**Certificateless public integrity checking of group shared data on cloud storage**

A Project Report Submitted in partial

fulfillment of the Requirements for

the award of the degree of

**MASTER OF COMPUTER APPLICATION**

**In**

**DEPARTMENT OF COMPUTER APPLICATION**

**By**

# ` ROHIT VARMA.M

# (21H71F0046)

**Under the esteemed guidance of**

**Mr.Vamsi.S**

**ASSISTANT PROFESSOR**



**Department of Computer Application**

**DVR & Dr. HS**

**MIC College of Technology**

**(Autonomous)**

**Kanchikacherla–521180, NTR Dist., Andhra Pradesh**

**July- 2023**

## DECLARATION

I certify that the work contained in this report is original and has been done by me

under the guidance of my supervisor **Mrs. VAMSI.S**. The work has not been submitted to any other institute for any degree or diploma. I have followed the guidelines provided by the institute in preparing the report. I have confirmed to the norms and guidelines given in the Ethical Code of Conduct of the institute. Whenever I have used materials (data, theoretical analysis, figures, and text) from other sources, I have given due credit to the text of the report

and giving their details in their references. Further, I have taken permission from the copyright

owners of the sources, whenever necessary.

**Place: Kanchikacherla**

**Date:**

**ROHIT VARMA.M**

**(21H71F0046)**



**CERTIFICATE**

This is to certify that the project report entitled **“CERTIFICATELESS PUBLIC INTIGRITY CHECKING OF GROUP DATA ON CLOUD STORAGE”** submitted by **Mr. ROHIT VARMA.M. (21H71F0046)** to the DVR & Dr. HS MICCollege of Technology in partial fulfillment of the requirements for the award of the Degree **Master of Computer Application** in Department of DCA is a bonafide record of work carried out by her under my guidance and supervision. The contents of this report, in full or in parts, have not been submitted to any other institute for the award of any Degree.

**Supervisor Head of the Department**

**Principal**

**Internal Examiner** **External Examiner**

**ACKNOWLEDGEMENT**

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people who made it possible and whose constant guidance and engagement crown all the efforts with success. I thank our college management and respected

**Sri D. PANDURANGA RAO,** CEO for providing us the necessary infrastructure to carry out the project work.

I express my sincere thanks to our beloved Principal Dr**.T. Vamsee Kiran** who has been a great source of information for my work.

**Mrs. A. Anuradha Head** of the Department of Computer Applicationsfor extending her support to carry on this project work.

I am thankful to my Project Coordinator **Mr. K. Mahanthi,** Assistant Professor, Department of DCA for their internal support and professionalism who helped me in shaping the project into successful one.

I would like to thank my project supervisor, **Mr. S. Vamsi,** Assistant Professor, Department of Computer Applicationsfor the guidance and support, especially for the valuable ideas and knowledge shared to me throughout the project.

I take this opportunity to express my thanks to one and all who directly or indirectly helped me in bringing this effort to present form.

Finally, my special thanks go to my family for their continuous support and help throughout my academic years and for their continual support and encouragement for the completion of the project.

**ABSTRACT**

Cloud storage service supplies people with an efficient method to share data within a group. The cloud server is not trustworthy, so lots of remote data possession checking (RDPC) protocols are proposed and thought to be an effective way to ensure the data integrity. However, most of RDPC protocols are based on the mechanism of traditional public key infrastructure (PKI), which has obvious security flaw and bears big burden of certificate management. To avoid this shortcoming, identity-based cryptography (IBC) is often chosen to be the basis of RDPC. Unfortunately, IBC has an inherent drawback of key escrow. To solve these problems, we utilize the technique of certificateless signature to present a new RDPC protocol for checking the integrity of data shared among a group. In our scheme, user's private key includes two parts: a partial key generated by the group manager and a secret value chosen by herself/himself. To ensure the right public keys are chosen during the data integrity checking, the public key of each user is associated with her unique identity, for example the name or telephone number. Thus, the certificate is not needed and the problem of key escrow is eliminated too. Meanwhile, the data integrity can still be audited by public verifier without downloading the whole data. In addition, our scheme also supports efficient user revocation from the group. The security of our scheme is reduced to the assumptions of computational Diffie-Hellman (CDH) and discrete logarithm (DL). Experiment results exhibit that the new protocol is very efficient and feasible.

**SYSTEM REQUIREMENTS**

**H/W System Configuration: -**

➢ Processor - Pentium –IV

➢ RAM - 4 GB (min)

➢ Hard Disk - 20 GB

➢ Key Board - Standard Windows Keyboard

➢ Mouse - Two or Three Button Mouse

➢ Monitor - SVGA

**Software Requirements:**

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

**Chapter - 1**

**INTRODUCTION**

**1.1. BACKGROUND WORK**

Cloud Computing refers to each the applications delivered as services over the web and also the hardware and computer program within the datacenters that give those services. The services themselves have long been named as computer code as a Service (SaaS). The datacenter hardware and computer code is named as a Cloud. once a Cloud is formed obtainable during a pay-as-you-go manner to the general public, it's referred to as as a Public Cloud. The service being sold is Utility Computing. Current samples of utility Computing embrace Amazon net Services, Google App Engine, and Microsoft Azure. The term non-public Cloud id accustomed visit internal datacenters of a business or alternative organization that aren't created obtainable to the general public. Thus, Cloud Computing is that the total of SaaS and Utility Computing, however doesn't usually embrace non-public Clouds. Cloud Computing term is employed during a general manner, exchange it with one amongst the opposite terms only if clarity demands it. Fig. 1.1 shows the roles of the folks as users or suppliers of those layers of Cloud Computing.



Fig: 1.1 User and Providers of Cloud Computing

The future Internet covers all research and development activities dedicated to realizing tomorrow’s internet, i.e. enhancing a networking infrastructure which integrates all kind of resources, usage domains etc. As such, research related to cloud technologies form a vital part of the future Internet research agenda. Perplexities in regards to the angles secured by distributed computing as for the future Internet generally emerge from the expansive extent of qualities doled out to "mists", just like the intelligent result of the re-marking blast a few years prior.

In this way, most cloud frameworks have concentrated on facilitating applications and information on remote PCs, utilizing specifically replication techniques to guarantee accessibility and subsequently accomplishing a heap adjusting versatility. Notwithstanding, the reasonable model of mists surpasses such a basic specialized approach and prompts to challenges similar to the ones without bounds Internet, yet with marginally extraordinary concentration because of the mix of ideas and objectives certain to cloud frameworks. As it were, as a mechanical acknowledgment driven by a financial recommendation, cloud frameworks would offer capacities that empower significant parts without bounds Internet.

**Sorts of Cloud Services:**

There are four sorts of cloud administrations, as appeared in Figure 1.2.Some items offer Internet based administrations, for example, stockpiling, middleware, coordinated effort, and database capacities specifically to clients.

**SaaS:** Software-as-a-Service provides software as a shareware. The software can be used for a period of time as per the wish of the user and the user can pay accordingly.

**PaaS:** Platform-as-a-Service products offer usage of a Platform per usage payable. The user can utilize the Platform fully or partially.

**IaaS:** Infrastructure-as-a-Service product offer complete computer infrastructure as a service and the user can pay accordingly.

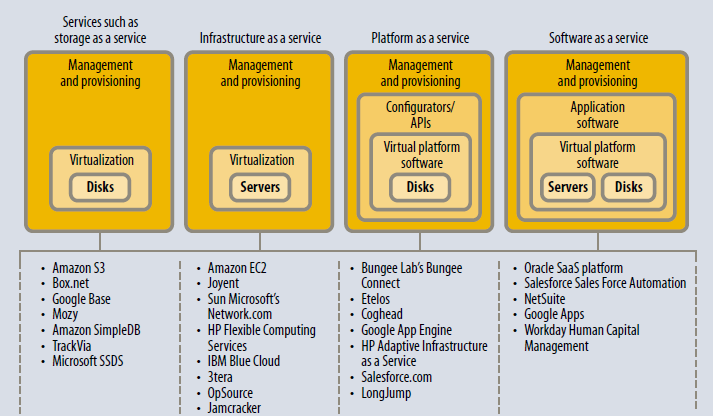


Fig: 1. 2Types of Cloud Service

**Advantages:**

Companies that run their own platforms must buy and maintain their own hardware and software infrastructures and hire staff to take care of the system, all of which can be expensive and time-consuming. Cloud computing eliminates the need to do this. Providing excess computing and storage capacity just to handle maximum workloads can be an inefficient use of resources. Most data centers don’t user more than 50 percent of the total capacity of the resources available.

**Availability:**

A small company generally do not offer much availability. But a large service provider which has several resources and duplicate equipment generally offer more availability.

**Application Integration and Support:**

Well-designed cloud platforms integrate multiple services assets into a powerful compound application which is generally more convenient. The systems generally use Simple Object Access Protocol (SOAP) or Web Services Description Language (WSDL). Other open-source protocols are based on Extensible Markup Language (XML). The resources and other infrastructure services can be supported by these protocols. Using these protocols, the application integration can be done effectively. The support can also be provided.

**Flexibility:**

Most distributed computing sellers don't require contracts and let clients work with their administrations as required. This makes distributed computing a decent approach to get the additional assets required for exercises, for example, testing new administrations or items.

**1.2. OBJECTIVES**

we design a secure, easily integrated, and fine-grained query results verification mechanism, by which, given an encrypted query results set, the query user not only can verify the correctness of each data file in the set but also can further check how many or which qualified data files are not returned if the set is incomplete before decryption. The verification scheme is loose-coupling to concrete secure search techniques and can be very easily integrated into any secure query scheme. We achieve the goal by constructing secure verification object for encrypted cloud data. Furthermore, a short signature technique with extremely small storage cost is proposed to guarantee the authenticity of verification object and a verification object request technique is presented to allow the query user to securely obtain the desired verification object. Performance evaluation shows that the proposed schemes are practical and efficient.

**1.3. PROJECT FEATURES**

1. Cost effectiveness will be provided

2. Application performance will be improved

**Chapter - 2**

**2.1 EXISTING SYSTEM**

Linear index table to support data dynamic. further optimized the implementation of linear index table and provided an efficient RDPC scheme. Presented a public remote integrity checking scheme, which could protect the user identity on file level to reduce the storage and communication cost.

Provided a cooperative PDP scheme for the multi-cloud setting, in which the data blocks were stored on different cloud servers. To improve the security. proposed another identity-based PDP scheme for multi-cloud setting without certificate management. Recently.

Presented an incentive and unconditionally anonymous identity-based public PDP scheme. In order to reduce the computation cost of data owner. presented a proxy-oriented PDP scheme which moved the work of tag generation from data owner to proxy. Two PDP scheme based on certificateless and certificate-based cryptography were proposed respectively.

All the schemes mentioned above focused on the integrity verification for personal data. In 2012, proposed a protocol for checking the integrity of data shared in a group. They utilized the technique of group signature to generate each authentication tag so as to preserve the tag generator's privacy.

Presented a PDP scheme without paring, which also supported dynamic group. Proposed a PDP scheme based on polynomial-based authentication tags, which aimed to solve the problem of multi-user modification for blocks.

**Disadvantages**

1) The system was not implemented Attribute Based Encryption Method which leads less security on outsourced data.

2) The system is less security due to lack of Attribute Based Encryption and there is no block verification.

**2.2 PROPOSED SYSTEM**

In the proposed system, the system mainly focuses on the integrity checking for data shared within a group. Suppose there is a scenario that a software engineer starts an open-source project and calls on volunteers from the world to join the project. They work as a temporary team. All the codes of the project are stored on certain cloud server so that all the team members upload and modify the source code by Internet.

The team may be very big, so it should be set up and managed efficiently. The volunteers may leave the team at any time, so the problem of user revocation from the team should be considered. The most important thing is that there need some way to guarantee the integrity of source codes on cloud sever.

Motivated by such requirement, the system proposes a new RDPC scheme for data shared in a group. Different from previous work, our scheme is based on the certifcateless signature technique to avoid the problems of certificate management and key escrow. In our scheme, the group creator generates the partial key for each group user on behalf of key generation centre. Each user selects a secret value privately. The private key of each group user contains two parts: a partial key and a secret value. All the data blocks are signed by group user to get corresponding authentication tags.

During the data verification, all the tags are aggregated to decrease the computation and communication cost. ased on CDH and DL assumptions, we prove the security of our scheme. Besides, our scheme supports public verification and efficient user revocation. We implement our scheme and perform some experiments. The experiment results indicate that our scheme has good efficiency.

**Advantages**

In the proposed scheme, the shared data is divided into many blocks and each block is attached with an authentication tag. Thus, the CSP stores all the blocks and the corresponding tags for cloud user.The data verifier is a person who checks the integrity of the data on CSP. Due to the feature of public verification, anyone could be the verifier in our scheme.

**2.3. SYSTEM STUDY**

**2.3.1. FEASIBILITY STUDY**

The possibility of the project is analyzed during this part and business proposal is place forth with a awfully general arrange for the project and a few value estimates. throughout system analysis the FEASIBILITY study of the projected system is to be distributed. this can be to make sure that the projected system isn't a burden to the corporate. For risk analysis, some understanding of the key needs for the system is important.

Three key issues concerned within the FEASIBILITY analysis area unit

* ECONOMICAL FEASIBILITY
* TECHNICAL FEASIBILITY
* SOCIAL FEASIBILITY

**ECONOMICAL FEASIBILITY:**  This study is distributed to visualize the economic impact that the system can wear the society. the quantity of fund that the corporate will pour into the analysis and development of the system is restricted. The expenditures should be even. so the developed system additionally inside the budget and this was achieved as a result of most of the technologies used area unit freely out there. solely the bespoken product had to be purchased.

**TECHNICAL FEASIBILITY:** This study is distributed to visualize the technical risk, that is, the technical needs of the system. Any system developed should not have a high demand on the out there technical resources. this can result in high demands on the out there technical resources. this can result in high demands being placed on the shopper. The developed system should have a modest demand, as solely borderline or null changes area unit needed for implementing this technique.

**SOCIAL FEASIBILITY:** The facet of study is to visualize the amount of acceptance of the system by the user. This includes the method of coaching the user to use the system expeditiously. The user should not feel vulnerable by the system, instead should settle for it as a necessity. the amount of acceptance by the users entirely depends on the ways that area unit used to teach the user regarding the system and to form him .

**2.4. LITERATURE SURVEY**

Literature [survey](http://www.blurtit.com/q876299.html) is the most important step in software development process. Before developing the tool it is necessary to determine the time factor, economy and company Traffic Redundancy Elimination, once these things are satisfied, then next steps are to determine which operating system and language can be used for developing the tool. Once the [programmers](http://www.blurtit.com/q876299.html) start building the tool the programmers need lot of external support.

This support can be obtained from senior programmers, from [book](http://www.blurtit.com/q876299.html) or from websites. Before building the system we have to knownthe below concepts for developing the proposed system.

**1) DAC-MACS: Effective data access control for multi-authority cloud storage systems**

**AUTHORS:**  K. Yang, X. Jia, and K. Ren

Data access control is an effective way to ensure the data security in the cloud. However, due to data outsourcing and untrusted cloud servers, the data access control becomes a challenging issue in cloud storage systems. Existing access control schemes are no longer applicable to cloud storage systems, because they either produce multiple encrypted copies of the same data or require a fully trusted cloud server. Ciphertext-Policy Attribute-based Encryption (CP-ABE) is a promising technique for access control of encrypted data. It requires a trusted authority manages all the attributes and distributes keys in the system. In cloud storage systems, there are multiple authorities co-exist and each authority is able to issue attributes independently. However, existing CP-ABE schemes cannot be directly applied to the access control for multi-authority cloud storage systems, due to the inefficiency of decryption and revocation. In this paper, we propose DAC-MACS (Data Access Control for Multi-Authority Cloud Storage), an effective and secure data access control scheme with efficient decryption and revocation. Specifically, we construct a new multi-authority CP-ABE scheme with efficient decryption and also design an efficient attribute revocation method.

