A Modified Trusted Cloud Computing Architecture based on Third Party Auditor (TPA) Private Key Mechanism

R.Ushadevi
Research Scholar,
Department of Computer Applications,
St. Peter's University
Chennai Tamilnadu, India

V. Rajamani
Department of Electronics and Communication Engg
Indra Ganesan College of Engg.,
Manikandam,
Tiruchirappalli, Tamilnadu, India

ABSTRACT

A modified security mechanism for cloud computing environment based on private key mechanism is presented in this paper. Typically, users will know neither the exact location of their data nor the other sources of the data collectively stored with theirs. The data can find in a cloud ranges from public sources which have minimal security concerns to private data containing which has highly sensitive information (such as social security numbers, medical records, or shipping manifests for hazardous material). It provides trust-overlay network over multiple data centers to implement a reputation system for establishing trust between service providers and data owners. To overcome these problems, data coloring and watermarking techniques are used to protect shared data objects and massively distributed software modules. These techniques safeguard multi-way authentications, enable single sign-on in the cloud, and tighten access control for sensitive data in both public and private clouds.

Keywords*:* cloud computing, watermarking, Data coloring, social security, multi-way authentications

1. INTRODUCTION

Today, we have an ability to utilize scalable, distributed computing environments within the confines of the Internet, a practice known as cloud computing. In this new world of computing, users are universally required to accept the underlying premise of trust. Within the cloud computing world, the virtual environment lets users' access computing power that exceeds that contained within their own physical worlds. Cloud computing is the process of providing computer facilities via internet. And it's provided us better and efficient way to access information in timely manner and also increases storage of capacity for user in [1].

Cloud computing enables a new business model that supports on-demand, pay for-use, and economies-of-scale ***IT services*** over the Internet. The Internet cloud works as a service factory built around virtualized data centers. Cloud platforms are dynamically built through virtualization with provisioned hardware, software, networks, and datasets. The idea is to migrate desktop computing to a service-oriented platform using virtual server clusters at data centers. However, a lack of trust between cloud users and providers has hindered the universal acceptance of clouds as outsourced computing services [2]. To promote multitenancy, we must design the cloud ecosystem to be secure, rust worthy, and dependable. In reality, trust is a social problem, not a purely technical issue. However, we believe that technology can enhance trust, justice, reputation, credibility, and assurance in Internet applications. To increase the adoption of Web and cloud services, ***cloud service providers*** **(CSPs)** must first establish trust and security to alleviate the worries of a large number of users [1].

Cloud users are anxious about whether data-center owners will misuse the system by arbitrarily using private datasets or releasing sensitive data to a ***third party*** Without permission. Cloud security is deployed to provide full of protection between data owner and service provider. To address these issues, we propose a reputation-based trust-management scheme augmented with data coloring and software watermarking [3].

The Cloud Security Alliance 5 has identified a few critical issues for trusted cloud computing, and several recent works discuss general issues on cloud security and privacy. Public and private clouds demand different levels of security enforcement. We can distinguish among different ***service-level agreements*** (SLAs) by their variable degree of shared responsibility between cloud providers and users. Critical security issues include data integrity, user confidentiality, and trust among providers, individual users, and user groups. The three most popular cloud service models have varying security demands; the ***infrastructure-as-a-service*** (IaaS) model sits at the innermost implementation layer, which is extended to form the ***platform-as-a-service*** (PaaS) layer by adding OS and middleware support. PaaS further extends to the ***software as-a-service*** (SaaS) model by creating applications on data, content, and metadata using special **APIs**. This implies that SaaS demands all protection functions at all levels. At the other extreme, IaaS demands protection mainly at the networking, trusted computing, and compute/storage levels, whereas PaaS embodies the IaaS support plus additional protection at the resource-management level.

1.1 Virtual Trust Difficulty in Cloud Services :

The cloud security grouping has identified a few vital issues for trusted cloud computing, and several recent works discuss general issues on cloud security and privacy. Public and private clouds order different levels of security enforcement. We can differentiate among different ***service-level agreements*** **(SLAs)** by their variable degree of shared responsibility of Security concerns which contains data integrity, user confidentiality, and trust among providers, individual users, and user groups. The three security demands should be varied from the three cloud service models that are described in below the infrastructure-as-a-service model sits at the innermost implementation layer, which is expanded to form the platform-as-a service (PaaS) layer by adding OS and middleware support. Pass further extends to the ***software-as-a-service*** **(SaaS)** model by creating applications on data, content, and meta-data using special **A**

PIs. This implies that SaaS demands all protection functions at all levels [4].

1.2 Securing Transportation as a service:

The Iaas model is works to compute networking and data storage, other resources in a virtualized environment. Amazon's Elastic Compute cloud is one of good example of **Iaas**, at the cloud infrastructure level, **CSP** can implement network security with intrusion-detection systems, firewalls, antivirus programs, distributed denial-of-service suspicion and so on.

1.3 Securing policy as a Service:

Cloud platforms built in Iaas with system integration and virtualization middle-ware. And these platforms can be used to users for implementing user-built software applications onto the cloud infrastructure using provider-supported programming languages and software tools like **Java, Python and Dot net** [5].

