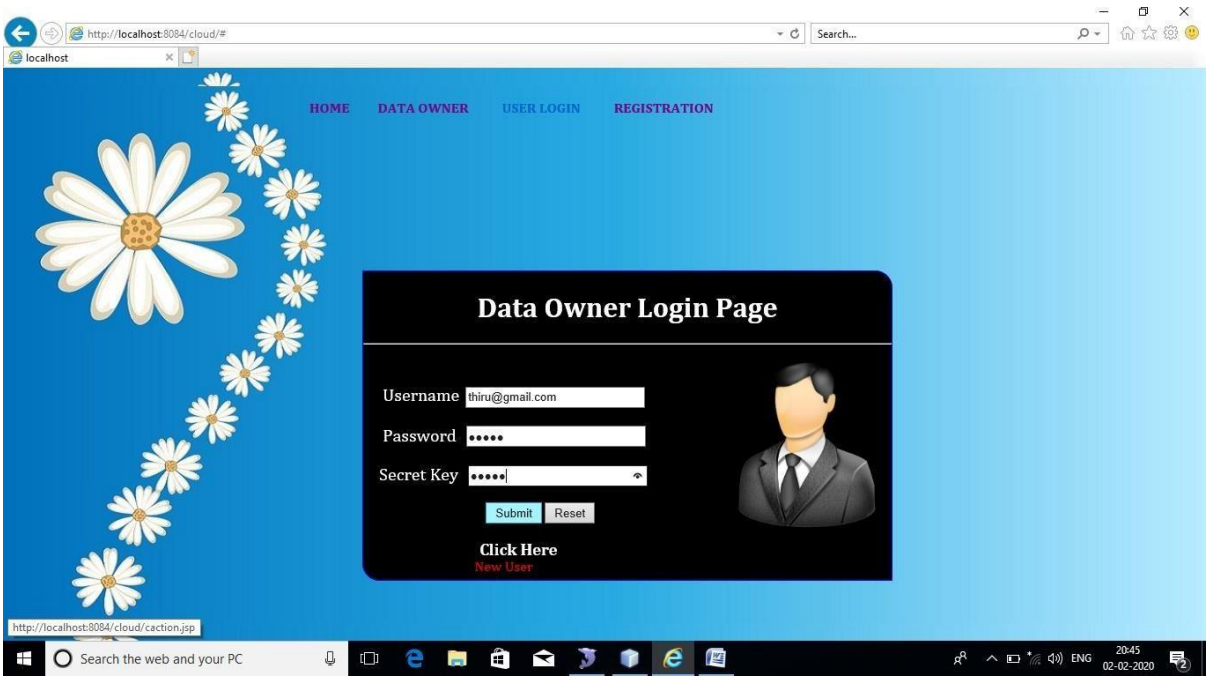


Fig. 7.3. User Registration

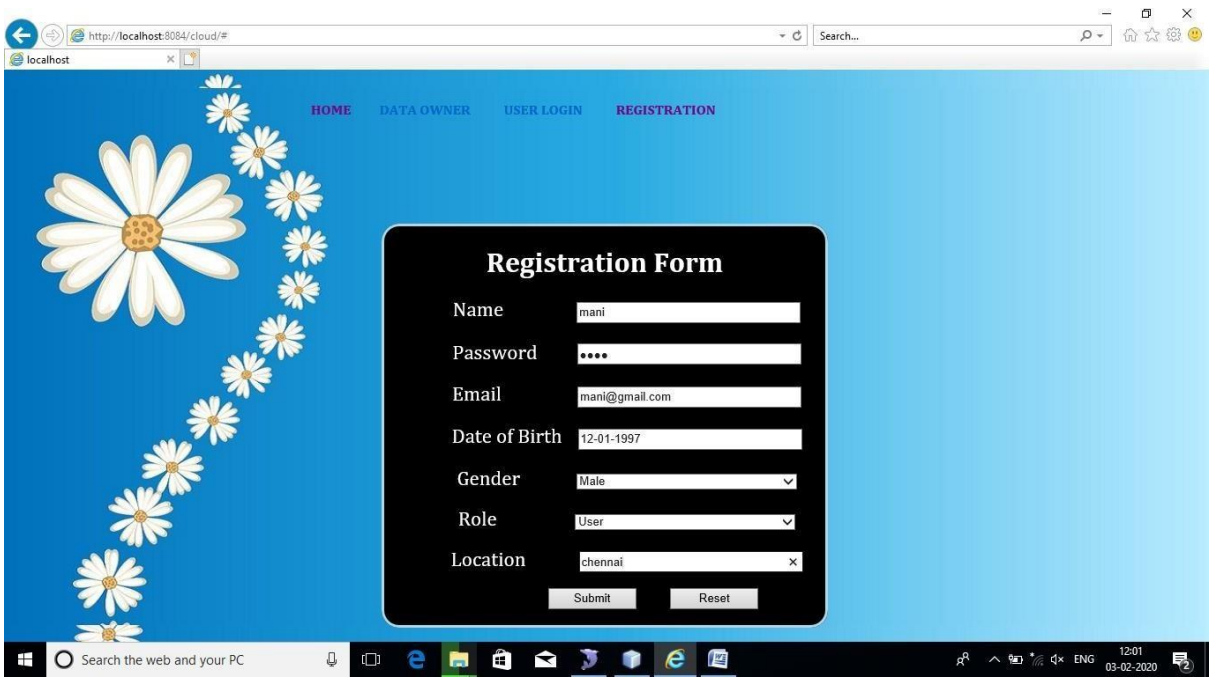


Fig. 7.4. Secret Key Generation

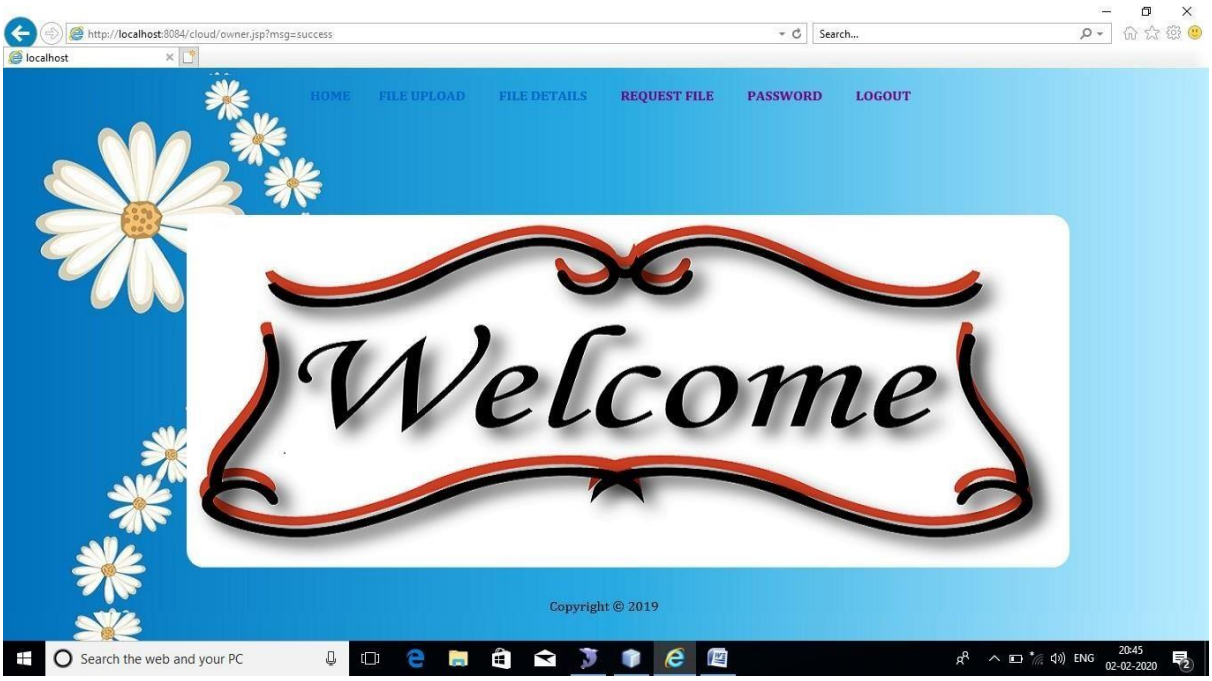


Fig. 7.5.1 File Upload

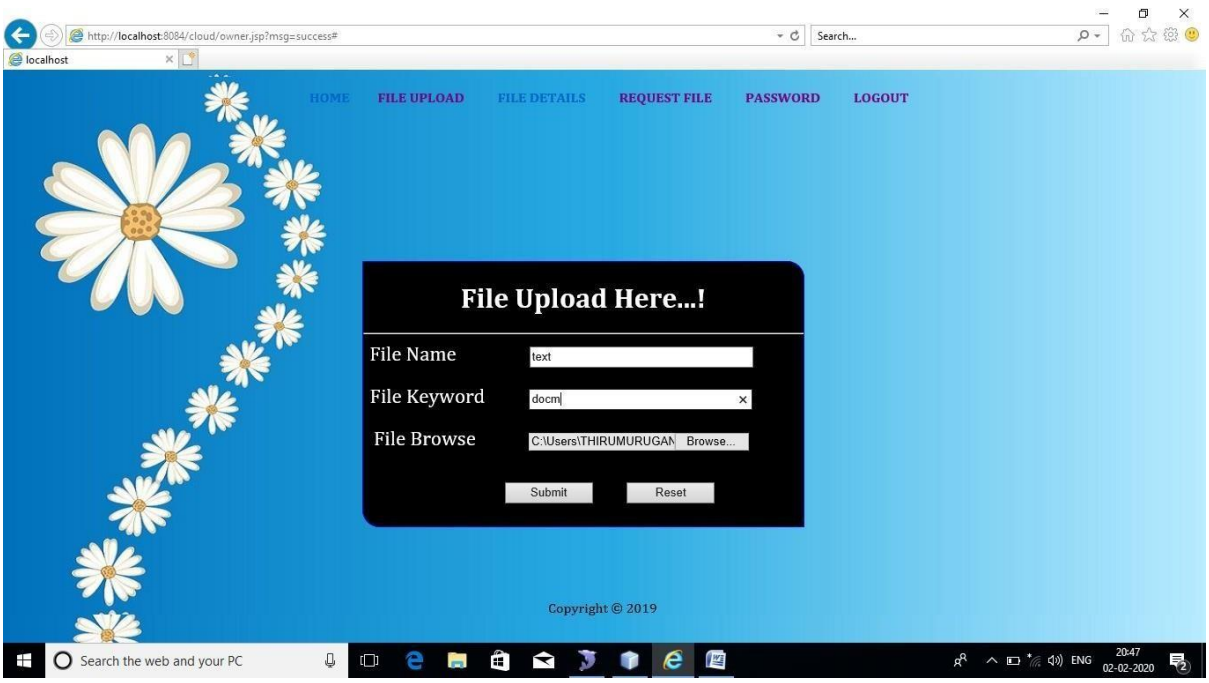


Fig. 7.5.2 File Upload

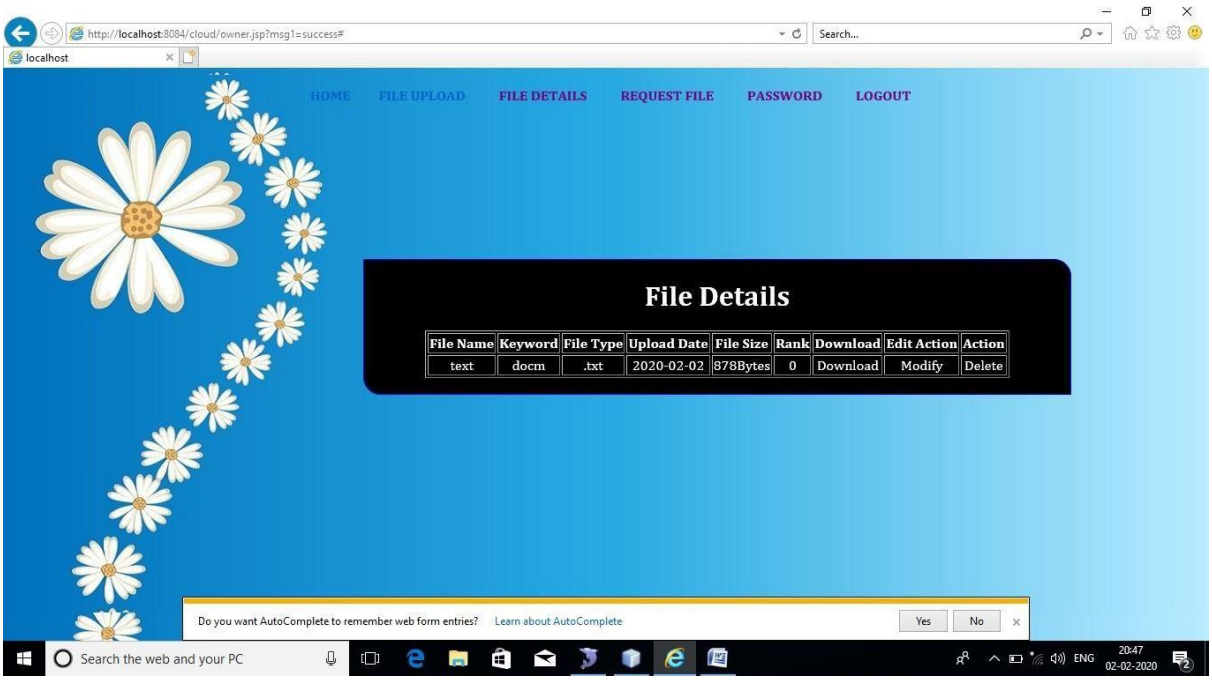


Fig. 7.6. File Details

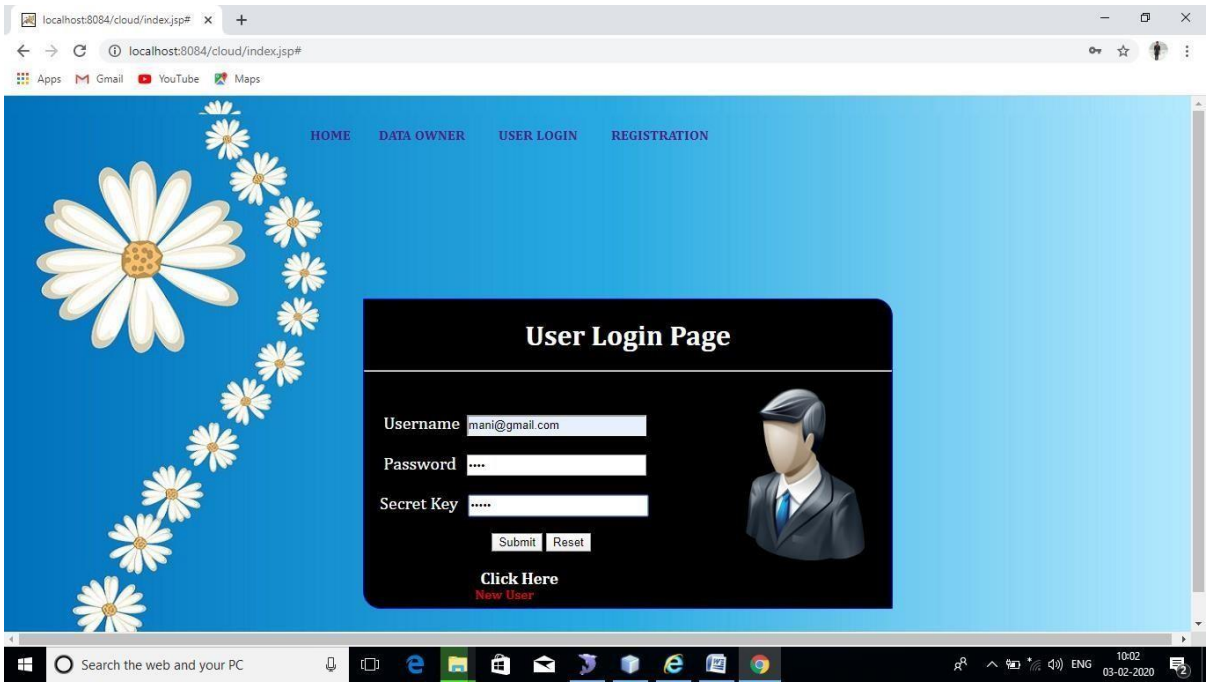