**2) DACC: Distributed access control in clouds**

**AUTHORS:** S. Raju, A. Nayak, and I. Tomjanovich

We propose a new model for data storage and access in clouds. Our scheme avoids storing multiple encrypted copies of same data. In our framework for secure data storage, cloud stores encrypted data (without being able to decrypt them). The main novelty of our model is addition of key distribution centers (KDCs). We propose DACC (Distributed Access Control in Clouds) algorithm, where one or more KDCs distribute keys to data owners and users. KDC may provide access to particular fields in all records. Thus, a single key replaces separate keys from owners. Owners and users are assigned certain set of attributes. Owner encrypts the data with the attributes it has and stores them in the cloud. The users with matching set of attributes can retrieve the data from the cloud. We apply attribute-based encryption based on bilinear pairings on elliptic curves. The scheme is collusion secure; two users cannot together decode any data that none of them has individual right to access. DACC also supports revocation of users, without redistributing keys to all the users of cloud services. We show that our approach results in lower communication, computation and storage overheads, compared to existing models and schemes.

3) **Expressive, efficient and revocable data access control for multi-authority cloud storage**

**AUTHORS:** K. Yang and X. Jia

Data access control is an effective way to ensure the data security in the cloud. Due to data outsourcing and untrusted cloud servers, the data access control becomes a challenging issue in cloud storage systems. Ciphertext-Policy Attribute-based Encryption (CP-ABE) is regarded as one of the most suitable technologies for data access control in cloud storage, because it gives data owners more direct control on access policies. However, it is difficult to directly apply existing CP-ABE schemes to data access control for cloud storage systems because of the attribute revocation problem. In this paper, we design an expressive, efficient and revocable data access control scheme for multi-authority cloud storage systems, where there are multiple authorities co-exist and each authority is able to issue attributes independently. Specifically, we propose a revocable multi-authority CP-ABE scheme, and apply it as the underlying techniques to design the data access control scheme. Our attribute revocation method can efficiently achieve both forward security and backward security. The analysis and simulation results show that our proposed data access control scheme is secure in the random oracle model and is more efficient than previous works.

**4) Privacy preserving cloud data access with multi-authorities**

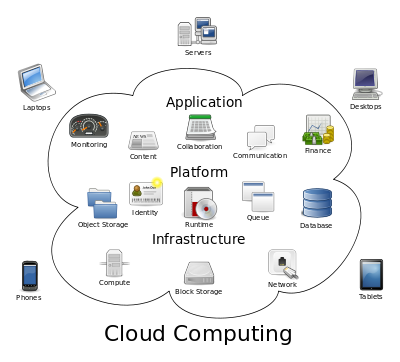
**AUTHORS:** T. Jung, X. Li, Z. Wan, and M. Wan

Cloud computing is a revolutionary computing paradigm which enables flexible, on-demand and low-cost usage of computing resources. Those advantages, ironically, are the causes of security and privacy problems, which emerge because the data owned by different users are stored in some cloud servers instead of under their own control. To deal with security problems, various schemes based on the Attribute-Based Encryption have been proposed recently. However, the privacy problem of cloud computing is yet to be solved. This paper presents an anonymous privilege control scheme Any Control to address not only the data privacy problem in a cloud storage, but also the user identity privacy issues in existing access control schemes. By using multiple authorities in cloud computing system, our proposed scheme achieves anonymous cloud data access and fine-grained privilege control. Our security proof and performance analysis shows that Any Control is both secure and efficient for cloud computing environment.

* 1. **CLOUD COMPUTING OVERVIEW**

**2.5.1 CLOUD OVERVIEW**

Distributed computing is a casual expression used to depict an assortment of various figuring ideas that include countless that are associated through a constant correspondence network(typically the Internet). Distributed computing is a language term without a generally acknowledged non-equivocal logical or specialized definition. In science, distributed computing is an equivalent word for circulated processing over a system and means the capacity to run a program on numerous associated PCs in the meantime. The notoriety of the term can be ascribed to its utilization in advertising to offer facilitated benefits in the feeling of use administration provisioning that run customer server programming on a remote area.



**Fig 2.2: Cloud Computing**

A few Traffic Redundancy Elimination are opening up the period of distributed computing, which is an Internet-based improvement and utilization of PC innovation. The ever less expensive and all the more capable processors, together with the Software as a Service (SaaS) registering engineering, are changing server farms into pools of figuring administration on an enormous scale. The expanding system transfer speed and dependable yet adaptable system associations make it even conceivable that clients can now subscribe top notch administrations from information and programming that live exclusively on remote server farms. Moving information into the cloud offers awesome accommodation to clients since they don't need to think about the complexities of direct equipment administration. The pioneer of distributed computing merchants, Amazon Simple Storage Service (S3), and Amazon Elastic Compute Cloud (EC2) are both surely understood illustrations. While these web based online administrations do give enormous measures of storage room and adaptable processing assets, this figuring stage move, be that as it may, is wiping out the duty of neighborhood machines for information upkeep in the meantime. Thus, clients are helpless before their cloud benefit suppliers (CSP) for the accessibility and honesty of their information from one viewpoint, in spite of the fact that the cloud frameworks are a great deal more capable and solid than individualized computing gadgets, expansive scope of both interior and outside dangers for information trustworthiness still exist. Cases of blackouts and information misfortune episodes of significant distributed storage administrations show up now and again.

Then again, since clients may not hold a neighborhood duplicate of outsourced information, there exist different motivators for CSP to carry on unfaithfully toward the cloud clients with respect to the status of their outsourced information. For instance, to build the net revenue by diminishing cost, it is workable for CSP to dispose of infrequently got to information without being recognized in an auspicious manner .Similarly, CSP may even endeavor to conceal information misfortune occurrences in order to keep up a notoriety Therefore, in spite of the fact that outsourcing information into the cloud is financially alluring for the cost and many-sided quality of long haul vast scale information stockpiling, its lacking of offering solid affirmation of information trustworthiness and accessibility may obstruct its wide selection by both venture and individual cloud clients. Keeping in mind the end goal to accomplish the affirmations of cloud information uprightness and accessibility and implement the nature of distributed storage benefit, proficient techniques that empower on-request information accuracy check for cloud clients must be planned.

Nonetheless, the way that clients no longer have physical ownership of information in the cloud precludes the immediate selection of conventional cryptographic primitives with the end goal of information trustworthiness insurance. Thus, the check of distributed storage accuracy must be directed without express warning of the entire information documents. In the mean time, distributed storage is not only an outsider information stockroom.

The information put away in the cloud may be gotten to as well as be much of the time overhauled by the clients, including addition, cancellation, adjustment, adding, and so forth. Along these lines, it is additionally basic to bolster the combination of this dynamic element into the distributed storage rightness certification, which makes the framework outline significantly all the more difficult. in order to lessen the information honesty and accessibility dangers. Along these lines, circulated conventions for capacity rightness confirmation will be of most significance in accomplishing hearty and secure distributed storage frameworks.

In any case, such imperative region stays to be completely investigated in the writing. As of late, the significance of guaranteeing the remote information respectability has been highlighted by the accompanying examination works under various framework and security models. These procedures, while can be valuable to guarantee the capacity rightness without having clients having nearby information, are all concentrating on single server situation. They might be valuable for nature of-administration testing, yet does not ensure the information accessibility in the event of server disappointments. Albeit guide applying these strategies to dispersed stockpiling (numerous servers) could be direct, the came about capacity check overhead would be straight to the quantity of servers. As a corresponding methodology, specialists have likewise proposed disseminated conventions for guaranteeing stockpiling rightness over numerous servers or companions. In any case, while giving proficient cross server stockpiling check and information accessibility protection, these plans are all concentrating on static or recorded information. Subsequently, their capacity of taking care of element information stays hazy, which definitely confines their full appropriateness in distributed storage situations.

|  |  |
| --- | --- |
|  | **Cloud Clients**  Web browser, Mobile, Thin client, terminal emulator. |
|  |  |
| **Application** | **SaaS**  CRM, Email, Virtual desktop, communication, Games. |
| **Platform** | **PaaS**  Execution runtime, database, Webserver, Development tools. |
| **Infrastructure** | **IaaS**  Virtual machines, Servers, Storage, Load balancers, Network. |

**Fig 2.3: Cloud Computing Layers**

n apparatus the developers need parcel of outer backing. This backing can be gotten from senior software engineers, from book or from sites. Before building the framework the above thought r considered for building up the proposed framework.

A viable and adaptable disseminated stockpiling check plan with unequivocal element information backing to guarantee the accuracy and accessibility of client information in the cloud, this depend on eradication amending code in the record circulation arrangement to give redundancies and assurance the information constancy against Byzantine servers, where a capacity server may come up short in self-assertive ways. This development radically decreases the correspondence and capacity overhead when contrasted with the conventional replication-based document appropriation methods. By using the homomorphism token with appropriated check of eradication coded data,this plan accomplishes the capacity accuracy protection and in addition information blunder restriction.

At whatever point information debasement has been distinguished amid the capacity rightness check, ouratur plan can just about surety the concurrent restriction of information mistakes, i.e., the recognizable proof of the getting out of hand server(s). With a specific end goal to strike a decent harmony between blunder versatility and information progression, this further investigate the arithmetical property of the token calculation and deletion coded information, and exhibit how to proficiently bolster dynamic operation on information pieces, while keeping up the same level of capacity rightness affirmation. Keeping in mind the end goal to spare the time, calculation assets, and even the related online weight of clients, this likewise give the augmentation of the proposed principle plan to bolster outsider inspecting, where clients can securely designate the honesty checking errands to outsider examiners and be effortless to utilize the distributed storage administrations. My work is among the initial couple of ones in this field to consider dispersed information stockpiling security in Cloud Computing.

At the establishment of distributed computing is the more extensive idea of base joining (or Converged Infrastructure) and shared administrations. This kind of information Traffic Redundancy Elimination environment permits undertakings to get their applications up and running quicker, with simpler sensibility and less upkeep, and empowers IT to all the more quickly alter IT assets, (for example, servers, stockpiling, and systems administration) to meet fluctuating and erratic business request.

The two words in the expression distributed computing have the accompanying understandings:

•Cloud: As a thing, this is an analogy for the Internetand as a descriptive word it implies relating to the Internet this use gets from the cloud images that speak to the Internet on charts.

•Computing: Any IT movement did by an element: At the point when utilizing "a neighborhood server or a PC", which infers that the IT assets are under the restrictive control of the element. To "store, oversee, and handle information", which infers that the information is private to the element, as in it is controlled by them, regardless of the possibility that it is open by others.

•Access to the assets is

1.Controlled by the element, and confined by them to their approved clients.

2.Delivered by means of the Internet to these clients.

•The assets are

1.Hosted by an administration supplier for the benefit of the substance.

2.Dedicated to the selective utilization of the substance.

•Data handled by the assets is

1.Private to the element and its partners.

2.Entered or gathered by them, or consequently delivered for them.

**2.5.2 SECURITY CONCERNS FOR CLOUD COMPUTING**

Distributed computing gets its name from the drawings ordinarily used to portray the Internet. Distributed computing is another utilization and conveyance model for IT administrations. The idea of distributed computing speaks to a movement in thought, in those end clients need not know the points of interest of a particular innovation. The administration is completely overseen by the supplier. Clients can devour administrations at a rate that is set by their specific needs. This on interest administration can be given whenever.

**Fig 2.4: Cloud Computing Security**

While cost and ease of use are two great benefits of cloud computing, there are significant security concerns that need to be addressed when considering moving critical applications and sensitive data to public and shared cloud environments. To address these concerns, the cloud provider must develop sufficient controls to provide the same or a greater level of security than the organization would have if the cloud were not used. Listed here are ten items to review when considering cloud computing.

**1.Where’s the data?** Different countries have different requirements and controls placed on access. Because your data is in the cloud, you may not realize that the data must reside in a physical location. Your cloud provider should agree in writing to provide the level of security required for your customers.

**2.Who has access?** Access control is a key concern, because insider attacks are a huge risk. A potential hacker is someone who has been entrusted with approved access to the cloud. If anyone doubts this, consider that in early 2009 an insider was accused of planting a logic bomb on Fanny Mae servers that, if launched, would have caused massive damage. Anyone considering using the cloud needs to look at who is managing their data and what types of controls are applied to these individuals.

**3**. **What happens if there is a security breach?** If a security incident occurs, what support will you receive from the cloud provider? While many providers promote their services as being unpacks able, cloud-based services are an attractive target to hackers.

**2.5.3 CLOUD COMPUTING MODELS**

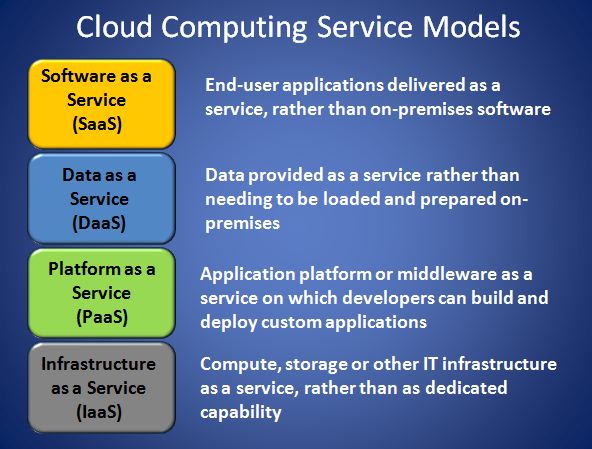
Distributed computing suppliers offer their administrations as indicated by a few key models: foundation as an administration (IaaS), stage as an administration (PaaS), and programming as an administration (SaaS) where IaaS is the most essential and each higher model modified works from the subtle elements of the lower models. Other key segments in anything as an administration (XaaS) are portrayed in a complete scientific classification demonstrate distributed in 2009, for example, Strategy-as-a-Service, Collaboration-as-a-Service, Business Process-as-a-Service, Database-as-a-Service, and so on. In 2012, arrange as an administration (NaaS) and correspondence as an administration (CaaS) were authoritatively included by ITU (International Telecommunication Union) as a component of the fundamental distributed computing models, perceived administration classes of a media transmission driven cloud biological system

**Fig 2.5: Cloud Services**

1.Infrastructure-as-a-Service (IaaS) – As the name infers, you are purchasing framework. You claim the product and are buying virtual energy to execute as required. This is much similar to running a virtual server all alone hardware, aside from you are currently running a virtual server on a virtual circle. This model is like a service organization display, as you pay for what you utilize. A case is Amazon Web Services at http://aws.amazon.com/.

2.Platform-as-a-Service (PaaS) – In this model of distributed computing, the supplier gives a stage to your utilization. Administrations gave by this model incorporate all periods of the framework advancement life cycle (SDLC) and can utilize application program interfaces(APIs), site entrances, or passage programming. Purchasers do need to take a gander at particular arrangements, since a few suppliers don't permit programming made by their clients to be gotten off the supplier's stage. A case of PaaS is Google Apps.

3. Software-as-a-Service (SaaS) – This model is intended to give everything and essentially lease the product to the client. The administration is normally given through some kind of front end or online interface. While the end client is allowed to utilize the administration from anyplace, the organization pays a for every utilization charge. Salesforce.com offers this sort of administration.



**Fig 2.6: Cloud Computing Service Models**

**2.5.4 CLOUD PROVIDERS**

Gartner predicts that cloud computing will surge to 150 billion dollars by 2013. Below is a partial list of companies that provide cloud computing services:

1. Amazon

2. Google

3. Microsoft

4. Salesforce.com

5. Citrix

6. IBM

7. Mozyhome

8. Sun

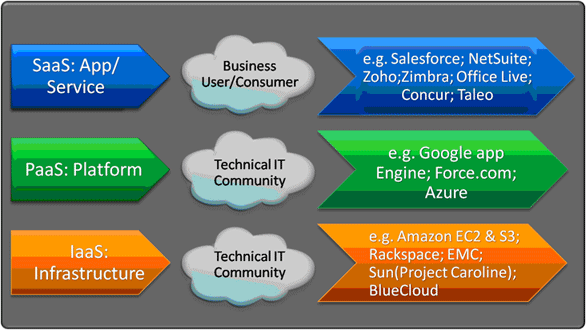
9. CohensiveFT

10. Icloud

11. Nivanix

12. VMware

13. Flexscale



**Fig 2.7: Cloud Computing Provider**

**2.5.5 CLOUD COMPUTING ATTACKS**

As more companies move to cloud computing, look for hackers to follow. Some of the potential attack vectors criminals may attempt include:

* Denial of Service (DoS) attacks - Some security professionals have argued that the cloud is more vulnerable to DoS attacks, because it is shared by many users, which makes DoS attacks much more damaging. Twitter suffered a devastating DoS attack during 2009.
* Side Channel attacks – An attacker could attempt to compromise the cloud by placing a malicious virtual machine in close proximity to a target cloud server and then launching a side channel attack.
* Authentication attacks – Authentication is a weak point in hosted and virtual services and is frequently targeted. There are many different ways to authenticate users; for example, based on what a person knows, has, or is.
* The mechanisms used to secure the authentication process and the methods used are a frequent target of attackers.
* Man-in-the-middle cryptographic attacks – This attack is carried out when an attacker places himself between two users.
  + 1. **SECURITY IN CLOUD COMPUTING**

**1. FEAR OF THE CLOUD**

The "security" worries that are keeping organizations from exploiting the cloud are Numerous studies, for instance IDC's 2008 Cloud Services User Survey of IT officials, refer to security as the main test for cloud clients.