1.4 Cloud Suppliers & Reported Services

Hardware and software features are presents in cloud security. Three main security requirements are used in cloud computing demands such as confidentiality, integrity and availability. Report is the process of maintaining user communicated details on database. And these details are viewed only by an authenticated user in cloud environments.

1.5 Data reliability and confidentiality Protection:

Data integrity is the ability of cloud provider keep data safe from **unauthorized person or hackers**. Confidentiality is essentially the way the cloud provider insures that the data is secured from unauthorized access. The measures that the cloud provider uses to insure that this goal is met includes; physical isolation and cryptology. Cloud computing is a **public network,** which brings a set of complicated challenges for the provider to produce isolation for the customer. Physical isolation is accomplished by using virtual local area networks and middle boxes. The second method the provider uses is cryptology, which essentially encrypts the data before it is placed into the cloud. These two methods are standard measures that are used to secure data in the cloud [6].

Many tools are available for constructing cloud applications on large datasets and it's provided by cloud software environment to desired users. Let's following features are presents in security and privacy such as:

- Fine-grained access control to conserve data integrity and deter intruders or hackers
- A method to stop **ISPs** or **CSPs** from attacking user privacy.
- CSPs that struggle against spyware and web bug

We can improve some of these features with cloud reputation systems and more efficient identity management systems [6]

Some features are,

1. Cloud resource can access security protocol like **http** and **secure socket layer**.
2. Fine grained access control to protect data integrity and data attacker.

2. RELATED WORKS

2.1 Trusted Cloud Computing over Data Centers:

Security aware cloud architecture and this used to identify the protection mechanisms needed. Intruder detection action is should be implemented by using these architecture [7].

2.2 Cloud protection Infrastructure:

Trusted and dependable cloud architecture is shown in **Figure 1**. This architecture helps protect network attack by launching trusted operational sectors for different cloud applications. The difficulties in security agreement that **CSPs** protect all data center servers and storage areas [8]. Our architecture protects **VM** checking from software based attacks and upholder data and information from robbery, fraud and natural failures. It provides strong authentication and authorized access to sensitive data and on-demand services. We had several design objectives for a trusted and dependable cloud when creating our architecture.

2.2.1 Measuring the Existing System with the Cloud:

In the Existing System we used the concept of only the data coloring process, which to secure the data in the safe manner, which to be assign a public key for the cloud [7]. But comparing with the existing system we used the concept of the water marking process for secure the data in the cloud servers from the attackers and unauthenticated persons which to be assigned each public key and private key to be sage the data or images in the server [5].

In this we used to images to save in cloud servers according to efficient space in cloud. The data must be **encrypt and decrypt** in client and the data must be send to cloud server, so whenever the user want to retrieve the information from the server [9].

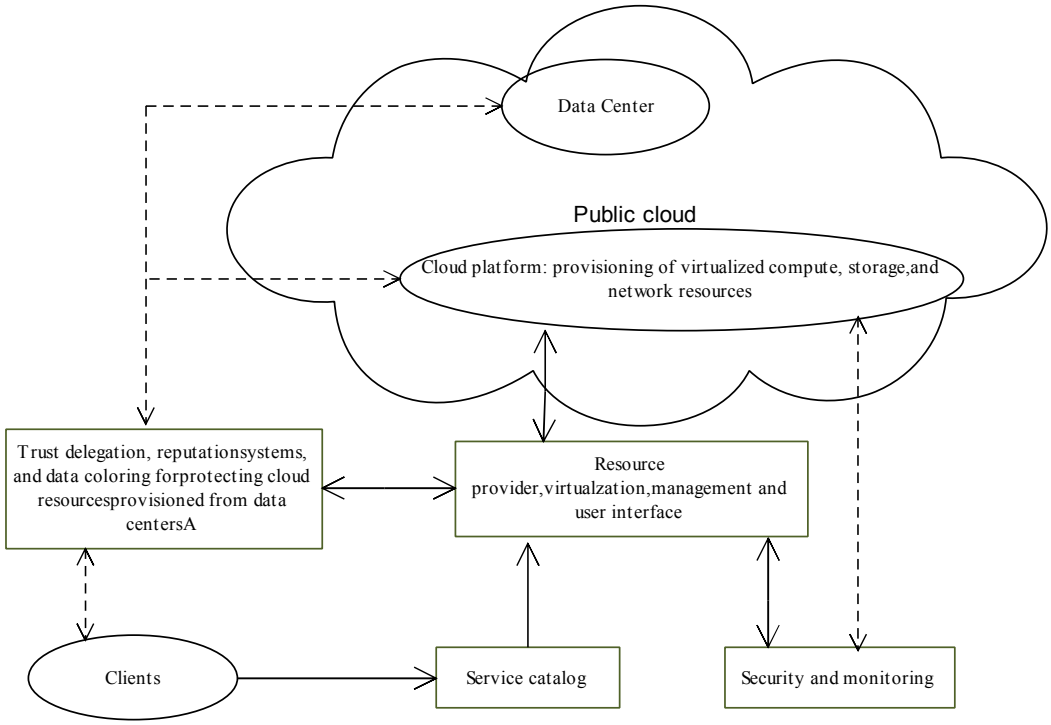

Figure1.A trusted and dependable cloud architecture.