Fig. 7. 7.1. Login

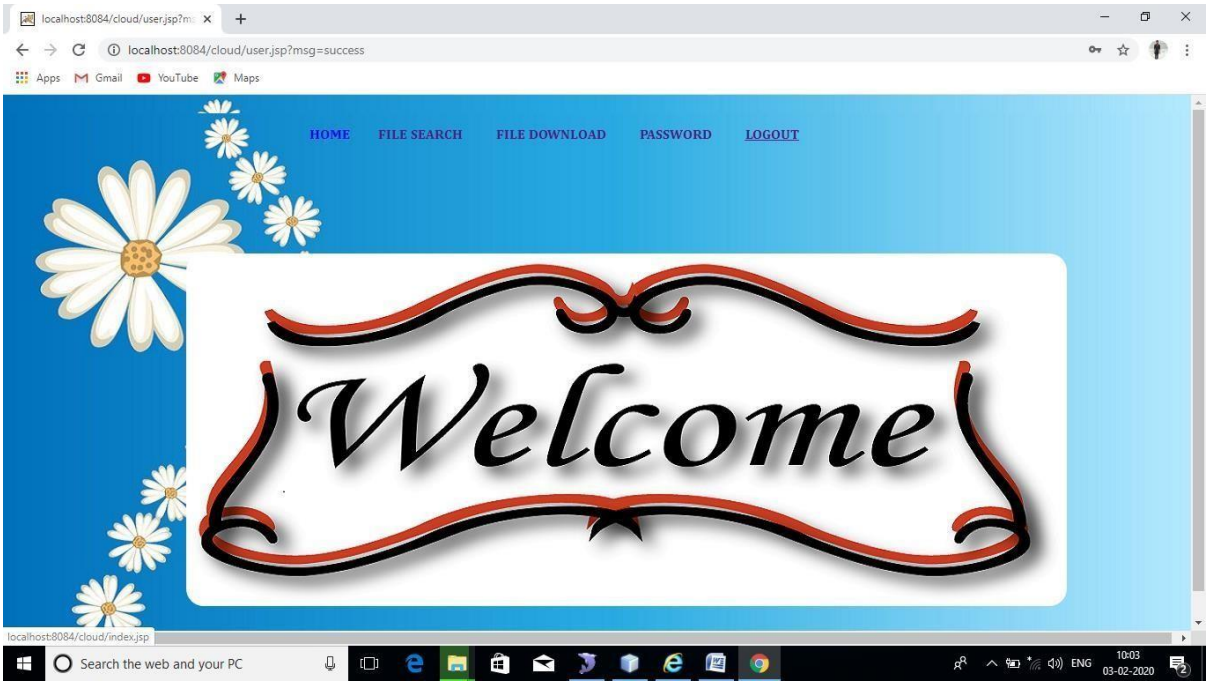


Fig. 7. 7.2. Login

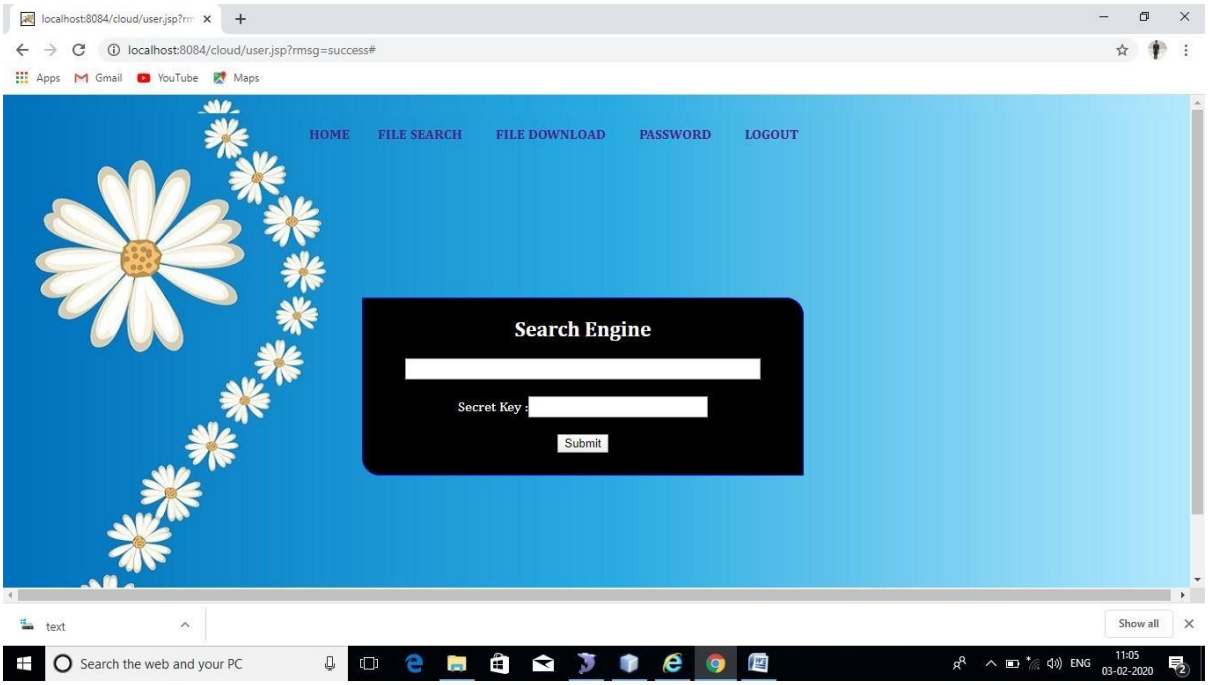


Fig. 7.8. File Search

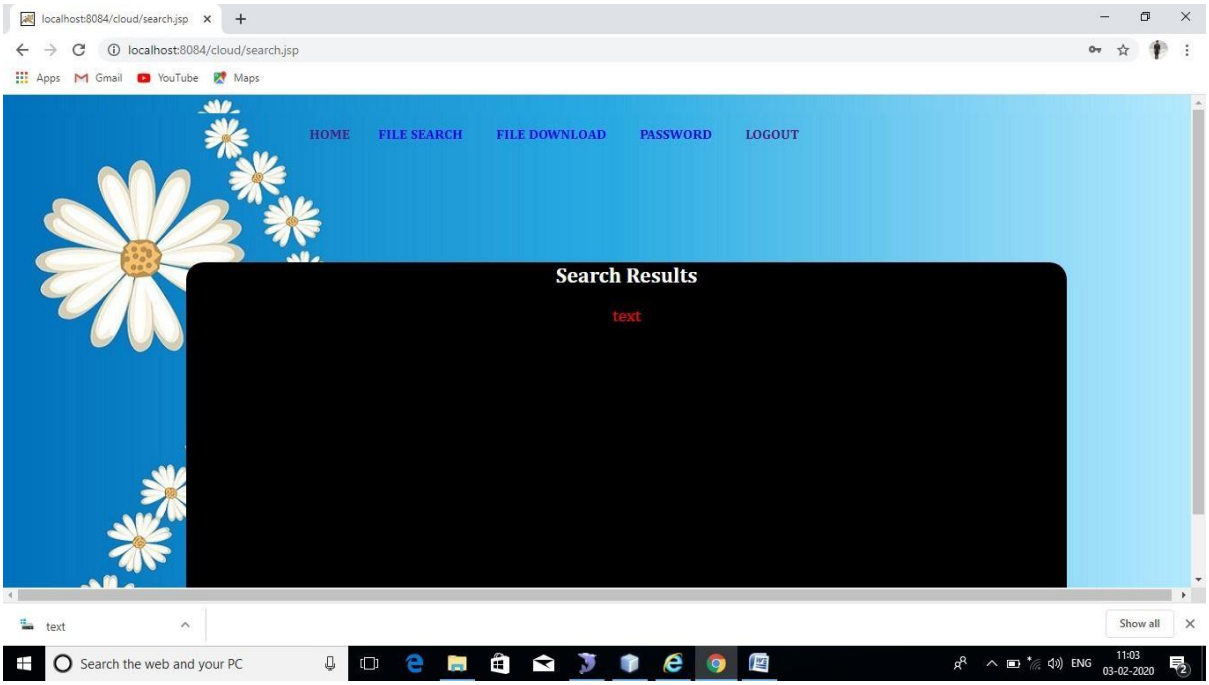


Fig. 7 9. Keyword Search

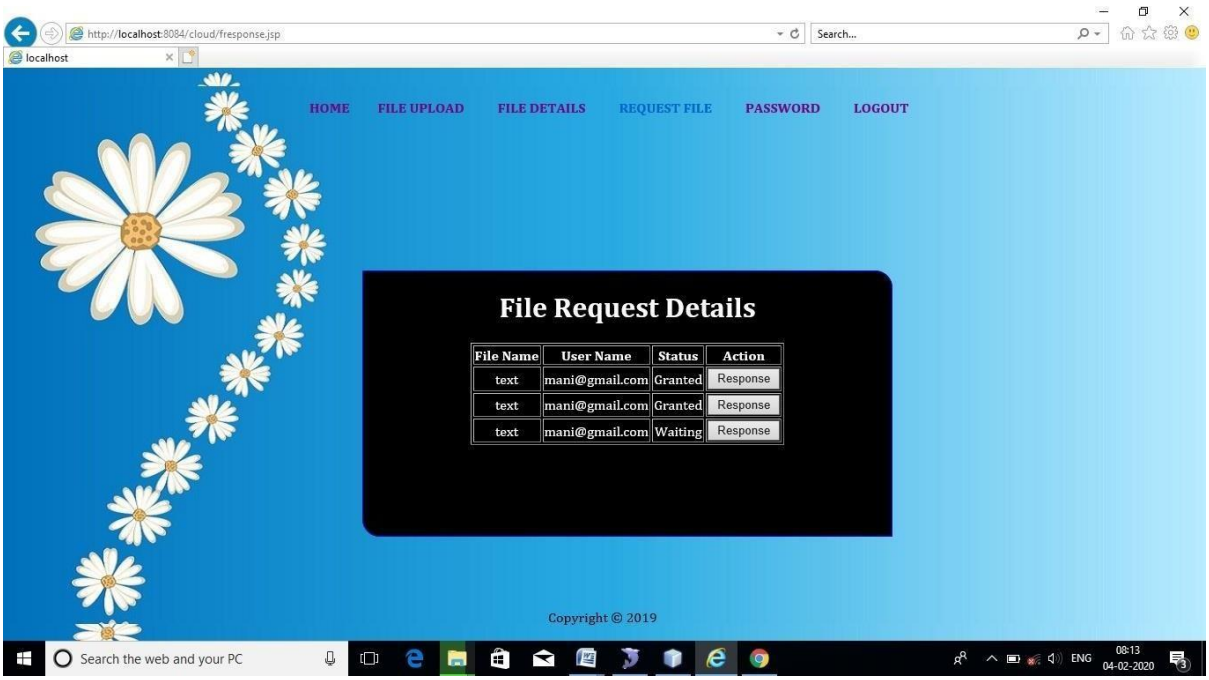


Fig. 7. 10.1. File Request Details

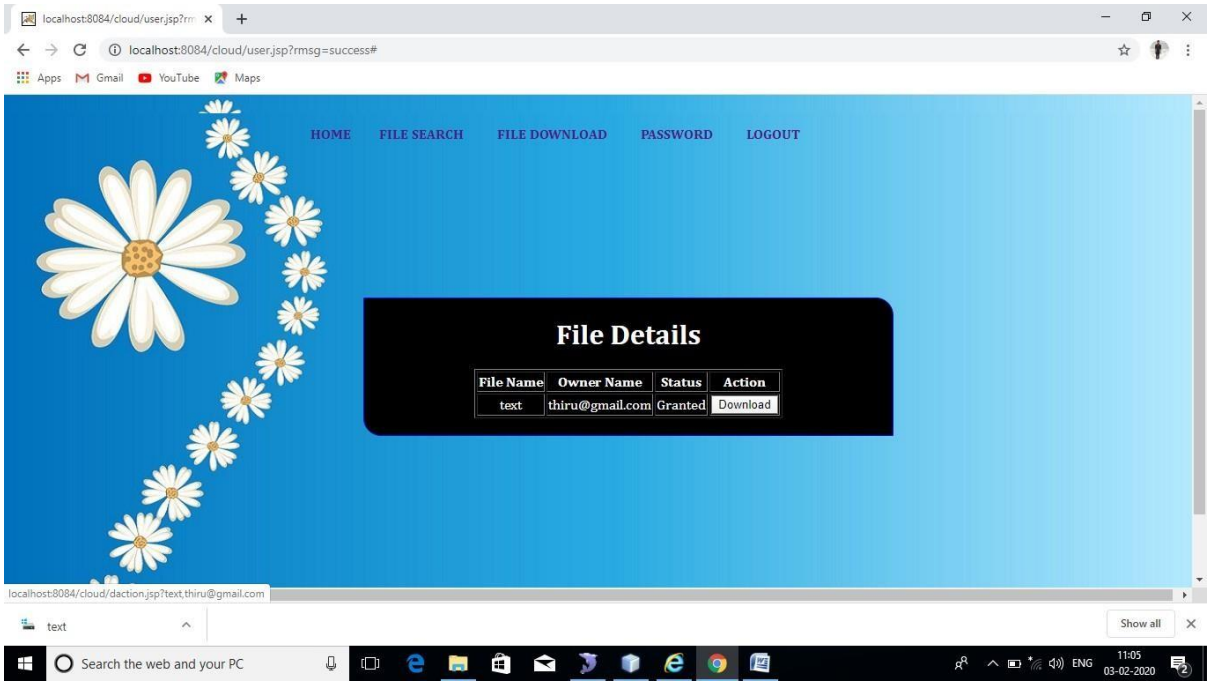


Fig. 7. 10. 2. File Request Details