The Cloud Security Alliance's underlying report contains an alternate kind of scientific classification in light of 15 distinctive security areas and the procedures that should be followed in a general cloud organization. I arrange the security worries as:

• Traditional security

• Availability

• Third-party information control

**2. Conventional SECURITY**

These worries include PC and system interruptions or assaults that will be made conceivable or possibly less demanding by moving to the cloud. Cloud suppliers react to these worries by belligerence that their efforts to establish safety and procedures are more develop and tried than those of the normal organization. Another contention, made by the Jericho Forum, is: "It could be simpler to secure data if it's controlled by an outsider as opposed to in-house, if organizations are stressed over insider dangers. What's more, it might be simpler to uphold security through contracts with online administrations suppliers than by means of inner controls." Concerns in this classification include:

TS1. VM-level assaults: Potential vulnerabilities in the hypervisor or VM innovation utilized by cloud merchants are a potential issue in multi-inhabitant structures. Vulnerabilities have showed up in VMWare, Xen, and Microsoft's Virtual PC and Virtual Server. Sellers, for example, Third Brigade relieve potential VM-level vulnerabilities through observing and firewalls.

TS2. Cloud benefit supplier vulnerabilities: These could be stage level, for example, a SQL-infusion or cross-site scripting helplessness in salesforce.com. For example, there have been a few late Google Docs vulnerabilities and The Google reaction to one of them is here: . There is just the same old thing new in the way of these vulnerabilities; just their setting is novel. Indeed, IBM has repositioned its Rational App Scan instrument, which checks for vulnerabilities in web benefits as a cloud security benefit.

TS3. Phishing cloud supplier: Phishes and other social architects have another assault vector, as the Sales drive phishing episode appears.

TS4. Extended system assault surface: The cloud client must ensure the foundation used to interface and communicate with the cloud, an assignment confused by the cloud being outside the firewall by and large. For example, demonstrates a case of how the cloud may assault the machine associating with it.

TS5. Confirmation and Authorization: The undertaking validation and approval system does not normally stretch out into the cloud. How does an organization merge its current system to incorporate cloud assets? Besides, how does an endeavor combine cloud security information (if even accessible) with its own security measurements and strategies?

TS6. Criminology in the cloud: This blog posting on the CLOIDIFIN extend outlines the trouble of cloud measurable examinations: "Conventional advanced scientific approachs allow specialists to seize gear and perform nitty gritty investigation on the media and information recouped. The probability along these lines, of the information being evacuated, overwritten, erased or devastated by the culprit for this situation is low. All the more firmly connected to a CC domain would be organizations that possess and keep up their own particular multi-server sort foundation, however this would be on a far littler scale in examination. Be that as it may, the size of the cloud and the rate at which information is overwritten is of concern."

**3. Accessibility**

These worries focus on basic applications and information being accessible. All around pitched occurrences of cloud blackouts incorporate Gmail (one-day blackout in mid-October 2008), Amazon S3 (more than seven-hour downtime on July 20, 2008), and Flexi Scale (18-hour blackout on October 31, 2008).

A1. Up time. Similarly as with the Traditional Security concerns, cloud suppliers contend that their server uptime contrasts well and the accessibility of the cloud client's own particular server farms.

Other than just administrations and applications being down, this incorporates the worry that an outsider cloud would not scale all around ok to handle certain applications. SAP's CEO, Leo Apotheker said: "There are sure things that you can't keep running in the cloud in light of the fact that the cloud would fall… Don't trust that any service organization will run its charging for 50 million shoppers in the cloud."

A2. Single purpose of disappointment: Cloud administrations are considered as giving more accessibility, yet maybe not – there are more single purposes of disappointment and assault.

A3. Affirmation of computational trustworthiness: Can an endeavor be guaranteed that a cloud supplier is reliably running a facilitated application and giving substantial results? For instance, Stanford's Folding@Home extend gives a similar undertaking to numerous customers to achieve an agreement on the right result.

**4. Outsider DATA CONTROL**

The legitimate ramifications of information and applications being held by an outsider are unpredictable and not surely knew. There is additionally a potential absence of control and straightforwardness when an outsider holds the information. Part of the buildup of distributed computing is that the cloud can be execution free, however as a general rule administrative consistence requires straightforwardness into the cloud. This is provoking a few organizations to manufacture private mists to maintain a strategic distance from these issues but then hold a portion of the benefits of distributed computing.

BL1. Due constancy: If served a subpoena or other lawful activity, can a cloud client force the cloud supplier to react in the required time period.

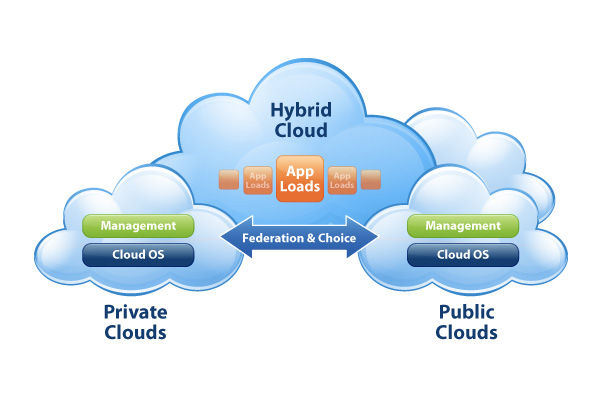
BL2. Review capacity: Audit trouble is another symptom of the absence of control in the cloud. Is there adequate straightforwardness in the operations of the cloud supplier for inspecting purposes? As of now, this straightforwardness is given by documentation and manual reviews. A related concern is appropriate administration of cloud-related movement. It's simple, maybe too simple, to begin utilizing a cloud benefit. One prevalent evaluating rule is the SAS 70, which characterizes rules for inspectors to survey inner controls, for example controls over the handling of touchy data. SOX and HIPAA are other surely understood controls. US government offices for the most part need to take after rules from FISMA, NIST, and FIPS. Certain controls require information and operations to stay in certain geographic areas. Cloud suppliers are starting to react with geo-focused on offerings.

BL3. Authoritative commitments: One issue with utilizing another organization's foundation other than the questionable arrangement of interests is that there may astound lawful ramifications. For example, here is a section from Amazon's terms of utilization:

BL4. Cloud Provider Espionage: This is the stress of robbery of organization exclusive data by the cloud supplier. For instance, Google Gmail and Google Apps are cases of administrations upheld by a private cloud foundation. Corporate clients of these administrations are worried about privacy and accessibility of their information. As indicated by a CNN article:

BL5. Information Lock-in: How does a cloud client maintain a strategic distance from secure to a specific distributed computing seller? The information may itself be secured an exclusive configuration, and there are additionally issues with preparing and procedures. There is additionally the issue of the cloud client having no power over incessant changes in cloud-based administrations. Coghead is one case of a cloud stage whose shutdown left clients scrambling to re-compose their applications to keep running on an alternate stage. Obviously, one response to secure is institutionalization, for example GoGrid API.

**2.5.7 TYPES OF CLOUD COMPUTING**

There are numerous contemplations for distributed computing draftsmen to make when moving from a standard undertaking application arrangement model to one in light of distributed computing. There are open and private mists that offer reciprocal advantages, there are three fundamental administration models to consider, and there the estimation of open APIs versus exclusiveones.****

**Fig 2.8: Cloud Computing Types**

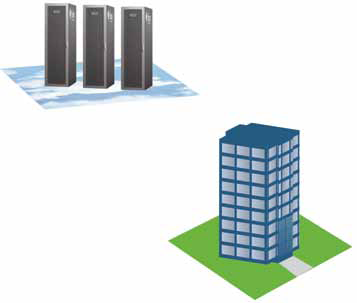
**Public, Private and Hybrid Clouds**

IT associations can convey applications on open, private, or cross breed mists, each of which has its exchange offs. The terms open, private, and mixture don't manage area. While open mists are regularly "out there" on the Internet and private mists are commonly situated on premises, a private cloud may be facilitated at a collocation office also. Organizations may make various contemplations as to which distributed computing model they utilize, and they may utilize more than one model to tackle diverse issues. An application required on a brief premise may be most appropriate for arrangement in an open cloud since it stays away from the need to buy extra hardware to settle an impermanent need. Similarly, a lasting application, or one that has particular prerequisites on nature of administration or area of information, may best be sent in a private or half and half cloud.

**1. Open Cloud**

Open cloud (additionally alluded to as "outer" cloud) depicts the traditional significance of distributed computing: adaptable, progressively provisioned, frequently virtualized assets accessible over the Internet from an off-website outsider supplier, which isolates up assets and bills its clients on an "utility" premise. A case is Think Grid, an organization that gives a multi-inhabitant engineering to providing administrations, for example, Hosted Desktops, Software as a Service and Platform as a Service. Other prevalent cloud merchants incorporate Salesforce.com, Amazon EC2 and Flexi scale. Open mists are controlled by outsiders, and applications from various clients are probably going to be combined on the cloud's servers, stockpiling frameworks, and systems. Open mists are regularly facilitated far from client premises, and they give an approach to diminish client hazard and cost by giving an adaptable, even transitory expansion to big business foundation. On the off chance that an open cloud is actualized in light of execution, security, and information region, the presence of different applications running in the cloud ought to be straightforward to both cloud engineers and end clients.

Undoubtedly, one of the advantages of open mists is that they can be much bigger than an organization's private cloud may offer, the capacity to scale here and there on request, and moving framework dangers from the undertaking to the cloud supplier, if even just briefly. Segments of an open cloud can be cut out for the select utilization of a solitary customer, making a virtual private datacenter. Instead of being restricted to sending virtual machine pictures in an open cloud, a virtual private datacenter gives clients more noteworthy perceivability into its foundation. Presently clients can control virtual machine pictures, as well as servers, stockpiling frameworks, arrange gadgets, and system topology..

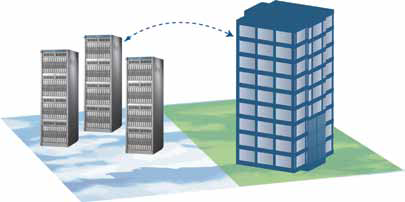


**Fig 2.9: Public Enterprise**

Surely, one of the advantages of open mists is that they can be much bigger than an organization's private cloud may offer, the capacity to scale all over on request, and moving foundation dangers from the undertaking to the cloud supplier, if even just briefly. Parts of an open cloud can be cut out for the restrictive utilization of a solitary customer, making a virtual private datacenter. Instead of being restricted to sending virtual machine pictures in an open cloud, a virtual private datacenter gives clients more noteworthy perceivability into its framework. Presently clients can control virtual machine pictures, as well as servers, stockpiling frameworks, organize gadgets, and system topology. An open cloud gives administrations to numerous clients, and is normally conveyed at a collocation office.

**2. Private Cloud**

To buy, set up and deal with their own particular mists. Private mists are worked for the select utilization of one customer, giving the nearly control over information, security, and nature of administration. The organization claims the foundation and has control over how applications are sent on it. Private mists might be sent in an undertaking datacenter, and they additionally might be conveyed at a collocation office. Private mists can be constructed and oversaw by an organization's own particular IT association or by a cloud supplier. In this "facilitated private" model, an organization, for example, Sun can introduce, design, and work the foundation to bolster a private cloud inside an organization's endeavor datacenter. This model gives organizations an abnormal state of control over the utilization of cloud assets while getting the mastery expected to build up and work nature.



**Fig 2.10: Private enterprise**

Private mists might be facilitated at a collocation office or in an endeavor datacenter. They might be bolstered by the organization, by a cloud supplier, or by an outsider, for example, an outsourcing firm.

**3. Half breed Cloud**

It has been proposed that a half breed cloud environment joining assets from both inward and outer suppliers will turn into the most prevalent decision for ventures. For instance, an organization could utilize an open cloud benefit for general processing, yet store its business-basic information inside its own server farm. This might be on the grounds that bigger associations are probably going to have as of now put vigorously in the foundation required to give assets in-house – or they might be worried about the security of open mists. Mixture mists consolidate both open and private cloud models.

They can give on-request, remotely provisioned scale. The capacity to enlarge a private cloud with the assets of an open cloud can be utilized to keep up administration levels despite fast workload changes. This is regularly observed with the utilization of capacity mists to bolster Web 2.0 applications. A half and half cloud additionally can be utilized to handle arranged workload spikes. Once in a while called "surge registering," an open cloud can be utilized to perform occasional errands that can be conveyed effectively on an open cloud. Cross breed mists present the many-sided quality of deciding how to disperse applications crosswise over both an open and private cloud. Among the issues that should be considered is the relationship amongst information and preparing assets. In the event that the information is little, or the application is stateless, a crossover cloud can be considerably more fruitful than if a lot of information must be moved into an open cloud for a little measure of preparing.

****

**Fig 2.11: Hybrid cloud**

.

**What services can be used in the cloud?**

There are various administrations that can be conveyed through distributed computing, exploiting the dispersed cloud show. Here are some concise portrayals of a couple of the most famous cloud-based IT arrangements:

Facilitated desktops:

Facilitated desktops expel the requirement for conventional desktop PCs in the workplace environment, and lessen the cost of giving the administrations that you require. A facilitated desktop looks and carries on like a customary desktop PC, yet the product and information clients utilize are housed in remote, very secure server farms, as opposed to all alone machines. Clients can just get to their facilitated desktops by means of a web association from anyplace on the planet, utilizing either a current PC or portable workstation or, for greatest cost proficiency, a particular gadget called a thin customer.

Facilitated email:

As more associations search for a safe, dependable email arrangement that won't cost the earth, they are progressively swinging to facilitated Microsoft Exchange® email arranges. Utilizing the world's chief email stage, this administration lets associations both vast and little receive the rewards of utilizing MS Exchange® accounts without investing in the expensive foundation themselves. Email is put away midway on oversaw servers, giving excess and quick availability from any area. This permits clients to get to their email, schedule, contacts and shared documents by an assortment of means, including Outlook®, Outlook Mobile Access (OMA) and Outlook Web Access (OWA).

Facilitated communication (VoIP) :

VOIP (Voice Over IP) is a method for conveying telephone calls and administrations crosswise over advanced web systems. Regarding essential use and usefulness, VOIP is the same to conventional communication, and a VOIP-empowered phone works precisely like an "ordinary" one, yet it has particular cost points of interest

Distributed storage:

Distributed storage is developing in prevalence because of the advantages it gives, for example, straightforward, Cap sans ex costs, anyplace get to and the evacuation of the weight of in-house support and administration. It is fundamentally the conveyance of information stockpiling as an administration, from an outsider supplier, with get to by means of the web and charging computed on limit utilized as a part of a specific period (e.g. every month).

Dynamic Servers:

Dynamic servers are the up and coming era of server environment, supplanting the traditional idea of the committed server. A supplier like Think Grid gives its clients access to assets that look and feel precisely like a devoted server, however that are completely versatile. You can straightforwardly control the measure of handling force and space you utilize, which means you don't need to pay for equipment you needn't bother with. Regularly, you can roll out improvements to your dynamic server whenever, on the fly, without the expenses connected with moving starting with one server then onto the next.

**2.5.8 CHARACTERISTICS OF CLOUD COMPUTING**

• High adaptability: Cloud situations empower overhauling of business prerequisites for bigger gatherings of people, through high versatility

• Agility: The cloud works in the 'conveyed mode' environment. It offers assets among clients and assignments, while enhancing proficiency and deftness (responsiveness) .

• High accessibility and unwavering quality: Availability of servers is high and more dependable as the odds of framework disappointment are negligible.

• Multi-sharing: With the cloud working in a dispersed and shared mode, different clients and applications can work all the more effectively with cost diminishments by sharing normal framework

• Services in pay-per-utilize mode: SLAs between the supplier and the client must be characterized when offering administrations in pay per utilize mode. This might be founded on the unpredictability of administrations advertised.