2.3 Virtual network security and trust negotiation

Virtual network security defends **VMs** in virtualized data centers and avoids data loss for other occupants .user must use annoyed documentation to assign trust across public key infrastructure spheres for data centers. Trust negotiation among different ***certificate authorities*** (CAs) determines policy clashes [5].

2.4 Worm containment and DDoS protection.

Internet worm containment and distributed security against **DDoS** hits are necessary to protect infrastructure from malware, Trojans and cyber criminals. This loads that we secure combines identities in public clouds [8].

2.5 Reputation systems for data centers:

We can make reputation systems using **peer-to-peer** technology or a hierarchy of reputation systems among virtualized data centers and distributed file systems. In such systems, we can defend logical copyright using practical content poisoning to avoid privacy [10].

2.6 Data embedded with coloring:

Our architecture uses data coloring at the software file or data object level. This lets us separate user access and protect susceptible information from provider access as **Figure 2** shows. Data coloring is the process of changing original input of **RGB** color image file into a gray scale image file [7].

2.7 Security of Virtualized Resources

Virtualization improves cloud security. First, **VMs** add a supplementary layer of software that could become a single point of failure. That is , virtualization is the process of dividing a single physical machine into multiple **VMs**(as with server consolidation), giving each **VM** better security separation and defending each partition from **DDoS** hits by other partitions. Security hits in one **VM** are separated and contained **VM** failures don't transmit to other **VMs**. A hypervisor provides the same visibility as the guest OS but with complete guest separation. This fault restraint and failure separation **VMs** provide allows for a more secure and robust environment [3].

This implies that we can save, clone, encrypt, move or restore the **VM** with ease. **VMs** also enable higher availability and faster disaster recovery [3].

2.8 Live Passage and Open Virtual Format.

Format live passage occurs when we move a **VM** from one device to another with least down time. By using live passage of VMs for securing cloud platforms to improve from failures or tragedies. We suggest live migration of **VMs** specifically designed for building distributed **IDSs**.**CPSs** can deploy multiple **IDS** VMs at various resource sites, including data centers. **DIDS** design demands trust negation among **PKI** domains. Providers must resolve security policy conflicts at design time and update them periodically. A defense scheme is needed to protect user data from server attacks. Additionally, user's private data must not be leaked to their users without authorization in [11].

Once users travel data into the cloud, they can't simply remove their data and programs from one cloud server to run on another. This directs to a data lock-in problem. One possible solution is to use standardized cloud **APIs** [2]. This involves building standardized virtual platforms that adhere to the open virtual Format. User can transmit data from one application to another with minimum loss of data [11].

3. PROPOSED WORKS

3.1 Reputation-Guided Data-Center Protection

P2P social networking or online shopping services are designed in reputation systems. We can control such systems to defend cloud platform resources or user applications on the cloud [12]. An implementation of centralized reputation system is simple but demands more powerful and reliable server resources. Distributed reputation systems are more scalable and reliable for handling failures. The reputation system we propose can help providers build content aware conviction zones using the **RSA DLP** package for data traversing monitoring. Reputation represents a combined estimate by users and resource owners. Researchers have proposed many reputation systems in the past for **P2P**. ***Multi-agents or e-commerce*** systems.

We can structure the spread over the surface with a scattered hash table to achieve fast aggregation of global reputations from frequent local reputation scores. And two overlay of layer should be established by using this structure.

At the bottom layer is the trust place on top for distributed trust conciliation and reputation aggregations over various resource sites.

3.1.1 Security Level in Proposed System Compare to Existing System:

In the Existing system we used coloring model, which the data could not be safe and the data could be lost when we retrieve from the cloud server, and with the help of only ***data coloring*** the data must be shared in the server side, and there must be having a chance to hack the data, and there should not have ***third party authority to check the key*** to retrieve the data from the **client side**. But in the Proposed system we used, both data coloring and **water marking** process, in order to store the data or image in the cloud server by assigning the public key, and this key and watermarking and data coloring images are send to third party and third party have full authority to check the key and send it to the server, and there Third Party Auditor must have public key whenever the data must be retrieve. In the watermarking process, the security level is high so the data or images cannot be identified by the attackers in the cloud.

3.2 Third Party Auditor:

The third party auditor works to generate **primary key** or **public key** to authenticated user and checking user to whether ***authorized or unauthorized*** person during user process. Suppose if unauthorized user enter into a system and it terminate that unauthorized entry into a system. This third party auditor can be used to checking that original user with respect to public key. And this process should be performed during ***encryption process as well as decryption process.***

Third Party Auditor: an entity, which has knowledge and capabilities that clients do not have, is trusted to assess and represents risk of cloud storage services on behalf of the clients upon request. In the cloud paradigm, by putting the large data files on the remote servers, the clients can be relieved of the load of storage and computation. As clients no longer acquire their data locally, it is of critical importance for the clients to ensure that their data are being correctly stored and maintained. That is, clients should be equipped with certain security means so that they can periodically verify the correctness of the remote data even without the existence of local copies. In case those clients do not necessarily have the time, feasibility or resources to monitor their time and data, they can delegate the monitoring task to a trusted **TPA**. In this paper, we only consider verification schemes with public audit ability: any **TPA** in control of the public key can act as a verifier. We assume that **TPA** is unbiased while the server is entrusted. For application purposes, the clients may interact with the cloud servers via **CSP** to access or retrieve their presorted data.

