ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 12, December 2013

Enabling Efficiency in Data Dynamics for Storage Security in Cloud Computing

K.Gayathri[1], P.Umamaheswari[2], P.Senthilkumar[3]

Assistant Professor, Anna University, Tamilnadu, India[1,2]

Assistant Professor, Adama Science and Technology University, Ethiopia, East Africa[3]

Abstract: Cloud Computing has been envisioned as the next-generation architecture of IT Enterprise. It moves the application software and databases to the centralized large data centers, where the management of the data and services may not be fully trustworthy. This unique paradigm brings about many new security challenges, which have not been well understood. This work studies the problem of ensuring the integrity of data storage in Cloud Computing. The introduction of TPA eliminates the involvement of the client through the auditing of whether his data stored in the cloud is indeed intact, which can be important in achieving economies of scale for Cloud Computing. The support for data dynamics via the most general forms of data operation, such as block modification, insertion and deletion, is also a significant step toward practicality, since services in Cloud Computing are not limited to archive or backup data only. We first identify the difficulties and potential security problems of direct extensions with fully dynamic data updates from prior works and then show how to construct an elegant verification scheme for the seamless integration of these two salient features in our protocol design. In particular, to achieve efficient data dynamics, we improve the existing proof of storage models by manipulating the classic Merkle Hash Tree construction for block tag authentication.

Keywords: Merkle Hash Tree, public auditability, homomorphic token with distributed verification, bilinear aggregate signature, localization of data error, Third Party Auditor(TPA), block tag authentication.

I. INTRODUCTION

Cloud Computing share distributed resources via network in the open environment thus it makes security problems. All types of users who require the secure transmission or storage of data in any kind of media or network. Since the data transmission on the internet or over any networks are vulnerable to the hackers attack [1],[7]. However, the system imposes a priori bound on the number of queries and does not support fully dynamic data operations, *i.e.*, it only allows very basic block operations with limited functionality, and block insertions cannot be supported. We consider dynamic data storage in a distributed scenario, and the proposed challenge response protocol can both determine the data correctness and locate possible errors. Here, we only consider partial support for dynamic data operation. Juels describe a "proof of retrievability" (PoR) model, where spot-checking and error-correcting codes are used to ensure both "posses- sion" and "retrievability" of data files on archive service systems.[4],[8]. However, the number of queries a client can perform is also a fixed priori, and the introductionof pre-computed "sentinels" prevents the development of realizing dynamic data updates.[1] Although the existing schemes aim at providing integrity verification for different data storage systems, the problem of supporting both public auditability and data dynamics has not been fully addressed. How to achieve a secure and efficient design to seamlessly integrate these two important components for data storage service remains an open challenging task in Cloud Computing. Portions of the work presented in this have previously appeared as an extended abstract. We revise the article a lot and add more technical details as compared to the previous model.

II.MODULE DESCRIPTION

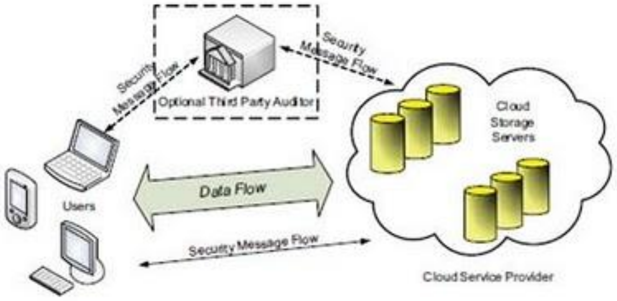


Fig.1. Architecture Diagram

Client:

An entity, which has large data files to be stored in the cloud and relies on the cloud for data maintenance and computation, can be either individual consumers or organizations. This modules is used by the client for uploading the data file or data string to the cloud server.[3]

Cloud Storage Server (CSS):

An entity, which is managed by Cloud Service Provider (CSP), has significant storage space and computation resource to maintain the clients data. It is mostly used to store the data which we need to store important information and also we can able to retrieve all the details or a particular details and also we can able to update those details.[4]

Third Party Auditor:

An entity, which has expertise and capabilities that clients do not have, is trusted to assess and expose risk of cloud storage services on behalf of the clients upon request. The third party auditor is used to verify the files which the other parties are sent to the third party auditor.

Mobile Alert:

An entity, which produces an alert to the cloud storage service provide or administrator who manages the cloud storage server.[3] For the safety and security purpose the customer need to login with their username and password after logged on the soft program the random number will be sent to the related mobile number which is based on the customers username and password. After enter the random number only can able to enter the soft program.