• Application Programming Interfaces: (APIs) might be offered to the clients so they can get to administrations on the cloud by utilizing these APIs

The on-request, self-benefit, pay-by-utilize nature of distributed computing is additionally an augmentation of built up Traffic Redundancy Elimination nds. An endeavor point of view, the on-request nature of distributed computing underpins the execution and limit parts of administration level targets. The self-benefit nature of distributed computing permits associations to make versatile situations that extend and contract in light of the workload and target execution parameters.

The compensation by-utilize nature of distributed computing may appear as gear rent that assurance a base level of administration from a cloud supplier. Virtualization is a key component of this model. IT associations have comprehended for quite a long time that virtualization permits them to rapidly and effectively make duplicates of existing situations — in some cases including various virtual machines — to bolster test, improvement, and arranging exercises. The cost of these situations is negligible in light of the fact that they can exist together on an indistinguishable servers from generation situations since they utilize couple of assets. In like manner, new applications can be created and conveyed.

Administrations are conveyed over the system are

It nearly abandons saying that distributed computing amplifies the current Traffic Redundancy Elimination and of making administrations accessible over the system. For all intents and purposes each business association has perceived the estimation of Web-based interfaces to their applications, whether they are made accessible to clients over the Internet, or whether they are inside applications that are made accessible to approved workers, accomplices, providers.

Advantages of Cloud Computing

So as to profit the most from distributed computing, designers must have the capacity to refractor their applications so they can best utilize the compositional and sending ideal models that distributed computing underpins. The advantages of conveying applications utilizing distributed computing incorporate decreasing run time and reaction time, minimizing the danger of sending physical foundation, bringing down the cost of passage, and expanding the pace of development.

As indicated by International Data Corporation (IDC), "The expansion of gadgets, consistence, enhanced frameworks execution, online business and expanded replication to auxiliary or reinforcement locales is adding to a yearly multiplying of the measure of data transmitted over the Internet." The cost of managing this measure of information is something that organizations must address. In today's economy, organizations are taking a gander at any cost sparing measures, and the main issue is that distributed computing gives much more prominent adaptability than past figuring models. The advantages of distributed computing are numerous. One is diminished cost, since you pay as you go. Different advantages are the transportability of the application is that clients can telecommute, work, or at customer areas. This expanded versatility implies workers can get to data anyplace they are. There is likewise the capacity of distributed computing to free-up IT laborers who may have been involved performing upgrades, introducing patches, or giving application bolster.

Security and information material science

Information is frequently the most profitable of an organization's advantages, and it must be secured with as much watchfulness as whatever other resource. It is anything but difficult to contend that more watchfulness is expected to ensure information in light of how a gatecrasher can possibly achieve an organization's information from anyplace on the Internet. A few stages to take include:

• Encrypt information very still so that if any interloper can infiltrate a cloud supplier's security or if a setup blunder makes that information open to unapproved parties, that the information can't be deciphered.

• Encrypt information in travel. Expect that the information will ignore open framework and could be seen by any gathering in the middle.

• Require solid confirmation between application segments with the goal that information is transmitted just to known gatherings.

• Pay consideration regarding cryptography and how calculations are split and are supplanted by new ones after some time. For instance, now that MD5 has been demonstrated helpless against assault, utilize a more grounded method, for example, SHA-256. • Consider utilizing solid, token-based confirmation for overseer parts.For client login/watchword get to, consider who deals with the validation server and whether it is under the organization or the cloud supplier's control.

**Chapter - 3**

**SYSTEM REQUIREMENTS**

# 3.1. HARDWARE REQUIREMENTS

# Processor - Intel Core2 Duo

* Speed - 2.4 GHz
* RAM - 2 GB(minimum)
* Hard Disk - 180 GB

# 3.2. SOFTWARE REQUIREMENTS

* Operating System : Windows XP
* Front End : JSP,Servelets

Database : My Sq

Database Connectivity : JDBC

**Chapter - 4**

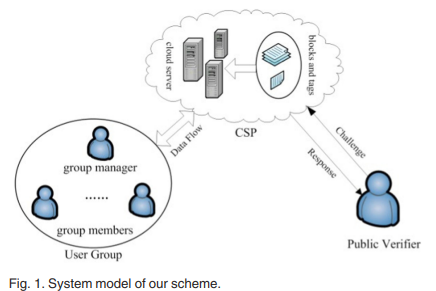
**SYSTEM DESIGN**

**4.1. SYSTEM ARCHITECTURE**

Robustness: We analyse the robustness of the RMM. We use the notion of robustness in a normative system, as developed in [9]. In this notion of robustness, it is assumed that a subset of agents in a normative multi-agent system always violate the norms. Given the fraction of such non-compliant agents, the multi-agent system is robust if it works properly as other agents remain compliant. In this paper, we use a similar notion of robustness. We show the demography of rational and irrational agents (with a majority of irrational agents) for which the proposed RMM remains functional.

2) Reputation of the CPs: We show that the reputations of the CPs who differentiate between good and malicious users, and do not allow them to share resources, increase compared with the CPs who do not make such differentiation.

3) Reputation of the users: We show that, a good user gets better reputation than a malicious user.



**4.2. UML DIAGRAMS**

• UML remains for Unified Modeling Language. UML is an institutionalized broadly useful displaying dialect in the field of protest situated programming designing. The standard is overseen, and was made by the Object Management Group. The objective is for UML to end up a typical dialect for making models of protest arranged PC programming. In its present shape UML is contained two noteworthy segments: a Meta-display and a documentation. Later on, some type of technique or process may likewise be added to; or connected with UML.

* The Unified Modeling Language is a standard dialect for indicating, Visualization, Constructing and recording the antiques of programming framework, and additionally for business displaying and other non-programming frameworks.
* The UML speaks to an accumulation of best building rehearses that have demonstrated effective in the displaying of vast and complex frameworks.
* The UML is an imperative piece of creating articles arranged programming and the product improvement prepare. The UML utilizes for the most part graphical documentations to express the plan of programming tasks.

**GOALS**

The Primary objectives in the plan of the UML are as per the following:

1. Provide clients a prepared to-utilize, expressive visual displaying Language with the goal that they can create and trade important models.

2. Provide extendibility and specialization instruments to develop the center ideas.

3. Be free of specific programming dialects and improvement handle.

4. Provide a formal reason for comprehension the displaying dialect.

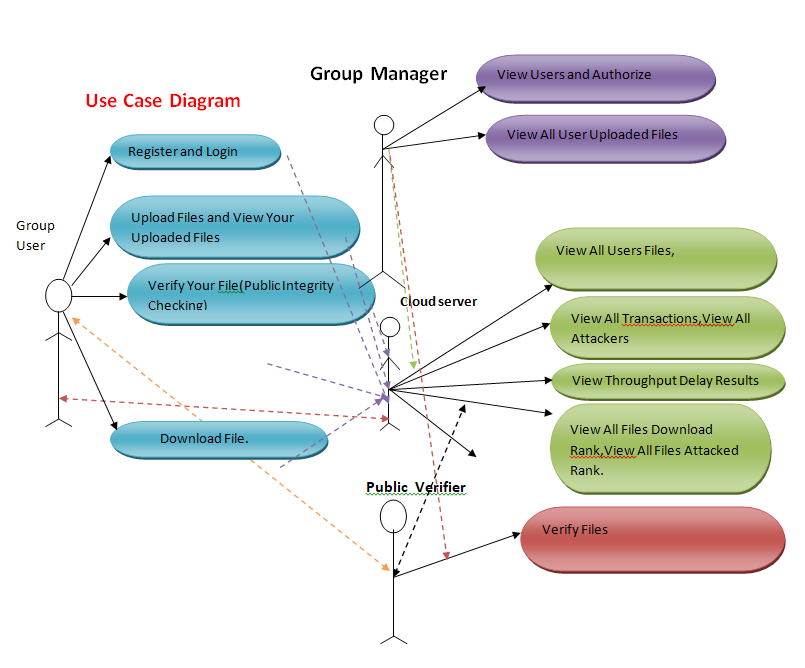
5. Encourage the development of OO devices showcase.

6. Support more elevated amount improvement ideas, for example, coordinated efforts, systems, examples and parts.

7. Integrate best practices.

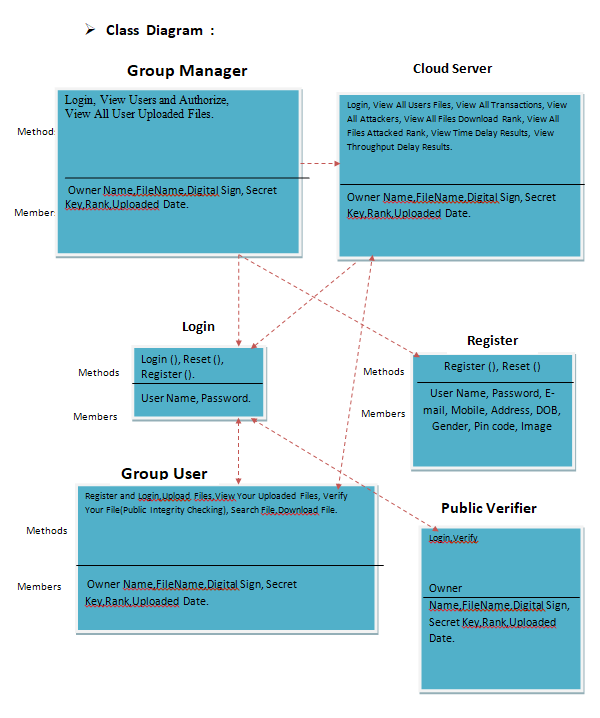
**4.2.1. USE CASE DIAGRAM**

A use case diagram within the unified modeling language (UML) may be a kind of activity diagram outlined by and created from a use-case analysis. Its purpose is to gift a graphical summary of the practicality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. the most purpose of a use case diagram is to indicate what system functions area unit performed that actor. Roles of the actors within the system is represented.

****

**Fig: 4.2. Use Case Diagram**

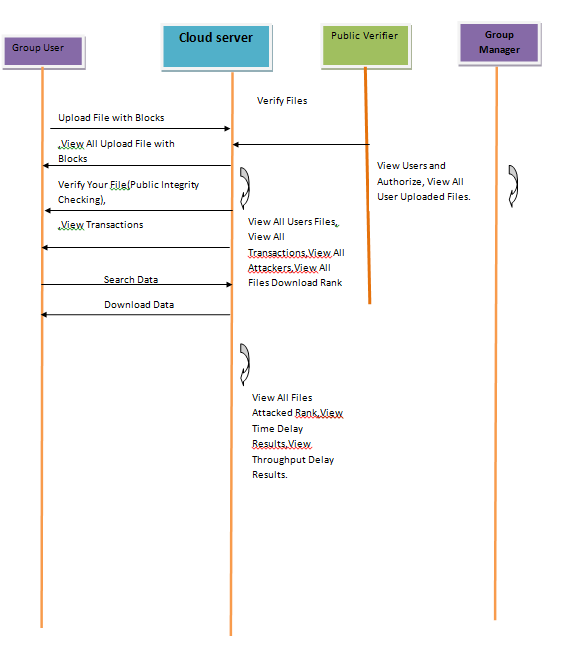
**2.2. CLASS DIAGRAM**

In computer code engineering, a category diagram within the Unified Modeling Language (UML) may be a kind of static structure diagram that describes the structure of a system by showing the system's categories, their attributes, operations (or methods), and also the relationships among the categories. It explains that category contains data.****

**Fig: 4.3. Class Diagram**

**4.2.3. SEQUENCE DIAGRAM**

A sequence diagram in Unified Modeling Language (UML) may be a quite interaction diagram that shows however processes operate with each other and in what order. it's a construct of a Message Sequence Chart. Sequence diagrams ar generally known as event diagrams, event situations, and temporal order diagrams.



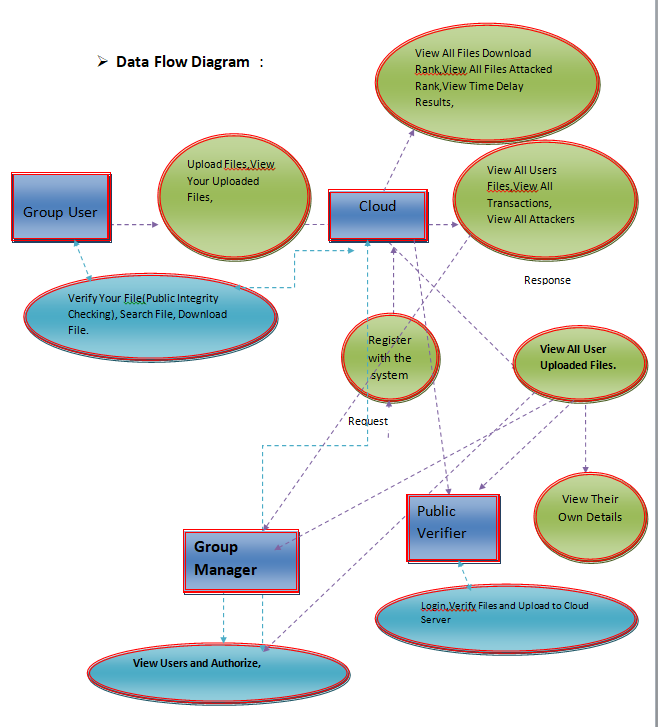
**Fig: 4.4. Sequence Diagram**

**SYSTEM DESIGN**

**Data Flow Diagram / Use Case Diagram / Flow Diagram**

The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of the input data to the system, various processing carried out on these data, and the output data is generated by the system.

**Data Flow Diagram:**

** Fig: Data Flow Diagram**

**Chapter - 5**

**SYSTEM IMPLEMENTATION**

**5.1. SYSTEM MODEL**

. Here collect 89 queries issued by the subjects, and name them as “UserQ”. As this approach might induce a bias towards topics in which lists are more useful than general web queries, we further randomly sample another set of 105 English queries from a query log of a commercial search engine, and name this set of queries as “Random”. We first ask a subject to manually create facets and add items that are covered by the query, based on his/her knowledge after a deep survey on any related resources (such as Wikipedia, Freebase, or official web sites related to the query).

**5.2. MODULE DESCRIPTION**

**Group User**

In this module, the data provider uploads their encrypted data in the Cloud server. For the security purpose the data owner encrypts the data file and then store in the server. The Data owner can have capable of manipulating the Upload Files,View Your Uploaded Files, Verify Your File(Public Integrity Checking), Search File, Download File and the user can only access the data file with the secret key. The user can search the file for a specified keyword. The data which matches for a particular keyword will be indexed in the cloud server .

**Cloud Server**

The **Cloud** server manages which is to provide data storage service for the Data Owners. Data owners encrypt their data files and store them in the Server for sharing with data consumers and performs the following operations such as View All Users Files,View All Transactions,View All Attackers,View All Files Download Rank,View All Files Attacked Rank,View Time Delay Results,View Throughput Delay Results.

**Group Manager**

In this module, the Group Manager will perform the following operations such as View Users and Authorize,View All User Uploaded Files.

**Public Verifier**

In this module, the Public Verifier performs the following operation such as Verify Files.

**5.3 SOFTWARE ENVIRONMENT**

The project experimental setup desires software system package Windows XP and

**Installation of JDK 1.6 and My SQL 6.0**

**1 Step By Step Procedure for JDK one.6**

The latest version of JDK is JDK sixu2 (Java(TM) SE Development Kit six Update 2), that's to boot referred to as Java SE six (Java traditional Edition 6). the following unit the steps to transfer and install JDK 6u2 on native machine.

**1**.Open the Java SE transfer page with this URL:

<http://java.sun.com/javase/downloads>.

**2.**Click the transfer button next to "JDK 6u2". It will' show a innovative page with a list of assorted transfer files of JDK 6u2.

**3.**Notice the "Windows Platform - Java(TM) SE Development Kit six Update 2" section.

**4**.Click the hyper link of "Windows Offline Installation (build 06), Multi-language", that links to jdk-6u2-windows-i586-p.exe with size of sixty 5.57 MB.

**5.**Save java development kitadev-6u2-windows-i586-p.exe to a quick directory.

**6.**Double-click on javadevelopmentkit-6u2-windows-i586-p.executable to start the installation wizard.

**7**.The installation wizard will guide to finish the installation.

**To test the installation, open a command window to undertake the java command. If it shows the following output, the installation was ok:**

C:\>program\java\jdk1.6.0\_02\bin\java-version

Java version \"1.6.0\_02\" Java SE Run time environment (construct one.6.0\_02-b06)

**2 Step By Step Procedure for MYSQL 6.0**

1.To place in MY SQL the file to transfer is “Windows thirty two bit and X86 Installer MSI Essentials” is required.