3.3 Data Coloring and Software Watermarking using TPA Key

Given cloud computing use of shared files and datasets, an opponent could negotiation privacy, security, and copy right in a cloud computing environment. Trusted software environment that provides useful tools for constructing cloud applications over protected datasets. Watermarking embeds a secret message into a cover message Water marking is the process of adding the user text behind of image files. And it's used to shading the text into an image files. And it's mainly used for digital patent administration. Watermarking can be used to conserves data as well as software modules. We propose second order ***fuzzy*** membership function for defending data owners. We extend this model to add unique data colors to maintain big volume of datasets in the cloud. Software watermarks come in two flavors, **static and dynamic**. Static watermarks are stored in the Application executable itself; whereas, dynamic watermarks are constructed at runtime and stored in the dynamic state of the program. While static watermarks have been around for a long time. **Figure 2 shows** the forward and backward color generation processes. Here, we add the cloud drops into the input photo and remove color to restore the original photo. The coloring process uses three data characteristics to generate the color.

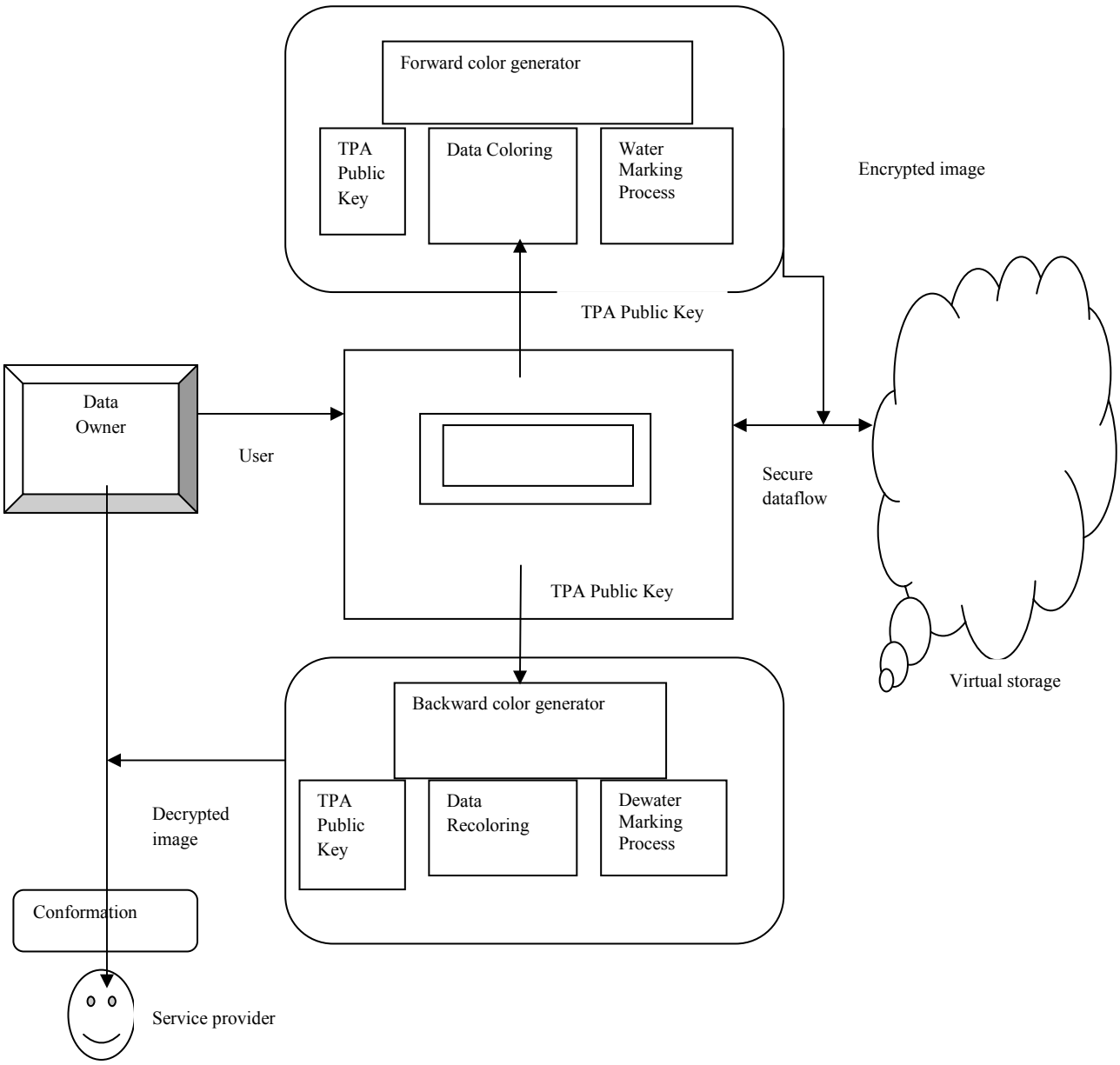

Figure 2 :Forward and backward color generation processes

The **expected value** (Ex) depends on the data content, whereas entropy and hyper entropy add randomness or uncertainty, which are independent of the data content and known only to the data owner. Collectively, these three functions generate a collection of cloud drops to form a unique **"color"** that providers or other cloud users can't detect. Additional details about this cloud watermark scheme are available.