III. CHARACTERISTICS

Cloud computing is cost-effective. Here, cost is greatly reduced as initial expense and recurring expenses are much lower than traditional computing. Maintenance cost is reduced as a third party maintains everything from running the cloud to storing data. Cloud is characterized by features such as platform, location and device independency, which make it easily adoptable for all sizes of businesses, in particular small and mid-sized.[7],[5]. However, owing to redundancy of computer system networks and storage system cloud may not be reliable for data, but it scores well as far as security is concerned. . In cloud computing, security is tremendously improved because of a superior technology security system, which is now easily available and affordable. Yet another important characteristic of cloud is scalability, which is achieved through server virtualization. Some of the most important five key characteristics are,

A. On-demand Self Service

A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service's provider.

B. Broad Network Access

Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms.

C. Resource Pooling

The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model with different physical and virtual resources dynamically assigned and reassigned according to consumer demand.[1],[2]. There is a sense of location independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or data center) .

D. Measured Service

Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service.[6] Resource usage can be monitored, controlled, and reported providing transparency for both the provider and consumer of the utilized service.

E. Selection of Provider

A good service provider is the key to good service. So, it is imperative to select the right service provider. One must make sure that the provider is reliable, well-reputed for their customer service and should have a proven track record in IT- related ventures. As cloud computing has taken hold, there are six major benefits that have become clear,

IV.SYSTEM FLOW DIAGRAM AND DESIGN GOALS

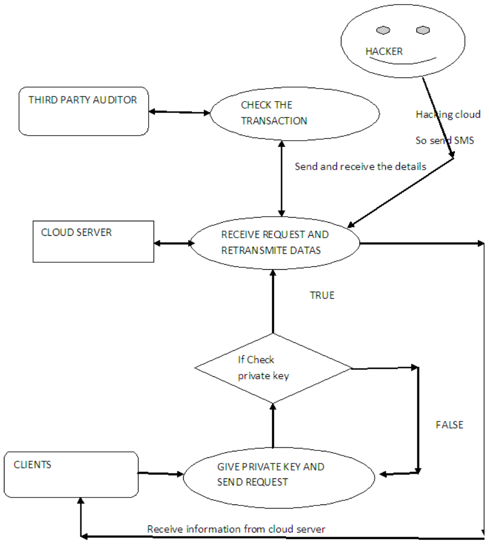

HACKER
THIRD PARTY AUDITOR
CHECK THE TRANSACTION
Hacking cloud
So send SMS
Send and receive the details
CLOUD SERVER
RECEIVE REQUEST AND RETRANSMITE DATAS
TRUE
If Check private key
FALSE
CLIENTS
GIVE PRIVATE KEY AND SEND REQUEST
Receive information from cloud server

Fig.2. System Flow Diagram

Our design goals can be summarized as the following:

1) Public auditability for storage correctness assurance: to allow anyone, not just the clients who originally stored the file on cloud servers, to have the capability to verify the correctness of the stored data on demand;

2) Dynamic data operation support: to allow the clients to perform block-level operations on the data files while maintaining the same level of data correctness assurance.[3],[9]. The design should be as efficient as possible so as to ensure the seamless integration of public auditability and dynamic data operation support;

3) Blockless verification: no challenged file blocks should be retrieved by the verifier (*e.g.*, TPA) during verification process for efficiency concern.

Designs for Blockless and Stateless verification

The naive way of realizing data integrity verification is to make the hashes of the original data blocks as the leaves in MHT, so the data integrity verification can be conducted without tag authentication and signature aggregation steps.[8],[10]. However, this construction requires the server to return all the challenged blocks for authentication, and thus is not efficient for verification purpose. Actually, one can easily defend this attack by storing the root *R* on the verifier, *i.e.*, *R* can be seen as public information. However, this makes the verifier not fully stateless in some sense since TPA will store this information for the rest of time.

Designs for Distributed Data Storage Security

To further enhance the availability of the data storage security, individual user's data can be redundantly stored in multiple physical locations. That is, besides being exploited at individual servers, data redundancy can also be employed across multiple servers to tolerate faults or server crashes as user's data grows in size and importance. By placing each of the m + k vectors on a different server, the original data file can survive the failure of any k of the m + k servers without any data loss.[6]. Such a distributed cryptographic system allows a set of servers to prove to a client that a stored file is intact and retrievable.

V. EXISTING SYSTEM

1. Firstly, traditional cryptographic primitives for the purpose of data security protection cannot be directly adopted due to the users' loss control of data under Cloud Computing. Therefore, verification of correct data storage in the cloud must be conducted without explicit knowledge of the whole data. Considering various kinds of data for each user stored in the cloud and the demand of long term continuous assurance of their data safety, the problem of verifying correctness of data storage in the cloud becomes even more challenging.[5].