**2.**On the downloaded My Structured query language double click and this will the installation of My SQL.

**3**.Select "I accept terms inside the license agreement “and click on next.

**4**.Inside consequent window, select the "custom “icon to change the path of the installation.

**5**.Next window looks, presently browse or navigate the folder C:\server\mystructurequery language to place within the files in that folder.

**6**.Inside consequent window, once the modification press OK.

**7**.Press install in next window.

**8.**Select a customary pattern, and click on next.

**9.**To place within the My structure Query language service, click next.

**10**. It'd sort of a secret for MySQL information, please enter a secret, please bear in mind the login name in most cases is root and thus the key user choosing, in my case, and choose admin.

**11.**It will perform all the commands enter, thus click "run".

**12**.To setup My SQL graphical computer program, head to programs menu (My SQL) click on "My SQL Administrator" login into information as local host - username and secret head to "service control" and click on the tab "configure service" and alter the path of the my.ini file to the right location:\server\mysql\my.ini and click on apply.

**TOMCAT 6.0 WEB SERVER**

Tomcat is associate degree open supply net server developed by Apache cluster. Apache domestic cat is that the servlet instrumentation that\'s utilized in the official Reference Implementation for the Java Servlet and Java Server Pages technologies. The Java Servlet and Java Server Pages specifications area unit developed by Sun beneath the Java Community method. net Servers like Apache domestic cat support solely net parts whereas associate degree application server supports net parts additionally as business parts (BEAs net logic, is one in all the popular application server).To develop {a net an internet an online} application with jsp servlet install any web server like Run, Tomcat etc. to run your application

1.To the URL <http://tomcat.apache.org/download-70.cgi>. immediately below the second heading, see four links: KEYS, 7.0.26, Browse, and Archives.

2.By clicking on 7.0.2.6, it will be forwarded toward the bottom of the same page to a heading with the same version number. Below the version heading, see the subheading “Core”. Below that, immediately above the next subheading, see three links arranged as follows: 32-bit/64-bit Windows Service Installer (pgp,md5).

1. Click on 32-bit/64-bit Windows Service Installer to download the file apache-tomacat-7.0.26.exe
2. After agreeing to the terms of the license, the file will be downloaded and can be installed.

# 

**Fig 5.1: Tomcat Web server**

### JAVA TECHNOLOGY

Java technology is used for programming language and a platform for both.

### The Java Programming Language

### The Java programming language is a sophisticated language that can be categorized by all of the following buzzwords:

* + - Simple
    - Architecture neutral
    - Object oriented
    - Portable
    - Distributed
    - High performance
    - Interpreted
    - Multithreaded
    - Robust
    - Dynamic
    - Secure

With furthermost programming languages, you either compile the program or interpret a program, that program can run it on your computer. The Java programming language is unfamiliar in that a program is compiled and interpreted or both. Within the compiler, first you convert a program into an intermediate language called Java byte codes —the platform-independent codes interpreted by the interpreter on the Java platform. The interpreter parses and runs each Java byte code instruction on the computer. Compilation happens just once; interpretation occurs each and every time the program is executed. The following diagram explains how this works.



**Fig 5.2: Working of Java Program**

You can consider that Java byte codes as the machine code directions for the Java Virtual Machine (Java VM). Each Java translator, whether it's an advancement apparatus or a Web program that can run applets, is an utilization of the Java VM. Java byte codes make "compose once, run anyplace (WORA)" conceivable. You can assemble a program into byte codes on any stage that has a Java compiler. The byte codes can then be keep running on any use of the Java VM. That implies that the length of a PC has a Java VM, a similar program written in the Java programming dialect can keep running on whatever other Operating. Frameworks.



### ODBC

Microsoft Open Database Connectivity (ODBC) is a standard programming interface for application fashioners and database structures providers. Before ODBC transformed into an acknowledged standard for Windows tasks to interface with database structures, engineers expected to use prohibitive vernaculars for each database they expected to take up with. In a matter of seconds, ODBC has settled on the choice of the database system for all intents and purposes wrong from a coding perspective, which is as it should be. Application engineers have altogether more basic things to worry over than the sentence structure that is required to port their program beginning with one database then onto the following when business needs out of the blue change.

Through the ODBC Administrator in Control Panel, you can demonstrate the particular database that is associated with a data source that an ODBC application program is created to use. Consider an ODBC data source as a gateway with a name on it. Each portal will lead you to a particular database. For example, the data source named Sales Figures might be a SQL Server database, while the Accounts Payable data source could suggest an Access database. The physical database suggested by a data source can live wherever on the LAN.

The ODBC structure records are not presented on your system by Windows 95. On the other hand perhaps, they are presented when you setup an alternate database application, for instance, SQL Server Client or Visual Basic 4.0. Right when the ODBC image is presented in Control Panel, it uses a record called ODBCINST.DLL. It is in like manner possible to deal with your ODBC data sources through a stay single program called ODBCADM.EXE. There is a 16-bit and a 32-bit version of this program and every keeps up an alternate once-over of ODBC data sources.

From a programming perspective, the appeal of ODBC is that theapplication can be created to use a comparable course of action of limit calls to interface with any data source, paying little mind to the database trader. The source code of the application doesn't change whether it banters with Oracle or SQL Server. We simply determine these two for example. There are ODBC drivers available for a couple of dozen understood database structures. For sure, even Excel spreadsheets and plain substance records can be changed into data sources.

The working system uses the Registry information made by ODBC Administrator to make sense of which low-level ODBC drivers are relied upon to chat with the data source, (for instance, the interface to Oracle or SQL Server). The stacking of the ODBC drivers is clear to the ODBC application program. In a client/server environment, the ODBC API even handles extensive segments of the framework issues for the application programming engineer.

The advantages of this arrangement are various to the point that you are apparently thinking there must be some catch. The fundamental weight of ODBC is that it isn't as powerful as talking particularly to the neighborhood database interface. ODBC has had various depreciators make the charge that it is excessively direct. Microsoft has constantly sued that the essential compute execution is the way of the driver programming that is used. As we might want to think, this is legitimate. The availability of good ODBC drivers has upgraded an amazing game plan starting late. Moreover, at any rate, the input about execution is to some degree like the people who said that compilers would never facilitate the speed of unadulterated low level processing develop. Maybe not, yet rather the compiler (or ODBC) allows you to form cleaner programs, which infers you finish sooner. Meanwhile, PCs get speedier reliably.

**JDBC**

With a ultimate objective to set a free database standard API for Java; Sun Microsystems made Java Database Connectivity, or JDBC. JDBC offers a non particular SQL database get the chance to instrument that gives a consistent interface to a variety of RDBMSs. This trustworthy interface is winning utilizing "module" database accessibility modules, or drivers. In case a database trader wishes to have JDBC support, he or she ought to give the driver to each phase that the database and Java continue running on.

To extend a more broad affirmation of JDBC, Sun develop JDBC's structure in light of ODBC. As you revealed before in this segment, ODBC has no matter how you look at it support on a collection of stages. Developing JDBC as for ODBC will allow venders to offer JDBC drivers available to be purchased to people in general much faster than working up an absolutely new accessibility game plan. JDBC was accounted for in March of 1996. It was released for a 90 day open review that completed June 8, 1996. In light of customer information, the last JDBC v1.0 assurance was released a little while later.

Whatever is left of this range will cover enough information about JDBC for you to grasp what it is about and how to use it effectively. This is by no means whatsoever, an aggregate survey of JDBC. That would fill an entire book.

**JDBC Goals**

Few programming groups are sketched out without goals as an essential concern. JDBC is one that, because of its various destinations, drove the change of the API. These destinations, in conjunction with early examiner input, have closed the JDBC class library into a solid framework for building database applications in Java. The targets that were set for JDBC are crucial. They will give you some comprehension in the matter of why certain classes and functionalities bear in transit they do. The eight diagram goals for JDBC are according to the accompanying:

1.SQL Level API The organizers felt that their essential target was to describe a SQL interface for Java. In spite of the way that not the most negligible database interface level possible, it is at an adequately low level for more raised sum instruments and APIs to be made. On the other hand, it is at an adequately abnormal state for application designers to use it surely. Fulfilling this target considers future instrument dealers to "convey" JDBC code and to disguise countless difficulties from the end customer.

2. SQL Conformance SQL punctuation vacillates as you move from database vendor to database vender. With a true objective to reinforce a wide arrangement of shippers, JDBC will allow any request announcement to be experienced it to the concealed database driver. This allows the system module to handle non-standard handiness in a way that is sensible for its customers.

3.JDBC must be implemental on top of essential database interfaces The JDBC SQL API must "sit" on top of other normal SQL level APIs. This target grants JDBC to use existing ODBC level drivers by the usage of an item interface. This interface would make a translation of JDBC calls to ODBC and the a different way.

4. Give a Java interface that is unsurprising with whatever is left of the Java system

In perspective of Java's affirmation in the customer aggregate as such, the organizers feel that they should not stray from the present diagram of the middle Java structure.

5.Keep it clear This target probably appears in all item plot target postings. JDBC is no unique case. Sun felt that the layout of JDBC should be extraordinarily clear, considering one and procedure for completing a task for every segment. Allowing duplicate convenience just serves to overwhelm the customers of the API.

6.Use in number, static composition wherever possible Strong composition considers more bumble checking to be done at amass time; also, less mix-up appear at runtime.

7.Keep the typical cases direct.

**What is JSP?**

JSP could be a technology that adds dynamic content to sites. In absence of JSP, to update the looks or the content of plain static hypertext mark-up language pages, it should be done by hand. even though an image must be modified, the hypertext mark-up language file should be emended for modifications.

Whereas with JSP, the content obsessed with several factors are often created, as well as the time of the day, the knowledge provided by the user, the user’s history of interaction together with your computing machine, and even the user’s browser sort.

This capability is crucial to supply on-line services which may be tailored to the viewer World Health Organization created the request, betting on the viewer’s preferences and needs. a vital facet of providing purposeful on-line services is for the system to be able to keep in mind knowledge related to the service and its users. That’s why databases play a vital role in dynamic sites.

**Viewing a JSP Page**

With JSP, the online page doesn’t really exist on the server.

The following steps justify however the online server creates the online page:

1.Like a traditional page, the browser sends associate protocol request to the online server. This doesn’t modification with JSP, though the uniform resource locator in all probability ends in .jsp rather than .html .

2.The online server isn't a traditional server, however rather a Java server, with the extensions necessary to spot and handle Java servlets. the online server acknowledges that the protocol request is for a JSP page and forwards it to a JSP engine.

3.The JSP engine masses the JSP page from disk and converts it into a Java servlet. From now on, this servlet is indistinguishable from the other servlet developed directly in Java instead of JSP, though the mechanically generated Java code of a JSP servlet isn't perpetually straightforward to browse, and you must ne\'er modify it by hand.

4.The JSP engine compiles the servlet into associate possible category and forwards the initial request to a different a part of the online server referred to as the servlet engine. Note that the Jsp engine solely converts the JSP page to Java and recompiles the servlet if it finds that the JSP page has modified since the last request. This makes the method a lot of economical than with alternative scripting languages and thus quicker.

5.The servlet engine masses the servlet category and executes it. throughout execution, the servlet produces associate output in hypertext mark-up language format, that the servlet engine passes to the online server within associate protocol response.

6.The online server forwards the protocol response to your browser.

7.Your application program handles the dynamically generated hypertext mark-up language page within the protocol response specifically as if it were a static page. In fact, static and dynamic sites square measure within the same format.

**Chapter-6**

**SYSTEM TESTING**

Testing is that the debugging program is one amongst the leading crucial aspects of the pc programming triggers, while not programming that works, the system would ne'er turn out relate in Nursing output of that it had been designed .Testing is best performed once user development is asked to help in characteristic all errors and bugs. The sample knowledge are used for testing. It is not amount however quality of the information used the matters of testing .Testing is aimed toward guaranteeing that the system was accurately relate in Nursing with efficiency before live operation commands.

**Testing objectives:** The most objective of testing is to uncover a bunch of errors, consistently and with minimum effort and time. Stating formally ,we can say, testing may be a method of corporal punishment a program with intent of finding miscalculation.

1. A productive check is one that uncovers Associate in Nursing hitherto undiscovered error.

2. A decent legal action is one that has likelihood of finding miscalculation, if it exists.

3. The check is insufficient to find probably gift errors.

4. The code additional or less confirms to the standard and reliable standards.

**6.1. TYPES OF TESTING**

**6.1.1. UNIT TESTING**

Unit testing we have a tendency to test every module separately and integrate with the general system. Unit testing focuses verification efforts on the littlest unit of code style within the module. this is often conjointly called module testing.

The module of the system is tested individually. as an example the validation check is completed for variable the user input given by the user that validity of the information entered. it's terribly straightforward to search out error rectify the system. Every Module will be tested victimization the subsequent 2 Strategies: recording machine Testing and White Box Testing.

**6.1.2. BLACK BOX TESTING**

Recording machine checking may be a code testing techniques during which practicality of the code below test (SUT) is tested while not staring at the interior code structure, implementation details and data of internal ways of the code .This type of testing is predicated entirely on the code needs and specifications .In recording machine Testing we have a tendency to simply concentrate on inputs and output of the package while not bothering concerning internal data of the code program. The on top of recording machine will be any package you wish to check. For example, Associate in Nursing software like Windows, a web site like Google ,a information like Oracle or maybe your own custom application. Under recording machine testing, you can check these applications by simply that specialize in the inputs and outputs while not knowing their internal code implementation.

**Types of Black Box Testing**

There are many varieties of recording machine Testing however following ar the outstanding ones.

**• Functional testing:** This recording machine testing kind is said to purposeful needs of a system; it's done by code testers.

**• Non-Functional testing:** This sort of recording machine testing isn't associated with testing of a selected practicality, however non-functional needs like performance, measurability, usability.

**• Regression testing:** Regression testing is completed once code fixes, upgrades or the other system maintenance to visualize the new code has not affected the prevailing code.

**6.1.3. WHITE BOX TESTING**

White Box Testing is that the testing of a code solution's internal committal to writing and infrastructure. It focuses totally on Traffic Redundancy Elimination ngthening security, the flow of inputs and outputs through the applying, and rising style and value. White box testing is additionally called clear, open, structural, and glass box testing. It is one amongst 2 elements of the "box testing" approach of code testing.

**System Testing:**

Once the individual module testing is completed, modules are assembled and integrated to perform as a system. The top down testing, that began from higher level to lower level module, was allotted to visualize whether or not the whole system is playacting satisfactorily. There are 3 main types of System testing: Alpha Testing, Beta Testing, Acceptance Testing.

**Alpha Testing:** This refers to the system checking that's allotted by the test team with the Organization.

**Beta Testing:** This refers to the system testing that's performed by a particular cluster of friendly customers.

**Acceptance Testing:** This refers to the system testing that's performed by the client to see whether or not or to not settle for the delivery of the system.

**6.2. 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.

**Test Scenarios**

General Scenarios

• All mandatory fields should be validated and indicated by asterisk (\*) symbol

• Validation error messages should be displayed properly at correct position

• All error messages should be displayed in same CSS style (e.g. using red color)

• General confirmation messages should be displayed using CSS style other than

error messages style (e.g. using green color)

• Dropdown fields should have first entry as blank or text like ‗Select

• Delete functionality for any record on page should ask for confirmation

**GUI and Usability Test Scenarios**

• All fields on page (e.g. text box, radio options, dropdown lists) should be aligned properly.

• Scroll bar should be enabled only when necessary

•Description text box should be multi-line User should be able to submit the form again by correcting the errors

• Default radio options should be pre-selected on page load

• Check all pages for broken images

**Test Scenarios for a Window**

• Check if default window size is correct

• Check if child window size is correct

• Check if child windows are getting closed on closing parent/opener window

• Check window minimize, maximize and close functionality

• Check if window is re-sizable

**Database Testing Test Scenarios**

* Check if correct data is getting saved in database upon successful page submit
* Check values for columns which are not accepting null values
* Check for data integrity. Data should be stored in single or multiple tables based on design
* For every database add/update operation log should be added
* Required table indexes should be created.