We can use data coloring at varying security levels based on the variable cost function applied. We can apply the method to protect documents, images, video, software and relational databases. **Figure 2 shows** the details involved in the color-matching process, which aims to associate a colored data object with its owner, whose user identification is also colored with the same **Ex**, En and He identification characteristics. The color-matching process assures that colors applied to user identification match the data colors. This can initiate various trust-management events, including authentication and authorization. Virtual storage supports color generation, embedding, and extraction.

Decoloring process should be performed **under decryption process**. And it is followed by Third Party Auditor because **TPA** works to checks user as registered or not. If suppose user is registered person then that member can able continues their decoloring process which is automatically stored into their allotted regions. Generally, decoloring is the process of reconstructing colored image. And it could be processed during **decryption process**.

3.4 General Watermarking Process

A watermarking system can be viewed as a communication system consisting of three main elements: an embedded a communication channel and a detector and is **shown in Figure 3**. Watermark information is embedded into original image itself, and it is performed in encryption process for making security on original information. Embedded is similar to encryption process which is used to change content into another format with help of **secret key.**

Detector process is also similar to decryption process which is used to perform reverse process of encryption. The watermark information is embedded within the original image before the watermarked image is transmitted over the communication channel, so that the watermark can be detected at the receiving end, that is, at the detector.

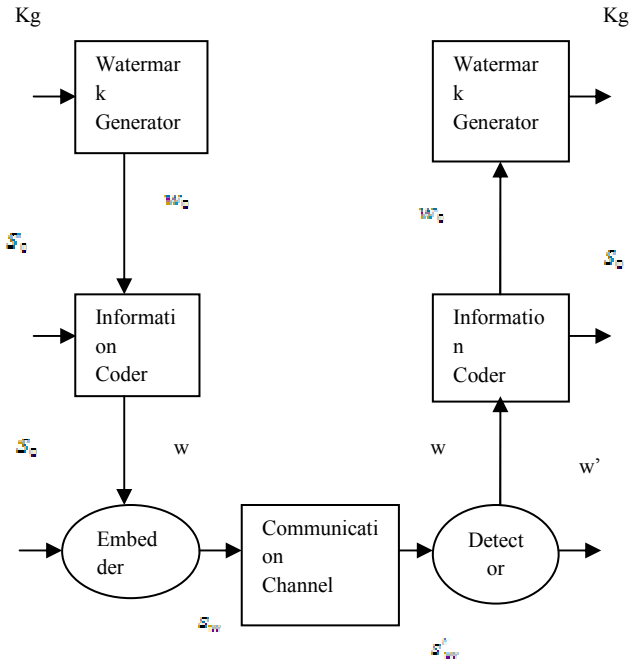

Figure 3 Basic watermarking processes in cloud computing environment

A watermark W_o is generated by the watermark generator possibly with a secret watermark generation key K_g. The watermark W_o can be a logo. Instead of directly embedding it into original image, the watermark W_o can be pre-coded to optimize the embedding process, i.e. to increase robustness against possible imperceptibility of the watermark. This is done by an information coder who may require the original information S_o. The outcome of the information coding component is denoted by symbol W that, together with the original information S_o and possibly a secret key K, are taken as input of the embedded. The secret key K is intended to differentiate between authorized users and unauthorized users at the detector in the absence of K_g. The embedded takes in W, S_o and K, so as to hide W within S_o in a most imperceptible way with the help of K, and produce the watermarked information S_w. Afterwards, S_w enters into the communication channel where a series of unknown signal processing operations and attacks may take place. The outcome of the communication channel is denoted by the symbol S'_w. At the receiving end, the detector works in an inversely similar way as the embedded, and it may require the secret key K_g, K, and the original signal S_o. Then the detector reads $S'w$ and decides if the received signal has the legal watermark.

3.5 Static Watermarking Techniques:

Static watermarks are stored in the application executable itself, static data watermarking method in which the watermark is embedded in an image using one of the many media **watermarking algorithms**. This image is then stored in the static data section of the program.

3.6 Dynamic Watermarking Techniques:

In each case, the mark is recognized by running the watermarked program with a predetermined input sequence $I=I_1, I_2 \ldots I_k$ This highly unusual input makes the application enter a state which represents the watermark [9]. There are **two algorithms** is used in **dynamic watermarking Techniques such as Embedding Algorithm and Detection Algorithm.**

Embedding Algorithm:

In this algorithm, original image is embedded into a secret message. Embedding process is a one of method to generating secret dependency information.

The watermark embedding algorithm steps are described in below

A. Generate a binary watermark w with the secret shared with detector

For each pixel i in image f

1. Calculate secret dependency information S(i) according to

$$S(I)=\sum^{n}{}_{\epsilon N(I)}(W(I)\pm W(j))(-1)^{W(j)}f_m \quad (1)$$

Where (±) denotes exclusive-or (XOR) operation.