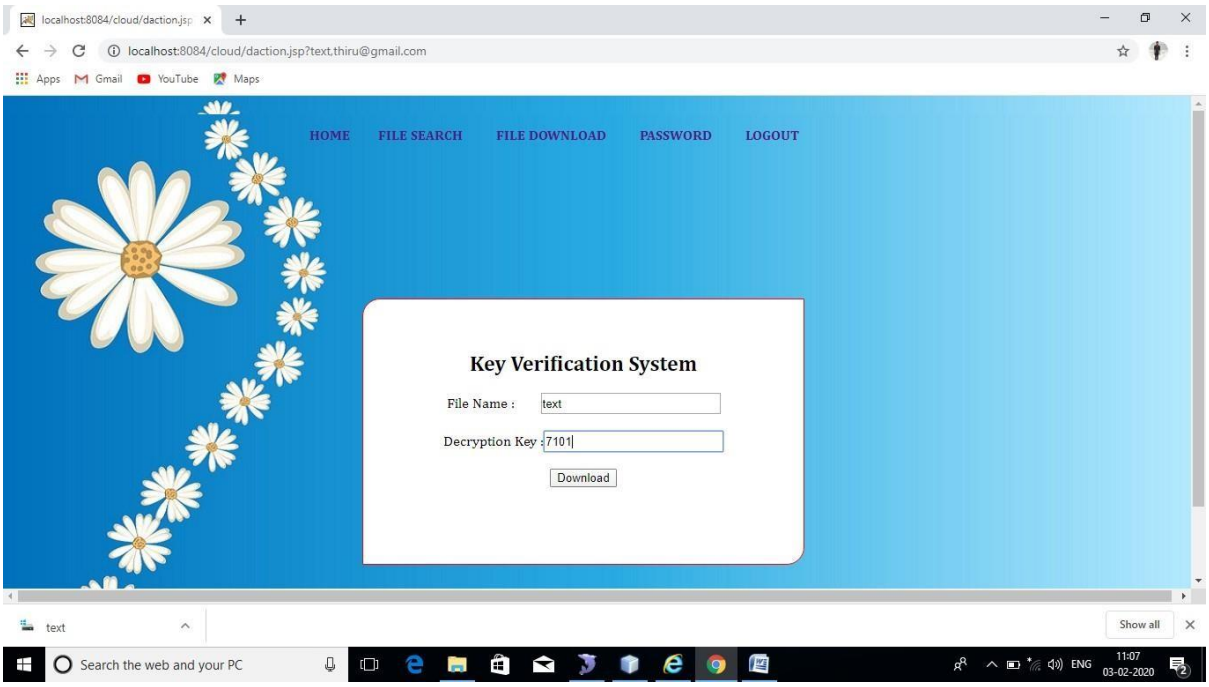


Fig .7. 11. Decryption File

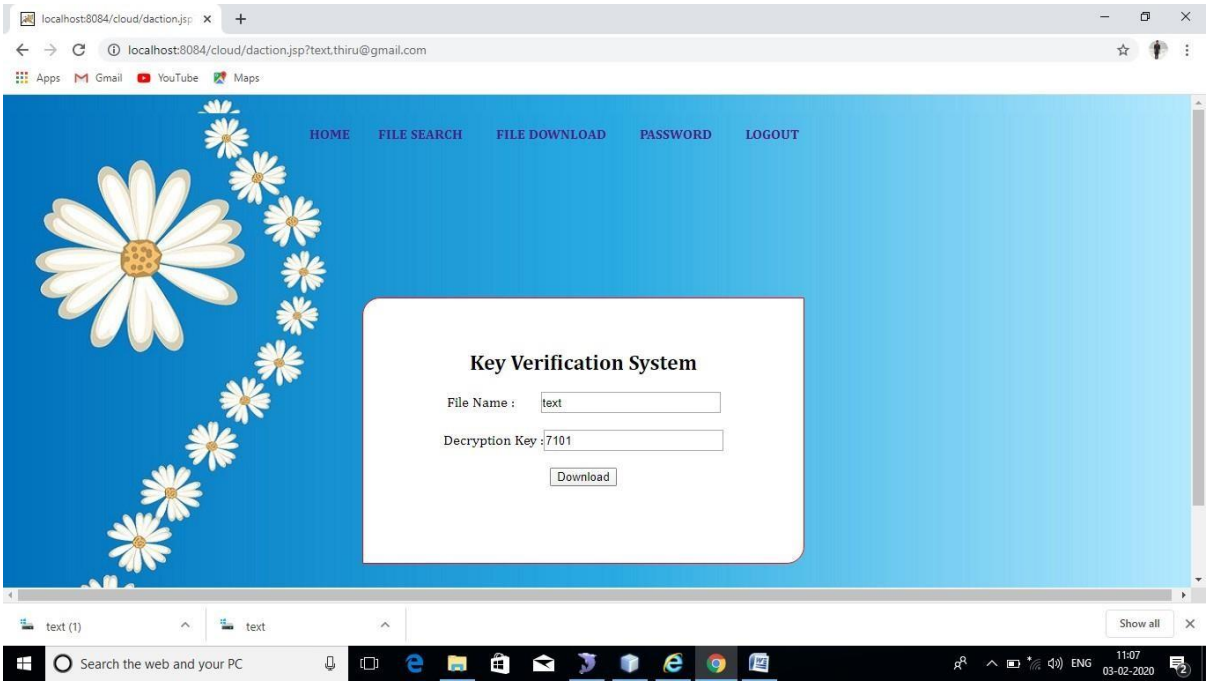


Fig. 7. 12. 1. Key Verification

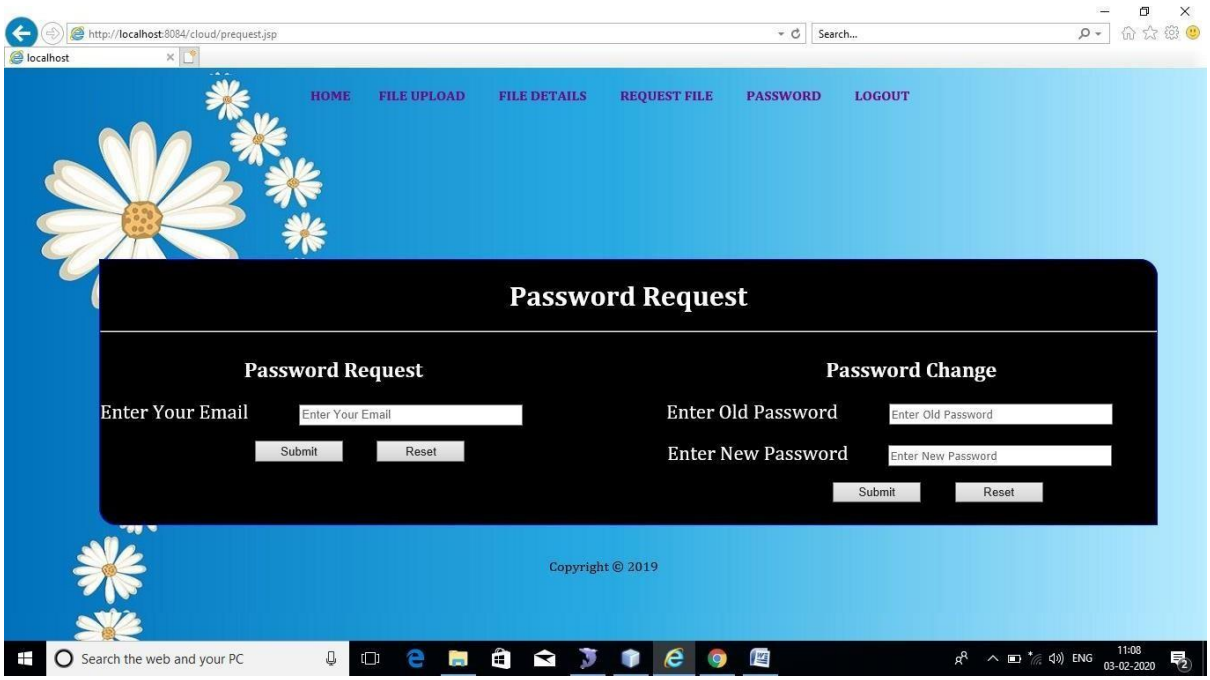


Fig. 7. 12. 2. Key Verification

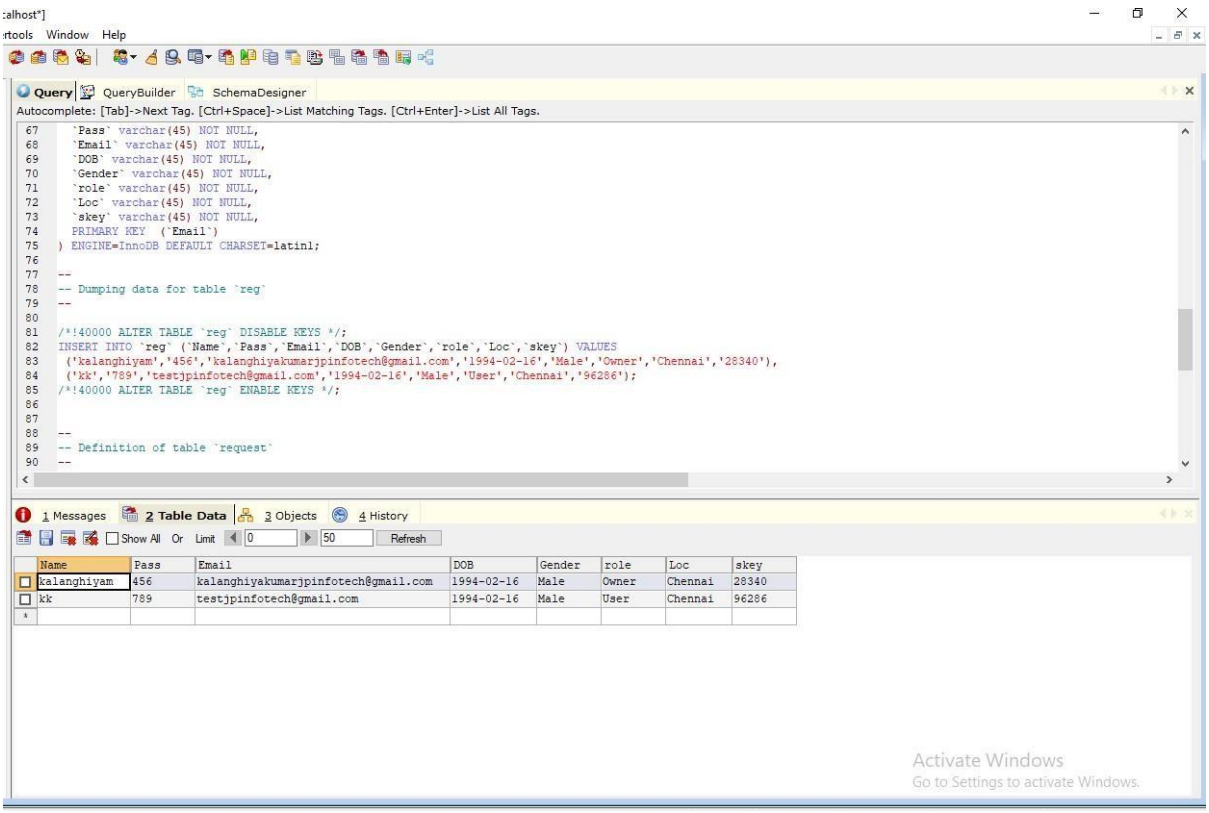


Fig. 7. 13. Auto Secret Key Generation

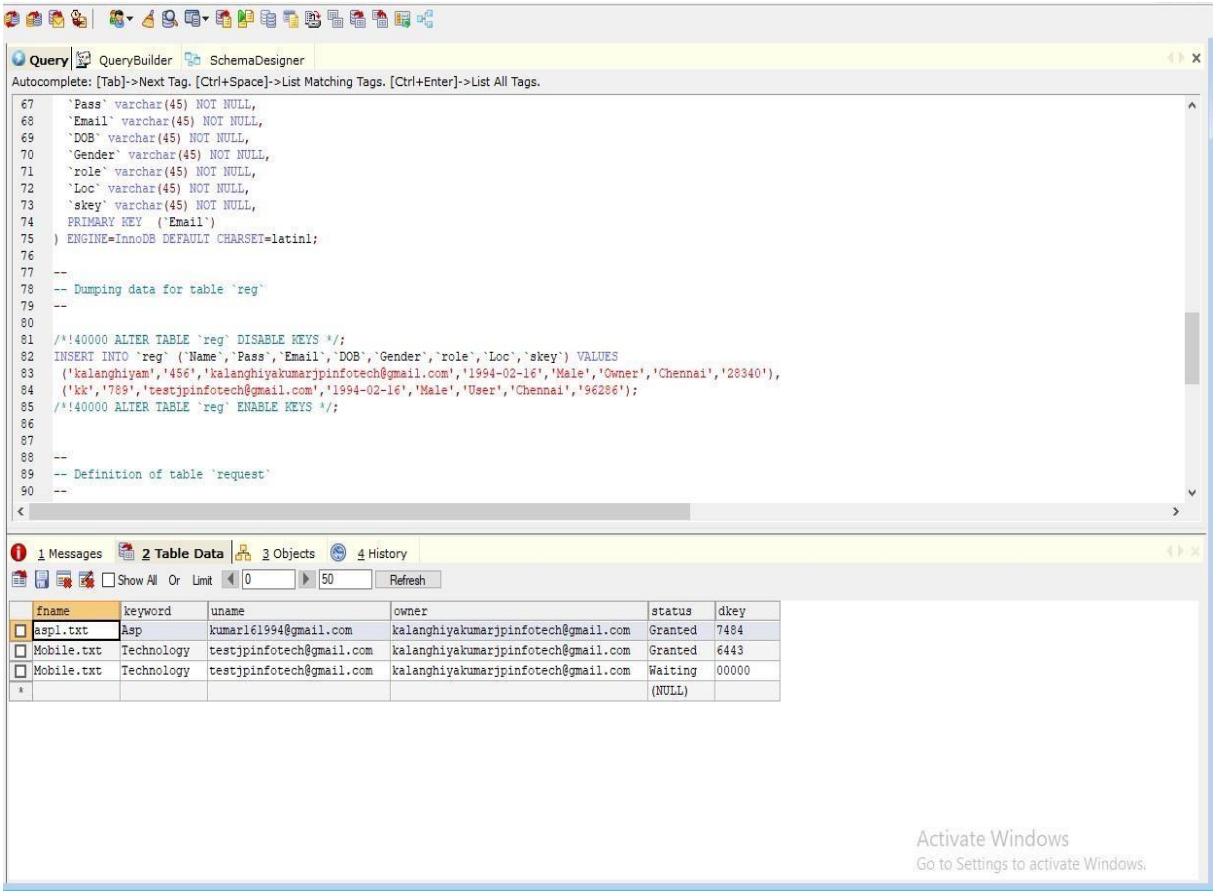


Fig. 7.14. Auto Decryption Key Generation

CHAPTER-8

CONCLUSION & FUTURE ENHANCEMENTS:

The system allow the client to upload their in encrypted form distribute data content to cloud nodes abd ensure data availability using cryptographic techniques. We intoduce a system that leverages block chain technology to provide a secure distributed data storage with keyboard search service. TKSE realizes server-side verifiability which protects honest cloud servers from being framed by malicious data owners in the data storage phase. Furthermore, block chain technologies and hash functions are used to enable payment fairness of search fees without introducing any third party even if the user or the cloud is malicious. Our security analysis and performance evaluation indicate that TKSE is secure and efficient and it is suitable for cloud computing.