**Security Testing Test Scenarios**

• Secure pages should use HTTPS protocol

• Check application logout functionality

• Check for Brute Force Attacks

6.3. TEST CASE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test case id** | **Test case description** | **Actual value** | **Entered value** | **Status** |
| **1** | Register user details in registration page | Fill all the fields while registering user | All the fields are filled | Pass |
| **2** | Give user name in text box | User name must be given in alphabets | User name given in alphabets and numeric values | Fail |
| **3** | Password to be entered in password box | Password must be given correctly | Password is entered wrongly | Fail |
| **4** | Phone number must be entered in phone number box during registration | Phone number must be given in 10 digits | Phone number given in 10 digits | Pass |

Table: 6.1. Test case Template

CODE

<title>Authentication</title>

<%@page import="java.io. BufferedInputStream"%>

<%@page import="java. security. DigestInputStream"%>

<%@page import="java.io. FileInputStream"%>

<%@page import="java.io. Print Stream"%>

<%@page import="java.io. FileOutputStream"%>

<%@page import="java. math. Big Integer"%>

<%@ page

import="java. security. Key, java. security. Key Pair, java. security. Keypair Generator, java. crypto. Cipher"%>

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

<%@page

import="java.util.\*,java.security.Key,java.util.Random,javax.crypto.Cipher,javax.crypto.spec.SecretKeySpec,org.bouncycastle.util.encoders.Base64"%>

<%@page import="java.security.MessageDigest"%>

<%@page import="java.text.SimpleDateFormat"%>

<style type="text/css">

<!--

.style2 {font-size: 18px}

.style3 {

font-size: 18px;

font-weight: bold;

color: #FF0000;

}

.style4 {font-size: 18px; font-weight: bold; }

-->

</style>

<%

try

{

String value=request.getParameter("value");

if(value.equalsIgnoreCase("userlogin"))

{

String username=request.getParameter("username");

String password=request.getParameter("password");

String sql = "SELECT \* FROM user where username='"+username+"' and password='"+password+"'";

Statement stmt = connection.createStatement();

ResultSet rs = stmt.executeQuery(sql);

if (rs.next()==true)

{

String sql1="SELECT \* FROM user where username='"+username+"' and status='Authorized' ";

Statement stmt1 = connection.createStatement();

ResultSet rs1 =stmt1.executeQuery(sql1);

if(rs1.next()==true)

int i = rs1.getInt(1);

String gkey=rs.getString(13); application.setAttribute("uname", username); application.setAttribute("gkey"gkeyresponse.sendRedirect("tenant1\_main.js);

}

else

{ <br/>

<palign="center“class="style3">You are not Authorized by the Verifier, Please wait!! </p><p align="center" class="style3"><br/>

<ahref="tenant1\_login.jsp" class="style4">Back</a></p>

}

}

else

{response.sendRedirect("U\_Wrong\_Login.jsp");

}

}

else if(value.equalsIgnoreCase("userstatus"))

{

int id=Integer.parseInt(request.getParameter("id"));

String sql1="update user set status='Authorized' where id='"+id+"'";

Statement stmt1 = connection.createStatement();

int k =stmt1.executeUpdate(sql1);

if(k>0)

{ response.sendRedirect("A\_AuthorizeUsers.jsp"); } else

{ %>

</p>

<p align="center"class="style3">Permision Not Granted..Please Try Again..<br/>

<br/>

</p>

<p align="center" class="style3"><ahref="A\_AuthorizeUsers.jsp" class="style2">Back</a>

</p>

<%

} }

else if(value.equalsIgnoreCase("groupkey"))

{

int id=Integer.parseInt(request.getParameter("id"));

String gname=request.getParameter("gname");

//Generate Group Key

Stringnamefile="filename.txt";

Instream=newPrintStream(new FileOutputStream(namefile));

p.print(newString(gname));

MessageDigestmd=MessageDigest.getInstance("SHA1");

FileInputStreamfis11=new FileInputStream(namefile); DigestInputStreamdis1=new DigestInputStream(fis11,md);

BufferedInputStreambis1=new BufferedInputStream(dis1); while(true)

{

b1=bis1.read();

if(b1==-1)

break;

}

BigIntegerbig1=newBigInteger(md.digest());

String spl1=bi1.toString();

String y=bi1.toString(16); String sql1="updateusesetgsign='"+gkey+"'where id='"+id+"'";

Statement stmt1 = connection.createStatement();

int k =stmt1.executeUpdate(sql1);

if(k>0)

{

response.sendRedirect("A\_AuthorizeUsers.jsp");

}

else

{ %>

</p>

<p align="center" class="style3">Permision Not Granted..Please Try Again..<br/>

<br/>

</p>

<p align="center" class="style3"><a href="A\_AuthorizeUsers.jsp" class="style2">Back</a>

</p>

<%

} }

else

{}

}

catch(Exception e)

{

out.print(e);

}

%>

</p>

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

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<title>Verifier Authorize Users</title>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link href="css/style.css" rel="stylesheet" type="text/css" />

<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />

<script type="text/javascript" src="js/cufon-yui.js"></script>

<script type="text/javascript" src="js/cufon-chunkfive.js"></script>

<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>

<script type="text/javascript" src="js/script.js"></script>

<script type="text/javascript" src="js/coin-slider.min.js"></script>

<style type="text/css">

<!--

.style1 {font-size: 36px}

.style2 {font-size: 16px}

.style3 {color: #FFFFFF}

.style4 {color: #FF00FF}

.style5 {color: #0000FF}

.style7 {font-weight: bold}

-->

</style>

</head>

<body>

<div class="main">

<div class="header">

<div class="header\_resize">

<div class="menu\_nav">

<ul>

<li><a href="index.html"><span>Home Page</span></a></li>

<li class="active"><a href=""><span>Cloud</span></a></li>

<li><a href="U\_Login.jsp"><span>User</span></a></li>

</ul>

</div>

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

<div class="logo">

<h1><a href="index.html" class="style1">Certificateless public integrity checking of group shared data on cloud storage</a></h1>

</div>

<div class="searchform"></div>

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

<div class="slider">

<div id="coin-slider"> <a href="#"><img src="images/slide1.jpg" width="960" height="360" alt="" /><span><big>Privacy Preserving Electronic Ticket Scheme with Attribute Based Credentials</big></span></a> <a href="#"><img src="images/slide2.jpg" width="960" height="360" alt="" /><span><big>Privacy Preserving Electronic Ticket Scheme with Attribute Based Credentials</big></span></a> <a href="#"><img src="images/slide3.jpg" width="960" height="360" alt="" /><span><big>Privacy Preserving Electronic Ticket Scheme with Attribute Based Credentials</big></span></a> </div>

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

</div>

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

</div>

</div>

<div class="content">

<div class="content\_resize">

<div class="mainbar">

<div class="article">

<h2><span class="style4">View Users and Authorize</span></h2>

<p>&nbsp;</p>

<table width="913" border="1" align="center" cellpadding="0" cellspacing="0" >

<tr>

<td width="51" height="37" valign="middle" bgcolor="#00FFFF" style="color: #2c83b0;"><div align="center" class="style27 style105 style65 style2"><em><strong>ID</strong></em></div></td>

<td width="157" valign="middle" bgcolor="#00FFFF" style="color: #2c83b0;"><div align="center" class="style27 style105 style65 style2"><em><strong>User Image</strong></em></div></td>

<td width="121" valign="middle" bgcolor="#00FFFF" style="color: #2c83b0;"><div align="center" class="style27 style105 style65 style2"><em><strong>User Name</strong></em></div></td>

<td width="142" valign="middle" bgcolor="#00FFFF" style="color: #2c83b0;"><div align="center" class="style27 style105 style65 style2"><em><strong>Email</strong></em></div></td>

<td width="135" valign="middle" bgcolor="#00FFFF" style="color: #2c83b0;"><div align="center" class="style27 style105 style65 style2"><em><strong>Mobile</strong></em></div></td>

<td width="121" valign="middle" bgcolor="#00FFFF" style="color: #2c83b0;"><div align="center" class="style27 style105 style65 style2"><em><strong>Location</strong></em></div></td>

<td width="108" valign="middle" bgcolor="#00FFFF" style="color: #2c83b0;"><div align="center" class="style27 style105 style65 style2"><em><strong>Group Name</strong></em></div></td>

<td width="108" valign="middle" bgcolor="#00FFFF" style="color: #2c83b0;"><div align="center" class="style27 style105 style65 style2"><em><strong>Status</strong></em></div></td>

<td width="108" valign="middle" bgcolor="#00FFFF" style="color: #2c83b0;"><div align="center" class="style27 style105 style65 style2"><em><strong>Group Key Status</strong></em></div></td>

</tr>

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

<%

String s1,s2,s3,s4,s5,s6,s7,s8;

int i=0;

try

{

String query="select \* from user";

Statementst=connection.createStatement();

ResultSetrs=st.executeQuery(query);

while ( rs.next() )

{

i=rs.getInt(1);

s1=rs.getString(2);

s2=rs.getString(4);

s5=rs.getString(5);

s3=rs.getString("location");

s4=rs.getString(10);

s6=rs.getString(9);

s8=rs.getString(12);

s7=rs.getString(13); %>

<tr bgcolor="#99CCFF">

<td height="122" align="center" valign="middle" bgcolor="#FF0000"><div align="center" class="style3 style71 style70 style37 style54 style55 style86 style57"><strong>

<%out.println(i);%>

</strong></div></td>

<td rowspan="1" align="center" valign="middle" bgcolor="#FF0000" ><div class="style3 style71 style70 style37 style54 style55 style86 style57" style="margin:10px 13px 10px 13px;" ><strong>

<input name="image" type="image" src="images.jsp?id=<%=i%>" style="width:120px; height:100px;" />

</strong></div></td>

<td align="center" valign="middle" bgcolor="#FF0000"><div align="center" class="style3 style71 style70 style20 style37 style54 style55 style86 style57"><strong>

<%out.println(s1);%>

</strong></div></td>

<td align="center" valign="middle" bgcolor="#FF0000"><div align="center" class="style3 style71 style70 style20 style37 style54 style55 style86 style57"><strong>

<%out.println(s2);%>

</strong></div></td>

<td align="center" valign="middle" bgcolor="#FF0000"><div align="center" class="style3 style71 style70 style20 style37 style54 style55 style86 style57"><strong>

<%out.println(s5);%>

</strong></div></td>

<td align="center" valign="middle" bgcolor="#FF0000"><div align="center" class="style3 style71 style70 style20 style37 style54 style55 style86 style57"><strong>

<%out.println(s6);%>

</strong></div></td>

<td align="center" valign="middle" bgcolor="#FF0000"><div align="center" class="style3 style71 style70 style20 style37 style54 style55 style86 style57"><strong>

<%out.println(s8);%>

</strong></div></td>

<% if(s4.equalsIgnoreCase("waiting"))

{

%>

<td align="center" valign="middle" bgcolor="#FF0000" style="color:#000000;"><div align="center" class="style20 style30 style37 stylle70 style71 style3 style7">

<div align="center"><a href="Authentication.jsp?value=<%="userstatus"%>&amp;id=<%=i%>" class="style5">waiting</a></div>

</div></td>

<%

}

else

{

%>

<td width="76" align="center" valign="middle" bgcolor="#FF0000"><div align="center" class="style3 style72 style71 style30 style55 style37 style20"><strong>

<%out.println(s4);%>

</strong></div></td>

<%

}

%>

<%

if(s7.equalsIgnoreCase("Not Assigned"))

{

%>

<td align="center" valign="middle" bgcolor="#FF0000" style="color:#000000;"><div align="center" class="style20 style30 style37 style86 style70 style71 style3 style7">

<div align="center"><a href="Authentication.jsp?value=<%="groupkey"%>&amp;id=<%=i%>&amp;gname=<%=s8%>" class="style5">Generate</a></div>

</div></td>

<%

}

else

{

%>

<td width="76" align="center" valign="middle" bgcolor="#FF0000"><div align="center" class="style3 style72 style71 style30 style55 style37 style20"><strong>

<%out.println(s7);%>

</strong></div></td>

<%

}

%>

</tr>

<%

}

connection.close();

}

catch(Exception e)

{

out.println(e);

}

%>

</table>

<p align="right"><a href="tenant2\_main.jsp">Back</a></p>

</div>

</div>

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

</div>

</div>

<div class="fbg">

<p>&nbsp;</p>

<p>&nbsp;</p>

<p>&nbsp;</p>

<p>&nbsp;</p>

</div>

<div class="footer">

<div class="footer\_resize">

<div style="clear:both;"></div>

</div>

</div>

</div>

<div align=center></div>

</body>

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

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<title>Attacker</title>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link href="css/style.css" rel="stylesheet" type="text/css" />

<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />

<script type="text/javascript" src="js/cufon-yui.js"></script>

<script type="text/javascript" src="js/cufon-georgia.js"></script>

<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>

<script type="text/javascript" src="js/script.js"></script>

<script type="text/javascript" src="js/coin-slider.min.js"></script>

<script language="javascript" type='text/javascript'>

function loadFileAsText()

{

var fileToLoad = document.getElementById("file").files[0];

var fileReader = new FileReader();

fileReader.onload = function(fileLoadedEvent)

{

var textFromFileLoaded = fileLoadedEvent.target.result;

document.getElementById("textarea").value = textFromFileLoaded;

};

fileReader.readAsText(fileToLoad, "UTF-8");

}

</script>

<style type="text/css">

<!--

.style3 {font-size: 26px}

.style7 {color: #0000FF}

-->

</style>

</head>

<body>

<div class="main">

<div class="header">

<div class="header\_resize">

<div class="logo">

<h1><a href="index.html" class="style1 style3">Certificateless public integrity checking of group shared data on cloud storage</a></h1>

</div>

<div class="searchform"></div>

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

<div class="menu\_nav">

<ul>

<li><a href="index.html"><span>Home Page</span></a></li>

<li><a href="server\_login.jsp"><span> Cloud Server</span></a></li>

<li><a href="crms\_login.jsp"><span>Public Verifier </span></a></li>

<li><a href="tenant1\_login.jsp"><span>Group Users</span></a></li>

<li><a href="tenant2\_login.jsp"><span>Group Manager</span></a></li>

<li class="active"><a href="attack.jsp"><span>Attack File</span></a></li>

</ul>

</div>

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

<div class="slider">

<div id="coin-slider"> <a href="#"><img src="images/slide1.jpg" width="960" height="360" alt="" /></a> <a href="#"><img src="images/slide2.jpg" width="960" height="360" alt="" /></a> <a href="#"><img src="images/slide3.jpg" width="960" height="360" alt="" /></a> </div>

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

</div>

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

</div>

</div>

<div class="content">

<div class="content\_resize">

<div class="mainbar">

<div class="article">

<h2 align="center"><span>Attack File </span></h2>

<p>&nbsp;</p>

<p>&nbsp;</p>

<form action="attack1.jsp" method="post" name="form1" id="form1">

<table border="0" align="center">

<tr>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

<tr>

<td width="223"><span class="style1 style7">Enter File Name :-</span></td>

<td width="245"><label> <input required name="t1"

type="text" size="40" /> </label></td>

</tr>

<tr>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

<tr> <td><span class="style1 style7">Your Name :-</span></td>

<td><input name="ot" type="text" size="40" /></td>

</tr>

<tr> <td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

<tr>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

<tr>

<td>

<div align="right"><input type="submit" name="Submit"

value="Attack" /></div>

</td>

</tr>

</table>

</form>

<p align="center" class="style16"><a href="index.html" class="style16">Back</a></p>

</div>

</div>

<div class="sidebar">

<div class="gadget">

<h2 class="star"><span>Side</span> Menu</h2>

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

<ul class="sb\_menu">

<li><a href="index.html"><span>Home Page </span></a></li>

<li><a href="attack.jsp"><span>Log Out</span></a></li>

</ul>

</div>

</div>

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

</div>

</div>

<div class="fbg"></div>

<div class="footer">

<div class="footer\_resize">

<div style="clear:both;"></div>

</div>

</div>

</div>

<div align=center></div>

</body>

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

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<title>Attacker</title>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link href="css/style.css" rel="stylesheet" type="text/css" />

<link rel="stylesheet" type="text/css" href="css/coin-slider.css" />

<script type="text/javascript" src="js/cufon-yui.js"></script>

<script type="text/javascript" src="js/cufon-georgia.js"></script>

<script type="text/javascript" src="js/jquery-1.4.2.min.js"></script>

<script type="text/javascript" src="js/script.js"></script>

<script type="text/javascript" src="js/coin-slider.min.js"></script>

<script language="javascript" type='text/javascript'>

function loadFileAsText()