2. Adjust $f_2(I)$ so it is hold following equation

$$PARITY(f(I)+S(I))=w(I) \quad (2)$$

Detection Algorithms:

In this algorithm, watermarked image is reconstructed during decryption process. Detection is a one of method to reconstruct watermarking image. Watermark detection algorithm steps are described in below:

1. Generate a binary watermark w with the secret key shared with the embedded.
2. For each pixel i
3. Calculate the secret dependency information S(I) from received image f' according to equation (1)
4. Extract the watermark bit w'(I) according to:

$$W'(I) = PARITY(f'(I) + S(I)) \quad (3)$$

Calculate difference D (I) between w (I)

and the extracted watermark w' (I) using:

$$D(i) = \begin{cases} 0 & w(I) = w'(I) \\ 255 & \text{otherwise} \end{cases} \quad (4)$$

Where,

f: e original gray scale image and

f(i): e gray scale of the i th pixel of f

f_M(i): the seven most significant bits of f(i)

f_l(i): the least significant bit of f(i)

f ′: The image received by the watermark detector. If tampered with, f ′ is the watermarked version of f

Z: the size of the image f

w: the secret key generated binary watermark of the s size M as the image f

w(i): the i th bit of *w*

w ′: the extracted binary watermark by the decoder

w ′ (i): the i th bit of *w ′*

D: The binary difference map between *w* and *w ′*with its i th pixel denoted as ***D*(*i*) (*D*(*i*) ∈ {0,255})** indicating whether *w*(*i*) and *w ′* (*i*) are different. Wherever the watermarked image is manipulated, noises are shown in the corresponding portion of the difference map *D*. We could also identify what type of manipulation has been done from *D*.

k: The length of dependency neighborhood.

L: the size of the neighborhood, ***L*= *k* × *k*.**

N(i): the square dependency neighborhood centered at pixel i consisting of **k**×**k** pixels including pixel i itself

S(i): The secret non-deterministic dependency information of pixel i calculated.

4. RESULTS AND DISCUSSIONS:

From the cloud computing server we can use are of sharing the files and data's or any images which is in the form of privacy security in the existing system. So it was difficult to secure the data in the cloud, from the hackers. And also in the existing system which we used the data coloring model to secure, but the security level is low. But in the proposed system we use the method watermarking to secure the data in the cloud server, which was the data must be secured in the cloud itself, to protect it from hackers. So generally, we have the **attacks** like **hackers, unauthorization**, and malware attacks etc, which it was **critical in cloud** systems.

This cloud platforms which cause many difficulties for business platforms which they loss their most of the money in this insecurity cloud platforms. So in this platforms itself, we introduce the method of watermarking and data color model to secure the data in the cloud server, which it should be in **high security.**

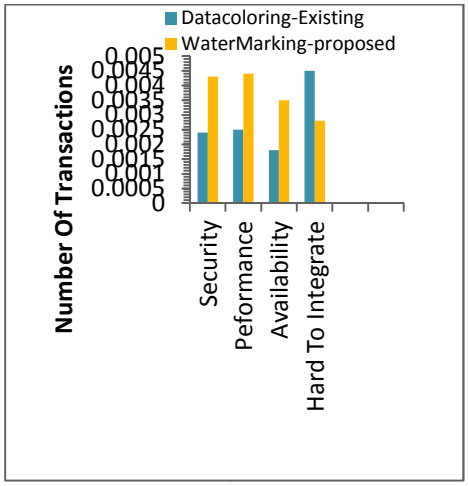

Figure 4 Security performance comparisons

In Figure 4 (0, 0.001, 0.002, 0.003, 0.004 and 0.005) represents the number of transactions which are given in the comparison scale for the existing system from the proposed system. In the proposed we shows the comparison from the **figure** which it is seen that the proposed security process from the security and performance is higher than the data coloring process from an existing system. The results which are indicate that the proposed method is extendable from the data coloring, which results in **watermarking process.** In existing method we used Data coloring process, which was not very efficient, because the data security is not safe. And also by comparing the performance of the data coloring with water marking, the performance is very less in existing method compare to the proposed system. The water marking availability is high in the cloud computing. We can use this water marking process at **different security levels** at business intelligence, and economics and also for **various market places** in cloud computing.

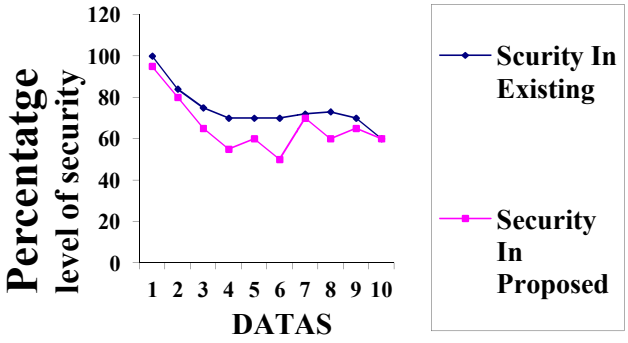

Figure 5: comparison of security level

In the **figure 5,** we compare the security level of our proposed system with the comparison of existing system. Cloud Computing has been envisioned as the next generation architecture of **IT** Enterprise. So that our proposed system gives the good result for the security in watermarking process. In contrast to traditional solutions, where the **IT services** are under proper physical, logical and personnel controls, Cloud Computing moves the application software and databases to the large data centers, where the management of the data and services may not be fully responsible. This unique attribute,

however, poses many new security challenges which have not been well understood. In this article, we focus on cloud data storage security, which has always been an important aspect of quality of service. To ensure the correctness of users' data in the cloud, we propose an effective and flexible distributed scheme with two salient features, opposing to its predecessors. By utilizing the homomorphism token with distributed verification of **erasure-coded data**, our scheme achieves the integration of storage correctness insurance and data error localization, i.e., the identification of misbehaving server in the existing method of data coloring and the security level of the proposed in the watermarking process. In cloud computing we use security for to process the data from the server, upload a file.