2. Secondly, Cloud Computing is not just a third party data warehouse. The data stored in the cloud may be frequently updated by the users, including insertion, deletion, modification, appending, reordering, etc..[3],[2].

Disadvantages of existing system:

These techniques, while can be useful to ensure the storage correctness without having users possessing data, cannot address all the security threats in cloud data storage, since they are all focusing on single server scenario and most of them do not consider dynamic data operations.[2]. As an complementary approach, researchers have also proposed distributed protocols for ensuring storage correctness across multiple servers or peers. Again, none of these distributed schemes is aware of dynamic data operations. As a result, their applicability in cloud data storage can be drastically limited.

VI.PROPOSED SYSTEM

In this paper, we propose an effective and flexible distributed scheme with explicit dynamic data support to ensure the correctness of users' data in the cloud. We rely on erasure correcting code in the file distribution preparation to provide redundancies and guarantee the data dependability.

This construction drastically reduces the communication and storage overhead as compared to the traditional replication-based file distribution techniques. By utilizing the homomorphic token with distributed verification of erasure-coded data, our scheme achieves the storage correctness insurance as well as data error localization: whenever data corruption has been detected during the storage correctness verification, our scheme can almost guarantee the simultaneous localization of data errors, i.e., the identification of the misbehaving server(s).

VII.OUTPUT SCREEN SHOTS

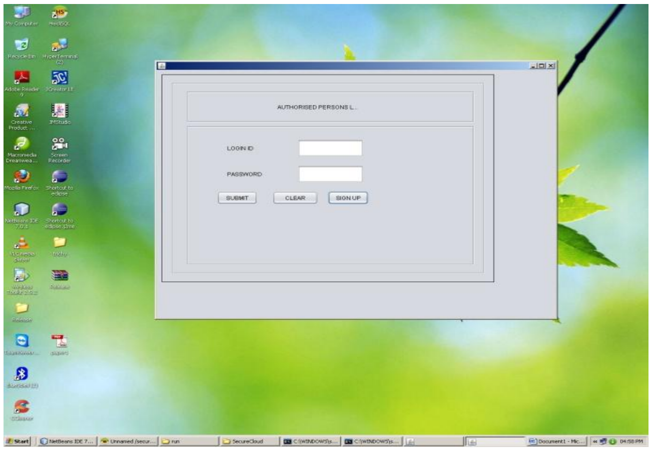
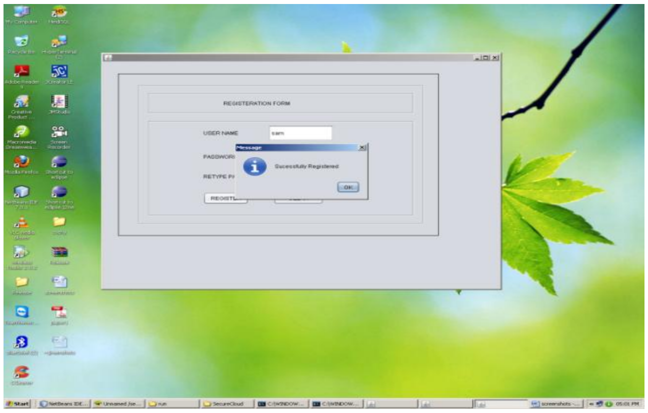