{

var fileToLoad = document.getElementById("file").files[0];

var fileReader = new FileReader();

fileReader.onload = function(fileLoadedEvent)

{

var textFromFileLoaded = fileLoadedEvent.target.result;

document.getElementById("textarea").value = textFromFileLoaded;

};

fileReader.readAsText(fileToLoad, "UTF-8");

}

</script>

<style type="text/css">

<!--

.style3 {font-size: 26px}

.style4 {

font-size: 40px;

color: #FF0000;

font-weight: bold;

}

-->

</style>

</head>

<body>

<div class="main">

<div class="header">

<div class="header\_resize">

<div class="logo">

<h1><a href="index.html" class="style1 style3">Certificateless public integrity checking of group shared data on cloud storage</a></h1>

</div>

<div class="searchform"></div>

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

<div class="menu\_nav">

<ul>

<li><a href="index.html"><span>Home Page</span></a></li>

<li><a href="server\_login.jsp"><span> Cloud Server</span></a></li>

<li><a href="crms\_login.jsp"><span>Public Verifier </span></a></li>

<li><a href="tenant1\_login.jsp"><span>Group User</span></a></li>

<li><a href="tenant2\_login.jsp"><span>Group Manager</span></a></li>

<li class="active"><a href="attack.jsp"><span>Attack File</span></a></li>

</ul>

</div>

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

<div class="slider">

<div id="coin-slider"> <a href="#"><img src="images/slide1.jpg" width="960" height="360" alt="" /></a> <a href="#"><img src="images/slide2.jpg" width="960" height="360" alt="" /></a> <a href="#"><img src="images/slide3.jpg" width="960" height="360" alt="" /></a> </div>

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

</div>

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

</div>

</div>

<div class="content">

<div class="content\_resize">

<div class="mainbar">

<div class="article">

<h2 align="center"><span>Attack File </span></h2>

<p>&nbsp;</p>

<p>&nbsp;</p>

<p align="justify">

<%@page import="java.util.\*"%>

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

<%@page

import="java.util.\*,java.security.Key,java.util.Random,javax.crypto.Cipher,javax.crypto.spec.SecretKeySpec,org.bouncycastle.util.encoders.Base64"%>

<%@ page

import="java.sql.\*,java.util.Random,java.io.PrintStream,java.io.FileOutputStream,java.io.FileInputStream,java.security.DigestInputStream,java.math.BigInteger,java.security.MessageDigest,java.io.BufferedInputStream"%>

<%@ page

import="java.security.Key,java.security.KeyPair,java.security.KeyPairGenerator,javax.crypto.Cipher"%>

<%@page

import="java.util.\*,java.text.SimpleDateFormat,java.util.Date,java.io.FileInputStream,java.io.FileOutputStream,java.io.PrintStream"%>

<%

int i;

String s2, s3, s4, s5, s6, s7, s12="", s13="", s14="", s15="", s16="";

try {

String file = request.getParameter("t1");

String on = request.getParameter("t12");

String cont = request.getParameter("text");//contents

String keys = "ef50a0ef2c3e3a5f";

String h1 = "";

String filename = "filename.txt";

i = 1;

byte[] keyValue = keys.getBytes();

Key key = new SecretKeySpec(keyValue, "AES");

Cipher c = Cipher.getInstance("AES");

c.init(Cipher.ENCRYPT\_MODE, key);

String encryptedValue = new String(Base64.encode(cont.getBytes()));

//D:/Workspace Jayam/SelCSp/WebContent

PrintStream p = new PrintStream(new FileOutputStream("D:/tom/webapps/Certificateless public integrity checking of group shared data on cloud storage/"+ filename));

p.print(new String(cont));

MessageDigest md = MessageDigest.getInstance("SHA1");

FileInputStream fis11 = new FileInputStream("D:/tom/webapps/Certificateless public integrity checking of group shared data on cloud storage/"+ filename);

DigestInputStream dis1 = new DigestInputStream(fis11, md);

BufferedInputStream bis1 = new BufferedInputStream(dis1);

//Read the bis so SHA1 is auto calculated at dis

while (true) {

int b1 = bis1.read();

if (b1 == -1)

break;

}

BigInteger bi1 = new BigInteger(md.digest());

String spl1 = bi1.toString();

h1 = bi1.toString(16);

KeyPairGenerator kg = KeyPairGenerator.getInstance("RSA");

Cipher encoder = Cipher.getInstance("RSA");

KeyPair kp = kg.generateKeyPair();

Key pubKey = kp.getPublic();

byte[] pub = pubKey.getEncoded();

String pk = String.valueOf(pub);

String type = "Malicious Data Attack";

SimpleDateFormatsdfDatenewSimpleDateFormat("dd/MM/yyyy");

SimpleDateFormatsdfTimenewSimpleDateFormat("HH:mm:ss");

Date now = new Date();

String strDate = sdfDate.format(now);

String strTime = sdfTime.format(now);

String dt = strDate + " " + strTime;

StringstrQuery2="updatecloudserversetct='"+cont+"',mac='" + h1 + "' where fname='" + file + "' ";

connection.createStatement().executeUpdate(strQuery2);

String strQuery25 = "insert into attacker(user,fname,sk,type,dt) values('"+ on+ "','"+ file+ "','"+ pk+ "','"+ type+ "','" + dt + "')";

connection.createStatement().executeUpdate(strQuery25);

%>

<p>

<h1 class="style4" >Attacked Successfully!!!</h1>

</p>

<br />

<%

connection.close();

} catch (Exception e) {

out.println(e.getMessage());

e.printStackTrace();