CHAPTER-9

REFERENCES:

[1] J. Li, J. Li, X. Chen, C. Jia, and W. Lou, "Identity-based encryption with outsourced revocation in cloud computing," IEEE Transactions on Computers, vol. 64, no. 2, pp. 425–437, 2015

[2] X. Chen, J. Li, X. Huang, J. Ma, and W. Lou, "New publicly verifiable databases with efficient updates," IEEE Transactions on Dependable and Secure Computing, vol. 12, no. 5, pp. 546–556, 2015

[3] H. Li, F. Zhang, J. He, and H. Tian, "A searchable symmetric encryption scheme using block chain," ar Xiv preprint, 2017.[Online]. Available: https: //arxiv.org/pdf/1711.01030.pdf

[4] H. G. Do and W. K. Ng, "Block chain based system for secure data storage with private keyword search," in Services(SERVICES),2017IEEEWorld Congress on. IEEE, 2017, pp. 90–93.

[5] [5] R. Guo, H. Shi, Q. Zhao, and D. Zheng, "Secure attribute-based signature scheme with multiple authorities for block chain in electronic health records systems," IEEE Access, vol. 776,no. 99, pp. 1–12, 2018.

[6] J. A. Akinyele, C. Garman, I. Miers, M. W. Pagano, M. Rushanan, M. Green, and A. D. Rubin, ''Charm: A framework for rapidly prototyping cryptosystems,'' J. Cryptogr. Eng., vol. 3, no. 2, pp. 111–128, Jun. 2013.

[7] S. Agrawal and D. Boneh, ''Homomorphic MACs: MAC-based integrity for network coding,'' in Applied Cryptography and Network Security, (Lecture Notes in Computer Science), vol. 5536. Berlin, Germany: Springer, 2009, pp. 292–305.

[8] G. Ateniese, R. Burns, R. Curtmola, Joseph Herring, L. Kissner, Z. Peterson, and D. Song, ''Provable data possession at untrusted stores,'' in Proc. 14th ACM Conf. Comput. Commun. Secur., Alexandria, VA, USA, 2007, pp. 598–609.

[9] G. Ateniese, S. Kamara, and J. Katz, ''Proofs of storage from homomorphic identification protocols,'' in Advances in Cryptology. Berlin, Germany: Springer, 2009, pp. 319–333.

[10] A. F. Barsoum and M. A. Hasan, ‘‘Provable multicopy dynamic data possession in cloud computing systems,’’ IEEE Trans. Inf. Forensics Security, vol. 10, no. 3, pp. 485–497, Mar. 2015.

[11] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, ‘‘Fog computing and its role in the Internet of Things,’’ in Proc. 1st Ed. MCC Workshop Mobile Cloud Comput., New York, NY, USA, 2012, pp. 13–16.

[12] K. Bowers, A. Juels, and A. Oprea, ‘‘Proofs of retrievability: Theory and implementation,’’ in Proc. ACM Workshop Cloud Comput. Secur., 2009, pp. 43–54.

[13] J. Chang, Y. Ji, M. Xu, and R. Xue, ‘‘General transformations from single-generation to multi-generation for homomorphic message authentication schemes in network coding,’’ Future Gener. Comput. Syst., vol. 91, pp. 416–425, Feb. 2019.

[14] J. Chang et al., ‘‘Secure network coding from secure proof of retrievability,’’ Sci. China Inf. Sci., early access, Oct. 2020.

[15] J. Chang, H. Wang, F. Wang, A. Zhang, and Y. Ji, ‘‘RKA security for identity-based signature scheme,’’ IEEE Access, vol. 8, pp. 17833–17841, 2020.

[16] C. Chu, S. Chow, W. Tzeng, J. Zhou, R. Deng, “Key-aggregate cryptosystem for scalable data sharing in cloud storage,” IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 2, pp.468-477, Feb. 2014.

[17] Y. Tong, J. Sun, S. Chow, P. Li, “Cloud-assisted mobile-access of health data with privacy and auditability”, IEEE Journal of Biomedical and Health Informatics, vol. 18, no. 2, Mar. 2014.

[18] Z. Pervez, A. Khattak, S. Lee, Y. Lee, “SAPDS: Self-healing attributebased privacy aware data sharing in cloud”, The Journal of Supercomputing, vol. 62, no. 1, pp. 431´lC460, Oct. 2012.

[19] C. Fan, V. Huang, H. Rung, “Arbitrary-state attribute-based encryption with dynamic membership”, IEEE Transactions on Computers, vol. 63, no. 8, pp. 1951-1961, Apr. 2013.

[20] D. Boneh, G. Crescenzo, R. Ostrovskyy, G. Persianoz, “Public key encryption with keyword search”, in Eurocrypt 2004, Interlaken, Switzerland, May 2-6, 2004, pp.506-522.

[21] N. Cao, C. Wang, M. Li, K. Ren, W. Lou, “Privacy-preserving multikeyword ranked search over encrypted cloud data”, IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 1, pp. 222-233, Nov. 2013.

[22] S. Seo, M. Nabeel, X. Ding, E. Bertino, “An efficient certificateless encryption for secure data sharing in public clouds”, IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 9, pp. 2107-2119, Sept. 2014.

[23] L.A. Dunning, R. Kresman, “Privacy preserving data sharing with anonymous ID assignment”, IEEE Transactions on Information Forensics and Security, vol. 8, no. 2, pp.402-413, Feb. 2013.

[24] X. Chen, X. Huang, J. Li, J. Ma, D. Wong, W. Lou, “New algorithms for secure outsourcing of large-scale systems of linear equations”, IEEE Transactions on Information and Forensics Security, vol. 10, no. 1, pp. 69- 78, Jan. 2015.

[25] X. Chen, J. Li, J. Weng, J. Ma, W. Lou, “Verifiable computation over large database with incremental updates” IEEE Transactions on Computers, vol. 65, no. 10: 3184-3195, Oct. 2016.

BioGecko

Vol 12 Issue 03 2023
ISSN NO: 2230-5807

Cloud Computing And Secure Keyword-Based Search: A Review

1st Dr.Arulprakash A
Department of Computer Science and Engineering
Bharath Institute of Higher Education and Research
Chennai, India
Prakash875@gmail.com

4th Rohith.G
Department of Computer Science and Engineering
Bharath Institute of Higher Education and Research
Chennai, India
rohithg302002@gmail.com

2nd Karthikeyan.K
Department of Computer Science and Engineering
Bharath Institute of Higher Education and Research
Chennai, India
karthicthalapathy0@gmail.com

5th Sai Vignesh
Department of Computer Science and Engineering
Bharath Institute of Higher Education and Research
Chennai, India
saivignesh2203@gmail.com

3rd Kasam Shree Veera Hanuman Reddy
Department of Computer Science and Engineering
Bharath Institute of Higher Education and Research
Chennai, India
kasamhanuman2001@gmail.com

ABSTRACT:

A system that combines Block chain technology is used for secure distributed data storage and a keyword search service. Users can upload encrypted files, the system distributes them across cloud nodes, and it makes use of cryptographic techniques to ensure data availability. The cloud server is also regarded as trustworthy. We first highlight the significance of maintaining the data in a public chain before providing a block chain-based solution for secure distributed data storage with keyword search functionality. We suggest a solution that combines block chain technology-based keyword search with secure distributed data storage. The technology makes it possible for users to upload data in an encrypted format, distributes it across cloud nodes, and guarantees data availability. Takes Exam offers server-side verifiability, preventing and it is imperative that dishonest data owners refrain from using dishonest cloud servers as props throughout the storage phase of the data lifecycle. Additionally, without the use of third parties, block chain technologies and hash functions enabled equitable payment of search fees. Regardless of TKSE is sufficiently secure and efficient to be used for cloud computing according to our security and performance analysis. This system possesses the main objective of being secure and efficient, additionally; our performance assessment and security research show that it may be employed in cloud computing systems.