So that our proposed system gives the good result for the security in watermarking process. In contrast to traditional solutions, where the **IT services** are under proper physical, logical and personnel controls, Cloud Computing moves the application software and databases to the large data centers, where the management of the data and services may not be fully responsible. This unique attribute, however, poses many new security challenges which have not been well understood. In this article, we focus on cloud data storage security, which has always been an important aspect of quality of service. To ensure the correctness of users' data in the cloud, we propose an effective and flexible distributed scheme with two salient features, opposing to its predecessors. By utilizing the homomorphism token with distributed verification of **erasure-coded data**, our scheme achieves the integration of storage correctness insurance and data error localization, i.e., the identification of misbehaving server in the existing method of data coloring and the security level of the proposed in the watermarking process. In cloud computing we use security for to process the data from the server, upload a file.

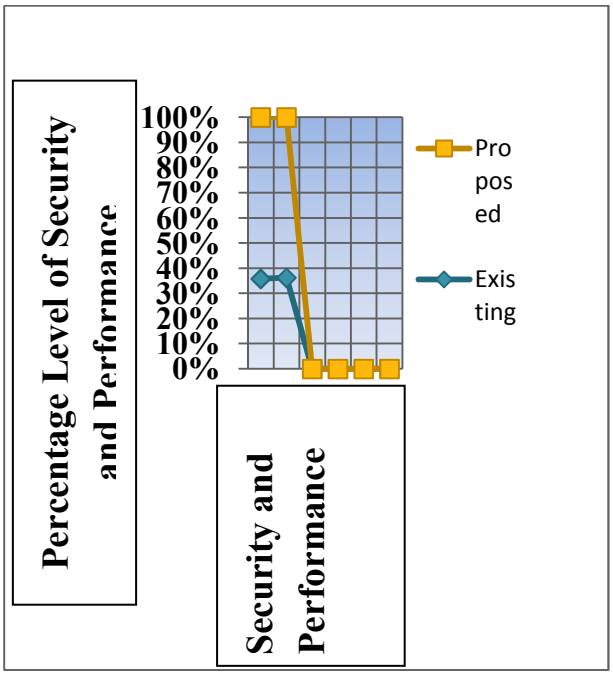

Figure 6: security and performance with percentage

In **figure 6** we show the security and performance of the watermarking from the data coloring, in our scheme, servers are required to operate on specified rows in each correctness verification for the calculation of requested indication from the user. We will show that this ***"sampling"*** strategy on selected rows instead of all can greatly reduce the computational overhead on the server, while maintaining the detection of the data corruption with high probability. Whenever data corruption has been detected during the storage correctness verification across the distributed servers, we can almost guarantee the simultaneous identification of the misbehaving server(s). Through detailed security and performance analysis, we show that our scheme is highly efficient and prevent the data from the attackers, malicious data modification attack, and even server colluding attacks. In the above **figure** we measure only the security and performance of the existing method with the comparison of the proposed methods. In the proposed method the data must be secured using water marking method in the cloud computing, to view the data we use this graph page to user.

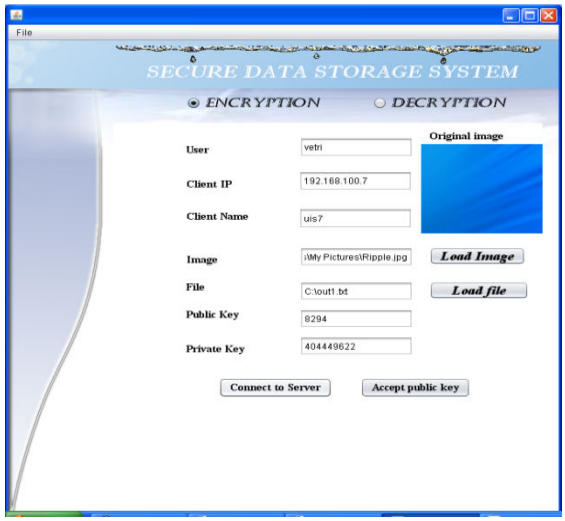

Figure 7 Original Image of the Cloud Computing Without Watermarking

In **figure 7,** we use the original image for the water marking process in the cloud computing. In this figure along with the user name and client **IP address** along with the given image for data hiding, so that in this client form using encryption we use to store the images along with the public key and private key. In cloud computing we use the two methods as **Encryption and Decryption method** to secure the data using private key mechanism. So when the user want to hide and retrieve the data using water marking method, first the data must be **encrypted** with its belonging **IP address** and the elevated image. So when the IP address and the image are saved in the server. The server will be given the private key to hide the data with the image which was given by the user. So this **encryption process** must be safe to store or to hide the data using water marking process of the original image to hide the data in the cloud computing. So that the storage capability is high when we use this method and it will be secured in the Data Storage System. Because cloud computing is built on top of virtualization, if there are security issues with virtualization, then there will also security issues with cloud computing.