}

%>

</p>

<p>&nbsp;</p>

<p><a href="attack.jsp" class="style16">Back</a></p>

<p>&nbsp;</p>

</div>

</div>

<div class="sidebar">

<div class="gadget">

<h2 class="star"><span>Side</span> Menu</h2>

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

<ul class="sb\_menu">

<li><a href="index.html"><span>Home Page </span></a></li>

<li><a href="attack.jsp"><span>Log Out</span></a></li>

</ul>

</div>

</div>

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

</div>

</div>

<div class="fbg"></div>

<div class="footer">

<div class="footer\_resize">

<div style="clear:both;"></div>

</div>

</div>

</div>

<div align=center></div>

</body>

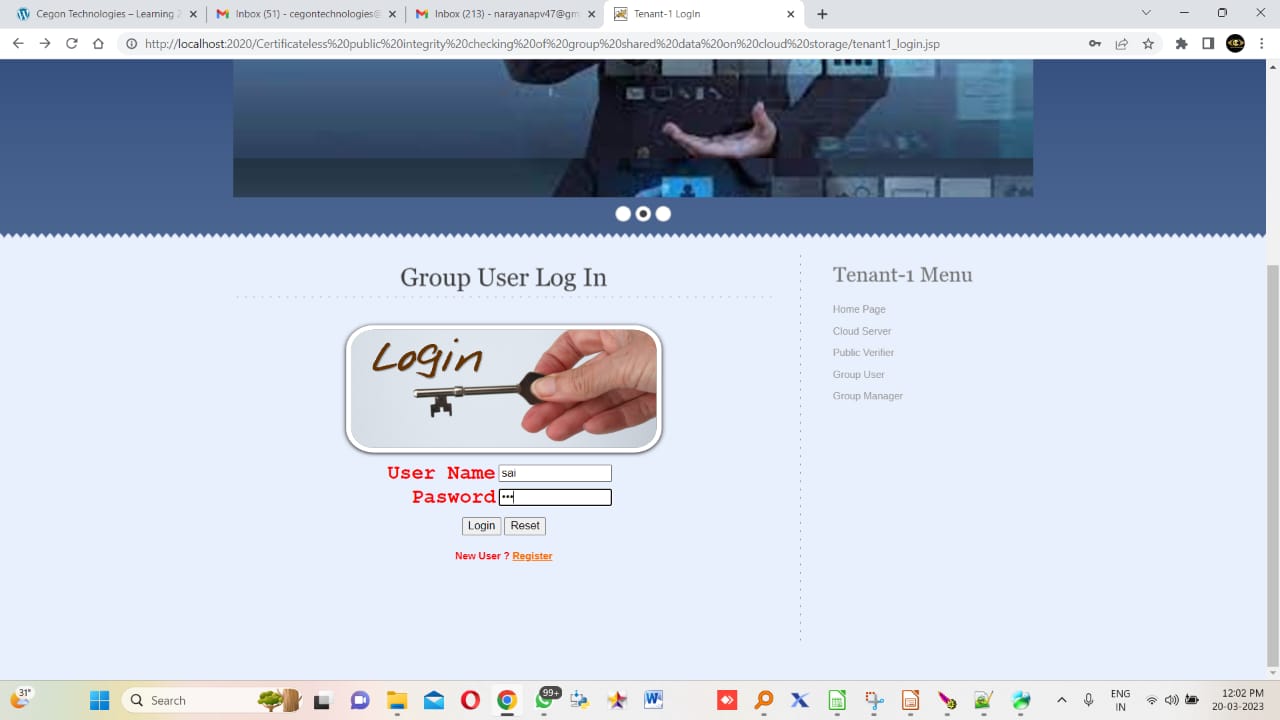
</html>

Chapter - 7

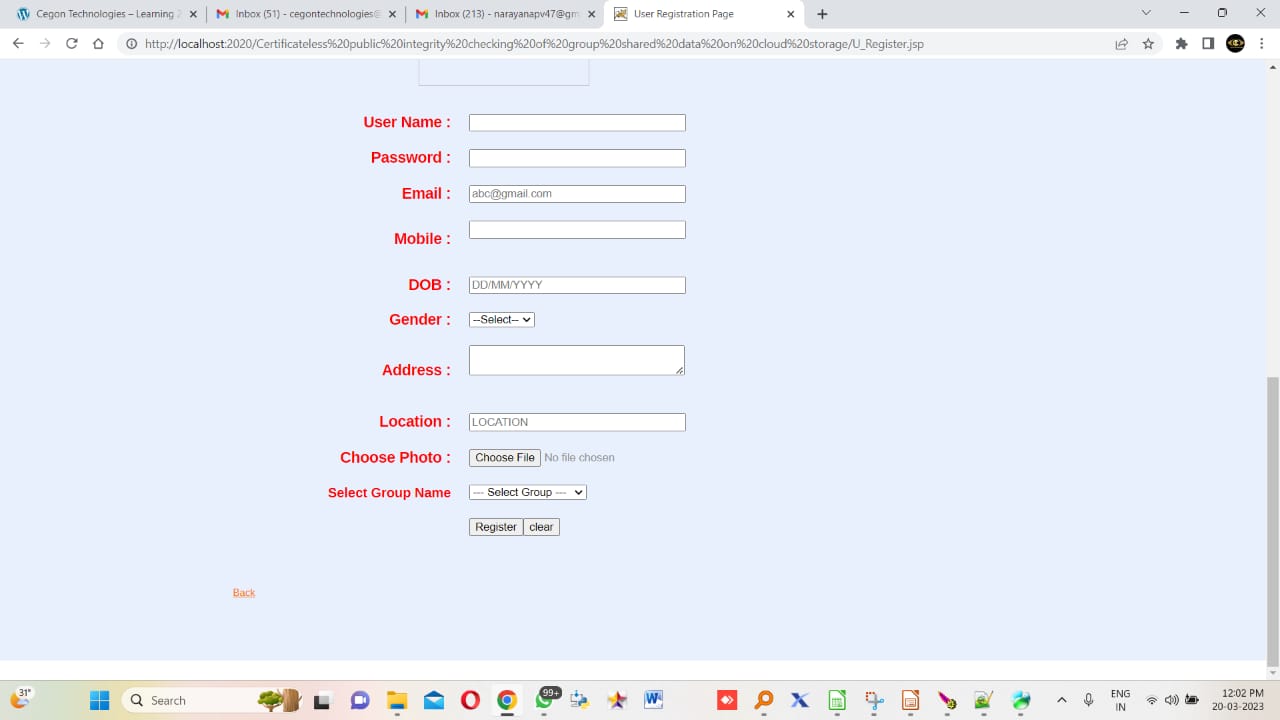
**7.**1 **SCREEN SHOTS**



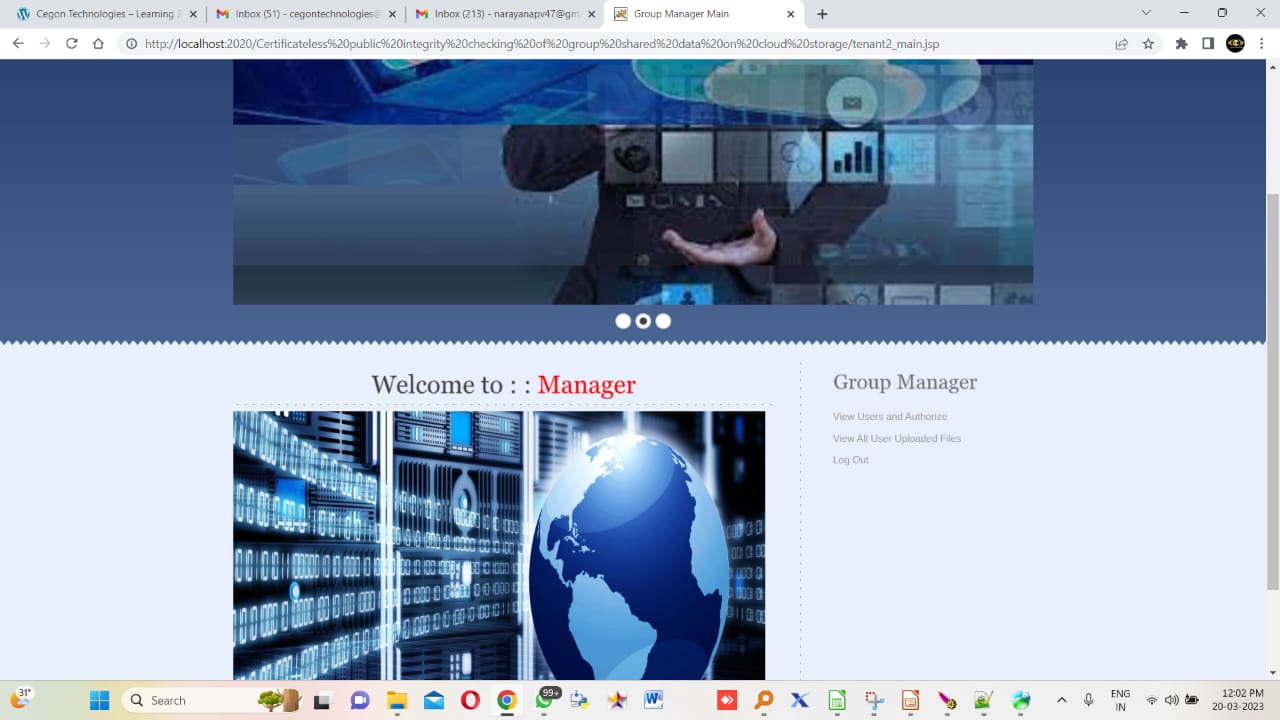
**7.2**



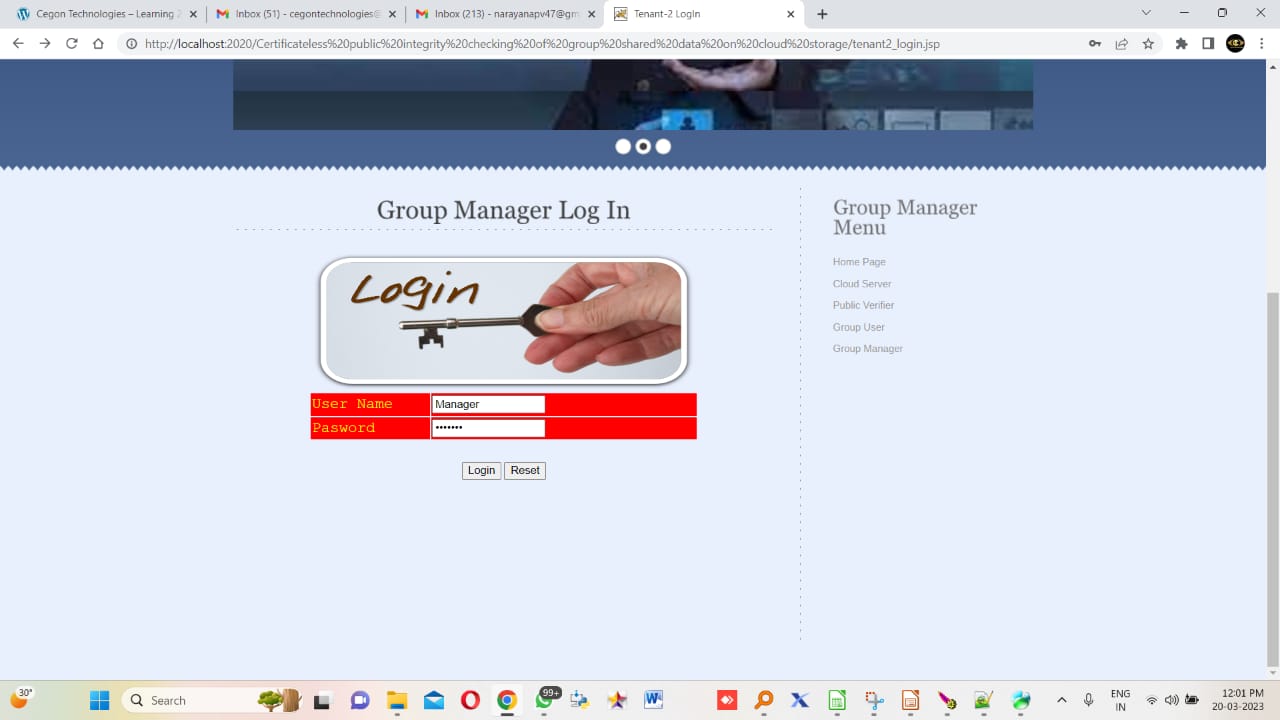
**7.3**



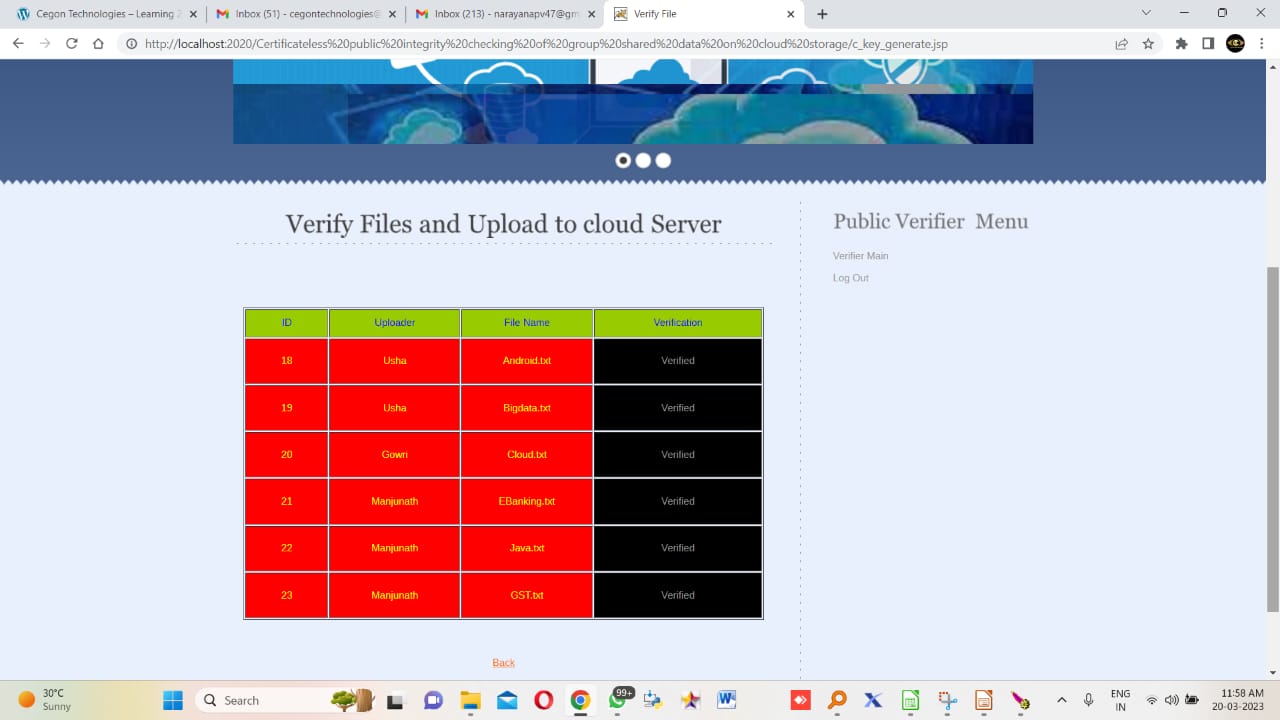
**7.4**



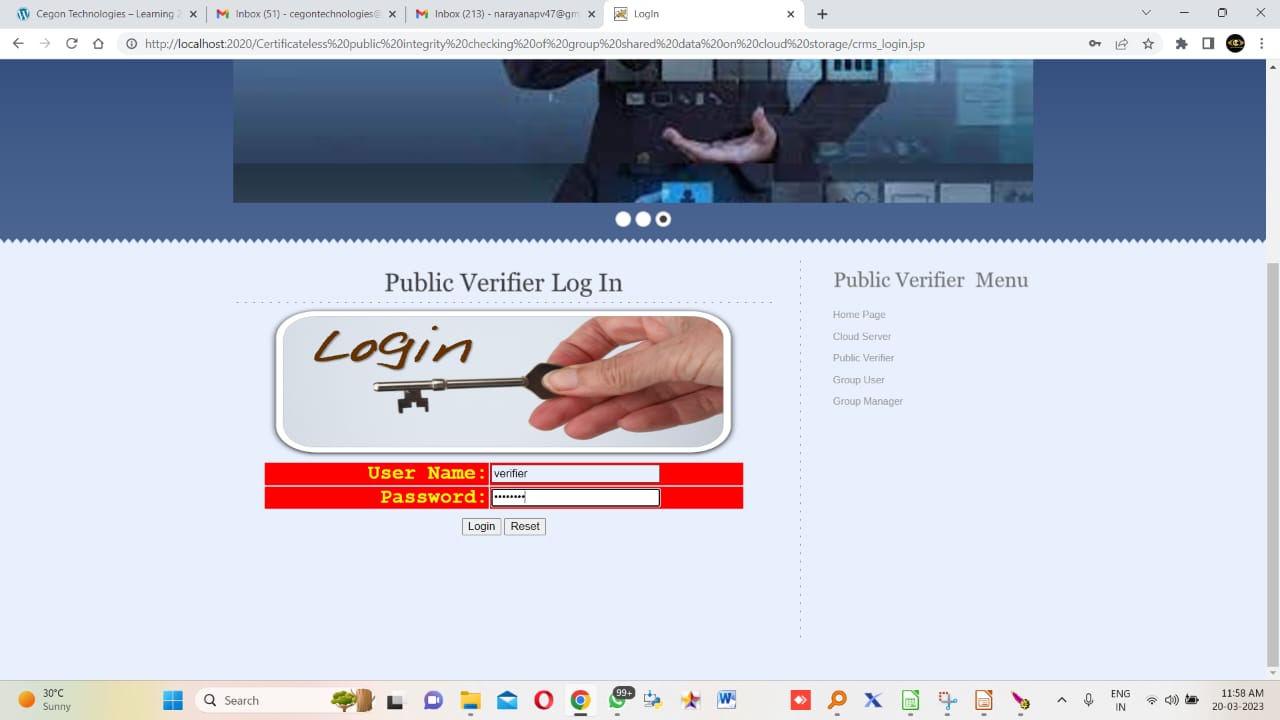
**7.5**



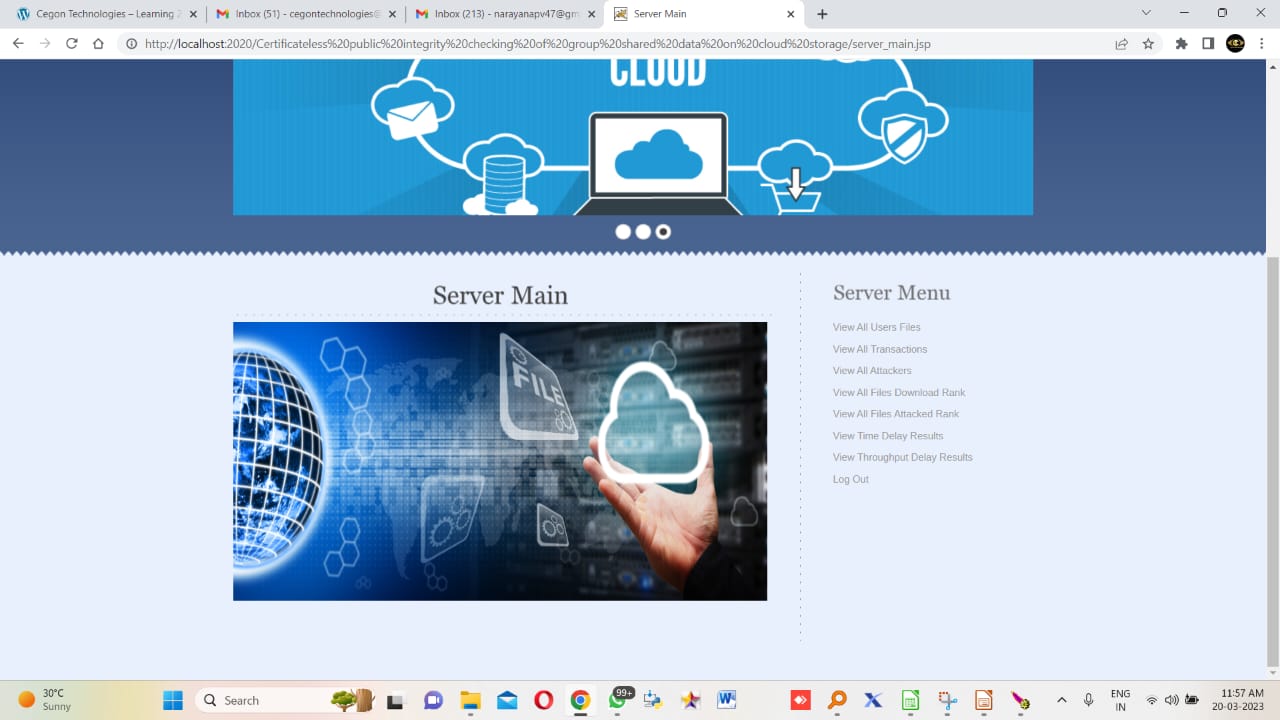
**7.6**



**7.7**



**7.8**



**7.9**



**Chapter - 8**

**CONCLUSION AND FUTURE ENHANCEMENT**

**8.1. CONCLUSION**

In this paper, we present a novel RDPC scheme for data outsourced on cloud server. Our scheme devotes to solve the integrity checking for the group data which is shared among many clients of a team. We utilize the idea of certificate less signature to generate all the block tags. Because each user of a group has both partial key and secret value, the problem of key escrow is eliminated in our scheme and the certificate management in PKI does not exist. Besides, our scheme supports public verification, efficient user revocation multiuser data modification detailed description of the system model and security model of our scheme. At last, based on the CDH and DL assumption, we prove the security of our scheme. The experiment results show that our scheme has good efficiency

.

**BIOGRAPHY**

**REFERENCES**

[1] Dropbox for Business. [Online]. Available: https://www.dropbox.com/ business, accessed Sep. 16, 2016. [2] TortoiseSVN. [Online Sep. 1, 16, 2016. [3] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic,“Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering 25, no. 6, pp. 599 – 616, 2009. [4] Y. Deswarte, J. J. Quisquater, and A. Saïdane, “Remote integrity checking,” in Proc. 6th Working Conf. Integr. Internal Control (IICIS’03), pp. 1-11. [5] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, ‘‘Provable Data Possession at Untrusted Stores,’’ in Proc. 14th ACM Conf. on Comput. and Comm 598-609. [6] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, ‘‘Scalable and Efficient Provable Data Possession,’’ in Pro and Privacy in Commun. Netw. (SecureComm’08), pp. 1-10. [7] F. Sebé, J. Domingo-Ferrer, A. Martinez-balleste, Y. Deswarte, and J. Quisquater, “Efficient Remote Data Possession Checking in Critical Information Infrastruc vol. 20, no. 8, pp. 1034-1038, Aug. 2008. [8] C. Erway, A. Küpçü, C. Papamanthou, and R. Tamassia, “Dynamic Provable Data Possession,’’ in Proc. 16th ACM C Commun. Security (CCS’09), pp. 213-222. [9] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, ‘‘Enabling Public Auditability and Data Computing,’’ IEEE Trans. Parallel Distrib. Syst., vol. 22, no. 5, pp. 847-859, May, 2011. [10] C. Wang, S. S. M. Chow, Q. Wang, preserving public auditing for secure cloud storage,” IEEE Trans. Comput., vol. 62, no. 2, pp. 362–375, Feb. 2013. [11] L. Chen, S. Zhou, X. Huang and L. X possession checking in cloud storage, ’’ Comput. Electr. Eng., vol. 39, no. 7, pp. 2413-2424, 2013. [12] Y. Yu, Y. Zhang, J. Ni, M.H. Au, L. Chen, and possession checking with enhanced security for cloud,’’ Future Gener. Comp. Syst., no. 52, pp. 77-85, 2015. [13] K. Yang and X. Jia, ‘‘An efficient and secure dynamic a for data storage in cloud computing,’’ IEEE Trans. Parallel Distrib. Syst., vol. 24, no. 9, pp. 1717-1726, 2013. [14] H. Yan, J. Li, J. Han and Y. Zhang, ‘‘A Novel Efficient Re Possession Checking Protocol in Cloud Storage, ’’ IEEE Trans. Inf. Foren. and Sec., vol. 12, no. 1, pp. 78-88, 2017. [15] Y. Feng, Y. Mu, Checking Scheme with User Privacy,’’ in Proc 20th Australasian Conf. on Inf. Security and Privacy (ACISP’15), pp. 377-394. [16] Y. Zhu, H. Hu, G. J. Ahn and M. Yu, ‘‘Cooperative provable data possession for integrity verification Parallel Distrib. Syst., vol. 23, no. 12, pp. 2231-2244, 2012. [17] H. Wang, ‘‘Identity-based distributed provable data possession in Multicloud storage,’’ IEEE Trans. Service Comput., vol. 8, no 328-340, 2015. [18] H. Wang, D. He, J. Yu and Z. Wang, “Incentive and Unconditionally Anonymous Identity-Based Public Provab Transactions on Services Computing, vol. PP, no. 99, pp. 1-1. doi: 10.1109/TSC.2016.2633260 [19] H. Wang, D. He, S. Tang, ‘‘Identity-Based Proxy-Orie Uploading and Remote Data Integrity Checking in Public Cloud,’’ IEEE Trans. Inf. Foren. and Sec., vol. 11, no .6, pp.1165-1176, 2016. [20] B. Wang, B. Li, H. Li, and F. Li, ‘‘Certificateless public auditing for data in the cloud, ’’ In Proc. IEEE Conf. on Communications and Network Security (CNS’13), pp. 136-144. [21] H. Wang and J. Li, ‘‘Private certificate in public clouds,’’ in Proc. 21th Int'l Computing and Combinatorics. (COCOON’15), LNCS 9198, pp. 575–586. [22] B. Wang, B. Li, and H. Li, “Knox: Privacy-Preserving Auditing Shared Data with Large Groups in the Cloud,” in Proc. 10th Int’l Conf. Applied Cryptography and Network Security (ACNS’ 12), pp. 507-525. [23] B. Wang, H. Li, and Shared Cloud Data Supporting Group Dynamics,” in Proc. IEEE Int’l Conf. Comm. (ICC’ 13), pp. 1946-1950. [24] X. Liu, Y. Zhang, B. Wang, an Data Sharing for Dynamic Groups in the Cloud,” IEEE Trans. Parallel and Distributed Systems, vol. 24, no. 6, pp. 1182-1191, June 2013. [25] B. Wang, B. Li, and H. Li, ‘‘Oruta: Privacy-preserving public auditing for shared data in pp. 43–56, 2014.

**Sites Reffered**

[https://www.**computer**.org](https://www.computer.org)

[www.chennaisunday.com](http://www.chennaisunday.com)

[www.jpinfotech.org](http://www.jpinfotech.org)

[www.ijariie.com](http://www.ijariie.com)

[www.ijcaonline.org](http://www.ijcaonline.org)

[www.ijetcse.com](http://www.ijetcse.com)

[www.academia.edu](http://www.academia.edu)

<https://edupediapublications.org>