Keywords: *searchable symmetric encryption schemes (SSE), block chain technology, trusted third party (TTP), ECDSA stands for Elliptic Curve Digital Signature Algorithm.TKSE, cloud computing.*

1. INTRODUCTION:

Cloud computing technologies have advanced quickly in recent years, and a number of research has been done on cloud computing security challenges, specifically access control and privacy protection [1]. Cloud storage requires both search functionality and data security as a typical cloud computing service. User-side verifiability takes harmful cloud server potential into serious consideration, which means it might purposefully produce inaccurate results or just return a portion of the search results [2]. The first topic covered in is user-side verifiability. However, without a reliable third party, these two systems are unable to offer server-side verifiability and fair remuneration. Server-side verifiability also considers the possibility of unscrupulous data owners, who can purposefully outsource inaccurate data during the data storage phase and then falsely seek recompense afterwards [3]. This issue has not been addressed, and even in the literature, it has not gotten much attention. Not least among other things, the majority of the earlier programmers rely on banks. In particular either the usual traditional payment system is utilized or it is necessary to install a reputable bank is an example of a trusted third party (TTP).to ensure payment fairness because The payment issue is not considered [4]. Fair payment procedures can motivate cloud servers and users to behave honestly [5]. Whatever the cloud server (or data owner) performs, if a detrimental behavior is identified [6]. If malicious activity is found based on user-side verifiability, the data owner (i.e. cloud server) will be notified immediately.) Will be adequately reimbursed (resp. server-side verifiability). This makes SSE's responsibility to make fair payments without the help of a third party a big and challenging one. We our performance review demonstrates the efficacy of TKSE and demonstrates its security. The following ideal characteristics best describe TKSE in particular. Searching for Keywords in Encrypted Data [8]. The Elliptic Curve Digital Signature Algorithm (ECDSA)-based encrypted data index enables users to browse through the encrypted material that has been outsourced. Verifiable by the user [9] In TKSE, a data owner can add search criteria to a joint transaction's output script in order to ensure that, and only in the event that, the script evaluates to true base on their turned search result, the data owner will receive the joint transaction's results. The cloud server may redeem the transaction. TKSE is used to achieve user-side verifiability as a result, and the data owner is able to fend against adversarial cloud servers [10]. On the server side, verifiable, similar to user-side verifiability, verifiability on the server is accomplished by the cloud server by recognizing fraudulent data owners. Just Compensation and No TTP block can be used without adding any TTP. Chain allow for a fair payment system in TKSE. Prior to outsourcing cloud computing, data must be encrypted because cloud servers are unreliable and users' data privacy must be protected [12]. Users are able to upload encrypted data, distribute it among cloud nodes, and employ cryptographic techniques to ensure data availability. Additionally The project's use of cloud computing and analysis based on search and data sharing to provide security is novel [13]. According to Our performance evaluation and security analysis show that TKSE is appropriate for cloud computing since it is both secure and effective. [14]. we provide a solution for Utilizing block chain technology, Through the implementation of server-side verifiability, TKSE safeguards trustworthy preventing malicious data owners from using cloud servers as props throughout the data storage phase. The system's social impact is to ensure data security and protect user privacy. The technology also decreases the expense of data management [15]. It offers message authentication, ensuring the security of data exchange. It also resolves the issue of integrity protection.

2. METHODOLOGY:

2.1. MODULES

- Login
- Registration
- Create Secrete Key
- Authentication Scheme
- Two-Side Verification

2.1.2. Login

Several websites, computer programs, and mobile applications demand a login. Security measures are put in place to guard against unauthorized access to confidential data. Access is denied to the user in the event that a login attempt is unsuccessful (i.e., the entered username and password do not correspond to an existent user account). Many systems stop users from even attempting to log in when they repeatedly fail.

2.1.3 Registration

A registered user is someone who has signed up for an account on a website, piece of software, or other platform in the past. The procedure through which signed-up users provide the system with credentials (such as a username, email address, and password to confirm their identity) is known as logging in. The majority of systems designed for public use enable any user to sign up by simply choosing a register or sign up function and entering these credentials for the first time. Additional rights may be granted to registered users over those given to unregistered users.

2.1.3. Create Secrete Key

Through the application of cryptography, secure communication in the presence of third parties is practiced and explored. The primary goal of cryptography in the past was encryption. When information is encrypted, it is converted from plain text to cypher text. Decryption is carried out backwards. By utilizing encryption, information can be kept secret from all parties except the intended recipients. Encryption and decryption are made possible by the cypher algorithm pair. The algorithm and the key determine how the cypher functions. The secret that communicators know is the solution.

2.1.4. Authentication scheme

It is used to address the difficulty of examining the person's keys (let's say "person B") that someone else ("person A") engages in conversation with or makes an effort to do so. To put it another way, it is the procedure used to ensure that the key held by "person B" belongs to "person A" and vice versa. Although other algorithms disclose the keys at the time of authentication as well, this is often done after the keys have been transferred between the two sides over some secure channel. The easiest way to solve this issue On the other hand, in systems with a sizable user base or in this is not practicable for situations where the users do not directly know one another (like online purchasing). To address this issue, a variety of symmetric key and asymmetric public key techniques are available.

2.1.5. Two-Side Verification

In this module, as shown in below figure 1:

In order to confirm that the person or entity requesting access is who or what they claim to be, two authentication methods must be used sequentially. This procedure is known as two-side verification.

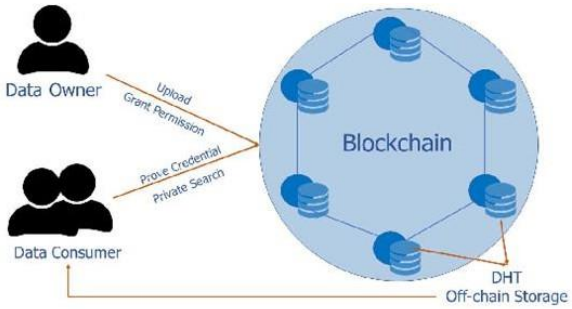


Figure 1: Two side Authentication method

Our performance review demonstrates the efficacy of TKSE and demonstrates its security. The following desirable characteristics best describe TKSE.

• Keyword using the outsourced encrypted data can be searched through by the user. Elliptical Curve Digital Signature Algorithm (ECDSA)-based encrypted data index.

• Verifiability on the user's side. The output script of a joint transaction can have search criteria included in it so that the cloud server can only redeem it if and when using the returned search result as a basis, the output script evaluates to true. TKSE as a result makes user-side verifiability possible and equips the data owner to fight off hostile cloud services.

• Server-side code that is verifiable. Server-side verifiability is accomplished by the cloud server by recognizing fraudulent data owners. Fair Recompense, lack of TTP due to its dependence on hash functions, and block can be used without adding any TTP. Chain allow for a fair payment system in TKSE.

2.1.6 Secure Hashing Algorithm

A 160-bit (20-byte) hash value is produced using the cryptographic hash function of Secure Hash Algorithm 1 (SHA-1). The "messages digest" is the term used to describe this hash value. This message digest frequently produces a 40-digit hexadecimal number. It was created by the US National Security Agency and is a Federal Information Processing Standard. The Java package's Message Digest Class is used to compute cryptographic hashing values security.

Algorithm for Pseudo-Random Number Generator

- Accept a seed or key input number as the first step.
- In step 2 uses that seed to prompt a series of calculations to produce the outcome. The random number is that outcome.
- Use the generated random number as the seed for the subsequent iteration in step three.
- Repeat the procedure in Step 4 to simulate randomness.

3. RESULT AND DISCUSSION:

3.1.1. Existing System

Furthermore, even if the cloud or user is evil, block chain technology. Since Based on digital signature technology offered by TKSE, the encrypted data index, a user can search through Verify that the cloud's search results match your defined criteria by accessing encrypted data. Our analysis the reliability and security of TKSE are demonstrated by its performance and security, making it appropriate for cloud computing. On the basis of our initial SSE technique with additionally, user-side verifiability has attained ide verifiability. Additionally, without adding any TTP, fair payment is accomplished via block chain technologies and hashing. The Drawbacks of Data confidentiality and privacy has been achieved but identity privacy neglected. It has Less safety. It does not include any data.

3.1.2. Proposed System

Because the terms for Searchable encryption techniques have been created in the symmetric key configuration and the user scenario, two examples. And CSP must agree on the search costs' redemption, which calls for the MAC secret key. This prevents the concept from being immediately integrated with block chain technologies. Cryptography hash function server considered a key component is a digital signature is one example of a security application or protocol that uses this aspect of information security. The creation of MAC signature schemes and random number generation are two techniques for safeguarding the integrity of data and authenticating its provenance. Many different applications, such as databases, computer vision, and the storage of passwords, use hashing algorithms. Benefits of the suggested system save money on data management. To protect the confidentiality and security of user data. Code for message authentication and protection of integrity we suggest a reliable The TKSE keyword search method uses encrypted data. It's necessary to involve a third party, to fully handle the aforementioned difficult concerns in cloud computing.

3.1.3 Functional Requirements

It is necessary for the technological requirements of the software products. The functional, performance, and security requirements for particular software systems are included in this phase of the requirement analysis process

[16]. Performance of the system is mostly determined by the high-quality hardware used to run the programmer with the necessary capabilities.