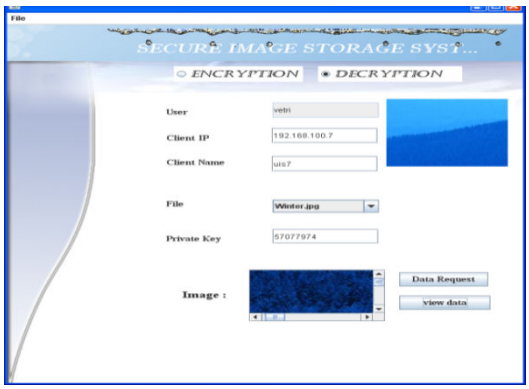

Figure 8. Original Image of the Cloud Computing With Watermarking.

In **figure 8**, after the encryption process, we use the decryption method to secure the data and store in the client and the data must be sending to the server. The servers will response to the client, if the user is authenticated or not. So that whenever the user must get the original data, the user must be authenticated using the key word. In this we had taken the original image to encrypt the image and declare the public key to secure the data in the safe manner in the cloud server. So after encrypting the same image as to be decrypt the image and save the image in the cloud server which to be given the public key. So prevent the data from the attacker in the cloud. So with this process, Cloud watermarking utilizes the robustness of human discrimination, and transforms user logo into a series of random realization by means of normal distribution. The process is uncertain and irreversible. ***Combining the mature embedding algorithm***, it can protect user copyright more efficiently in cloud computing.

4. CONCLUSION

Data coloring and software watermarking techniques protect shared data objects and **massively distributed software modules**. These techniques safeguard multi-way authentications, enable **single sign-on** in the cloud, and tighten access control for sensitive data in both **public and private** clouds. Water marking is the process of adding the user text behind of image files. And it's used to shading the text into an image files. And it's mainly used for digital patent administration these techniques can be used to protecting the data from unauthorized accesses. With the help of water marking and data coloring process, we can use this implementation in real time process, to hide the data in the safe manner in cloud computing. So that we can compute and hide the data in the safe side, so when retrieving the data we must have security only when the client can use to access the data in the ***cloud server***.

5. REFERENCES

[1] Hui Zheng "Cloud computing and its benefits" IT 103-005.

[2] L. Xiong and L. Liu, "Peer Trust: Supporting Reputation-Based Trust for Peer-to-Peer Electronic communities," IEEE Trans. Knowledge and Data Eng., July 2004, pp. 843–857.

[3] K. Hang, S. Kulkarni, and Y. Hu, "Cloud Security with Virtualized Defense and Reputation-Based Trust Management" IEEE Int'l Conf. Dependable, Autonomic and Secure Computing, IEEECS Press, 2009.

[4] Kyle Whitney "Security in Cloud" ITEC 620 Apr 8 ,2012.

[5] "Security Guidance for critical areas of focus in cloud computing," Cloud Security Alliance, apr. 2009; www.cloudsecurityalliance.org/guidance/csaguide.v.2.1.pdf.

[6] J. Rittinghouse and J. Ransome, Cloud Computing: Implementation, Management and Security, *CRC* Publisher, 2010.

[7] S. Song et al.,"Trusted P2P Tracsactions with Fuzzy Reputation Aggregation," IEEE internet computing, vol 9, no.6, 2005, pp. 24 -34.

[8] J. Nick, "Journey to the Private Cloud: Security and Compliance," tech. presentation, EMC, Tsinghua Univ., 25 May 2010.

[9] C. Collberg and C. Thomborson, "Watermarking, Tamper-Proofing, and Obfuscation-Tools for Software Protection," IEEE Trans. Software Eng., vol. 28, 2002, pp.

[10] R. Zhou, R. Zhou, and K. Hwang, "Power Trust: A Robust and Scalable Reputation System for Trusted Peer-to-Peer Computing," IEEE Trans. Parallel and Distributed Systems, Apr. 2007, pp. 460–473.

[11] C. Clark et al., "Live Migration of Virtual Machines," Proc. Symp. Networked Systems Design and Implementation, 2005, pp. 273–286.

[12] X.Lou and K. Hwang, "Collusive Piracy Prevention in P2P Content Delivery Networks," IEEE Trans. Computers, July 2009, pp. 970–983.

[13] T. Mather, S. Kumaraswamy, and S. Latif, Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, O'Reilly Media, 2009.

[14] D.Li, C. Liu, and W.Gan, "A New cognitive Model: Cloud Model," Int'l J. Intelligent Systems, Mar.2009, pp.357 - 375.

[15] D. Li and Y. Du, Artificial Intelligence with Uncertainty, Chapman & Hall, 2008.

[16] Hwang, G. Fox, and J. Dongarra, Distributed systems and cloud computing: Clusters, Grids/P2P, and Internet Clouds, Morgan Kanfmann, to appear, 2012.