Usability

It explains how user-friendly a system must be. Short or long questions can be asked with ease because the Porter stemming algorithm prompts the user's chosen response [17]-[23].

Robustness

It describes a programmer that operates effectively both in typical and unexpected circumstances. It is the user's capacity to handle execution faults for pointless requests.

Security

Security is the condition of allowing restricted access to resources. Unauthorized users cannot access the system because of the system's strong security measures.

Reliability

It is the likelihood of how frequently the software malfunctions. MTBF is a common unit of measurement (The average time between failures). To ensure that procedures are completed completely and without interruption, the requirement is required. It is capable of supporting any weight, enduring indefinitely, and even overcoming failures.

Compatibility

The version above all web browsers supports it. Any web server, including a local host, can be used to make the system real-time.

Flexibility

The project's adaptability is set up in a way that allows it to function in many surroundings while being used by various users.

Safety

Safety is a precaution done to avoid problems. Each enquiry is handled securely without revealing any personal information to third parties.

3.1.4 Non- Functional Requirements

Portability

The usability of the same software in various settings. Any operating system can be used to run the project.

Performance

The software's adaptability to different settings. The project can be executed on any operating system.

Accuracy

The information is retrieved quickly and with great accuracy thanks to the requesting query. The system offers a high level of security that is both efficient and reliable.

Maintainability

The project is straightforward since it is simple to make updates without compromising its stability. In essence, maintainability refers to how simple it is to maintain the system. It refers to how simple it is to test software, analyze data, make changes, and maintain systems. This project is easily maintainable because new modifications may be made without negatively impacting its stability.

4. CONCLUSION

We propose a system that combines secure utilizing block chain technology and the service provides keyword search capabilities with distributed data storage. The system distributes content to cloud nodes, permits users to submit data in encrypted form, and makes use of cryptographic methods to guarantee data availability. By achieving server-side verifiability, it stops dishonest data owners from using legitimate cloud servers as pawns. Additionally, Block chain technology and hash functions enable the payment Regardless of whether the user or cloud is malicious; search fees can be collected without the involvement of third parties. The results of our analysis of TKSE's effectiveness and security demonstrate that it is both, making it suitable for cloud computing.

REFERENCES:

[1] J. Li, J. Li, X. Chen, C. Jia, and W. Lou, "Identity-based encryption with outsourced revocation in cloud computing," IEEE Transactions on Computers, vol. 64, no. 2, pp. 425–437, 2015

[2] X. Chen, J. Li, X. Huang, J. Ma, and W. Lou, "New publicly verifiable databases with efficient updates," IEEE Transactions on Dependable and Secure Computing, vol. 12, no. 5, pp. 546–556, 2015

[3] H. Li, F. Zhang, J. He, and H. Tian, "A searchable symmetric encryption scheme using block chain, "arXivpreprint, 2017. [Online].Available: https: //arxiv.org/pdf/1711.01030.pdf

[4] H.G.DoandW.K.Ng,"Blockchainbasedsystemforsecuredatastoragewithprivatekeywordsearch,"inServices (SERVICES), 2017IEEEWorld Congress on. IEEE, 2017, pp. 90–93.

[5] R. K. Dhanaraj, L. Krishnasamy, O. Geman and D. R. Izdrui, "Black hole and sink hole attack detection in wireless body area networks," Computers, Materials & Continua, vol. 68, no.2, pp. 1949–1965, 2021. doi:10.32604/cmc.2021.015363

[6] Ramakrishnan, V., Chenniappan, P., Dhanaraj, R. K., Hsu, C.-H., Xiao, Y., & Al-Turjman, F. (2021). Bootstrap aggregative mean shift clustering for big data anti-pattern detection analytics in 5G/6G communication networks. In Computers & Electrical Engineering (Vol. 95, p. 107380). Elsevier BV. https://doi.org/10.1016/j.compeleceng.2021.107380

[7] Chandraprabha, M., & Dhanaraj, R. K. (2020, November 5). Machine learning based Pedantic Analysis of Predictive Algorithms in Crop Yield Management. 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA). 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA). https://doi.org/10.1109/iceca49313.2020.9297544

[8] Sathyamoorthy, M., Kuppusamy, S., Dhanaraj, R.K. et al. Improved K-Means Based Q Learning Algorithm for Optimal Clustering and Node Balancing in WSN. Wireless Pers Commun 122, 2745–2766 (2022). https://doi.org/10.1007/s11277-021-09028-4

[9] Rajesh Kumar D, & ManjupPriya S. (2013, December). Cloud based M-Healthcare emergency using SPOC. 2013 Fifth International Conference on Advanced Computing (ICoAC). 2013 Fifth International Conference on Advanced Computing (ICoAC). https://doi.org/10.1109/icoac.2013.6921965

[10] Rajesh Kumar Dhanaraj, Lalitha Krishnasamy et al Black-Hole Attack Mitigation in Medical Sensor Networks using the Enhanced Gravitational Search Algorithm, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems. https://doi.org/10.1142/S021848852140016X

[11] Lalitha, K., Kumar, D. R., Poongodi, C., & Arumugam, J. (2021). Healthcare Internet of Things – The Role of Communication Tools and Technologies. In Blockchain, Internet of Things, and Artificial Intelligence (pp. 331–348). Chapman and Hall/CRC. https://doi.org/10.1201/9780429352898-17

[12] Dhanaraj, R. K., Rajkumar, K., & Hariharan, U. (2020). Enterprise IoT Modeling: Supervised, Unsupervised, and Reinforcement Learning. In Business Intelligence for Enterprise Internet of Things (pp. 55–79). Springer International Publishing. https://doi.org/10.1007/978-3-030-44407-5_3

[13] Sathish, R., & Kumar, D. R. (2013, March). Proficient algorithms for replication attack detection in Wireless Sensor Networks—A survey. In 2013 IEEE International Conference ON Emerging Trends in Computing, Communication and Nanotechnology (ICECCN) (pp. 1-7). IEEE.

[14] Sathya, K., & Kumar, D. R. (2012, February). Energy efficient clustering in sensor networks using Cluster Manager. 2012 International Conference on Computing, Communication and Applications. 2012 International Conference on Computing, Communication and Applications (ICCCA). https://doi.org/10.1109/iccca.2012.6179177

[15] Prasanth, T., Gunasekaran, M., & Kumar, D. R. (2018, December). Big data Applications on Health Care. 2018 4th International Conference on Computing Communication and Automation (ICCCA). 2018 4th International Conference on Computing Communication and Automation (ICCCA). https://doi.org/10.1109/ccaa.2018.8777586

[16] Ali, M., Dhanaraj, R.K. (2023). IoT and Blockchain Oriented Gender Determination of Bangladeshi Populations. In: Santosh, K., Goyal, A., Aouada, D., Makkar, A., Chiang, YY., Singh, S.K. (eds) Recent

Trends in Image Processing and Pattern Recognition. RTIP2R 2022. Communications in Computer and Information Science, vol 1704. Springer, Cham. https://doi.org/10.1007/978-3-031-23599-3_25

[17] Rajesh, E., Basheer, S., Dhanaraj, R. K., Yadav, S., Kadry, S., Khan, M. A., Kim, Y. J., & Cha, J.-H. (2022). Machine Learning for Online Automatic Prediction of Common Disease Attributes Using Never-Ending Image Learner. In Diagnostics (Vol. 13, Issue 1, p. 95). MDPI AG. https://doi.org/10.3390/diagnostics13010095

[18] V. Juyal, Nitin Pandey and Ravish Saggar, "Opportunistic message forwarding in self organized cluster based DTN,"" 2017 International Conference on Infocom Technologies and Unmanned Systems (Trends and Future Directions) (ICTUS), Dubai, 2017, pp. 497-502. doi: 10.1109/ICTUS.2017.8286060"

[19] V. Juyal, Nitin Pandey and Ravish Saggar, "Performance comparison of DTN multicasting routing algorithms-opportunities and challenges,"" 2017 International Conference on Intelligent Sustainable Systems (ICISS), Palladam, 2017, pp. 53-57. doi:10.1109/ISS1.2017.8389238"

[20] V. Juyal, Nitin Pandey and Ravish Saggar, "An anatomy on routing in delay tolerant network," 2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), Chennai, 2016, pp. 1-4. doi: 10.1109/ICCIC.2016.7919724

[21] V. Juyal, Nitin Pandey and Ravish Saggar, "A heuristic lightweight security algorithm for resource constrained DTN routing," 2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), Chennai, 2016, pp. 1-4. doi: 10.1109/ICCIC.2016.7919695

[22] V. Juyal, Nitin Pandey and Ravish Saggar, "Impact of varying buffer space for routing protocols in delay tolerant networks," 2016 International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, 2016, pp. 2152- 2156. doi:10.1109/ICCSP.2016.7754562"

[23] V. Juyal, Ajay Vikram Singh and Ravish Saggar, "Message Multicasting in Near-Real Time Routing for Delay/Disruption Tolerant Network,"" 2015 IEEE International Conference on Computational Intelligence & Communication Technology, Ghaziabad, 2015, pp. 385-390. doi: 10.1109/CICT.2015.79."