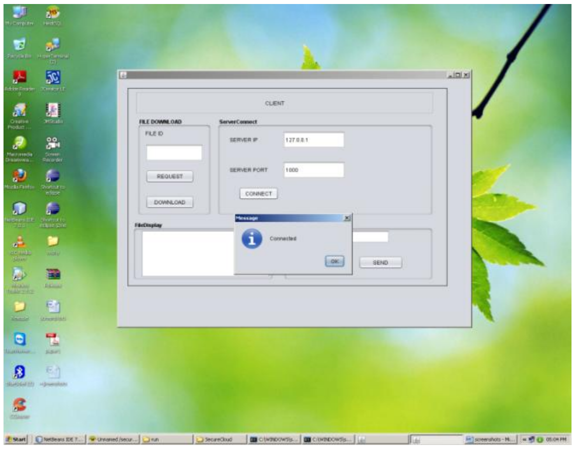
Fig 3.1 Authorised Persons



USER NAME sam

Sucessfully Registered

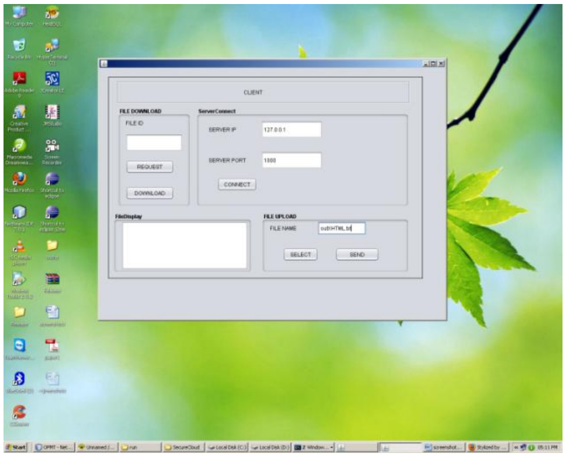
Fig 3.2 Registration Form



SERVER IP 127.0.0.1
SERVER PORT 1000

Message: Connected

Fig 3.3 Enter server ip and server port=1000 and click connect



SERVER IP 127.0.0.1
SERVER PORT 1000
FILE NAME outHTML.txt

Fig 3.4 Click select button to select a text file and Click send button in client to send the file to server

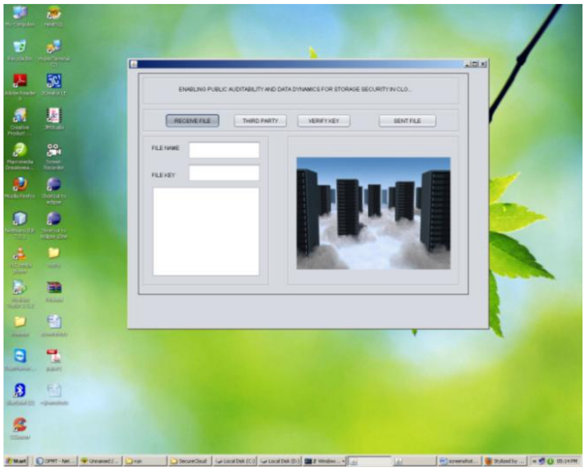
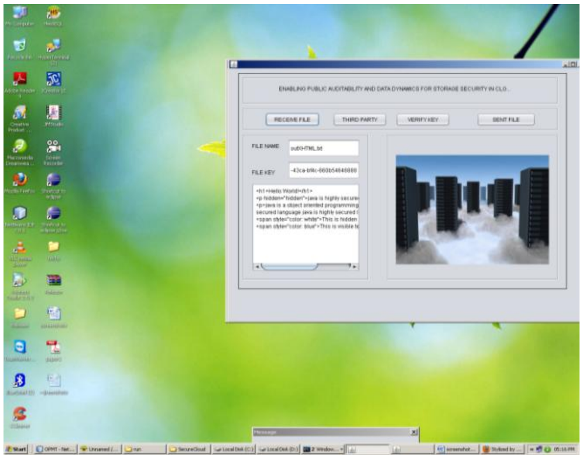

ENABLING PUBLIC AUDITABILITY AND DATA DYNAMICS FOR STORAGE SECURITY IN CLO...
RECEIVE FILE
THIRD PARTY
VERIFY KEY
SENT FILE
FILE NAME
FILE KEY

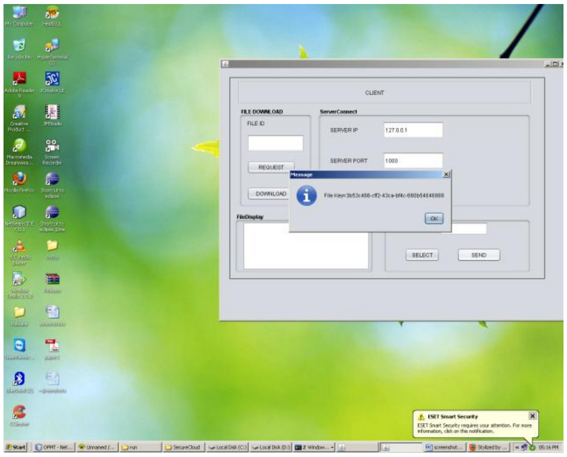
Fig 3.5 Click receive file in server


ENABLING PUBLIC AUDITABILITY AND DATA DYNAMICS FOR STORAGE SECURITY IN CLO...
RECEIVE FILE
THIRD PARTY
VERIFY KEY
SENT FILE

FILE NAME outHTML.txt
FILE KEY -43ca-bf4c-660b54646888

```
<h1>Hello World!</h1>
<p hidden="hidden">java is highly secure
<p>java is a object oriented programming
secured language java is highly secured
<span style="color: white">This is hidden
<span style="color: blue">This is visible to
```

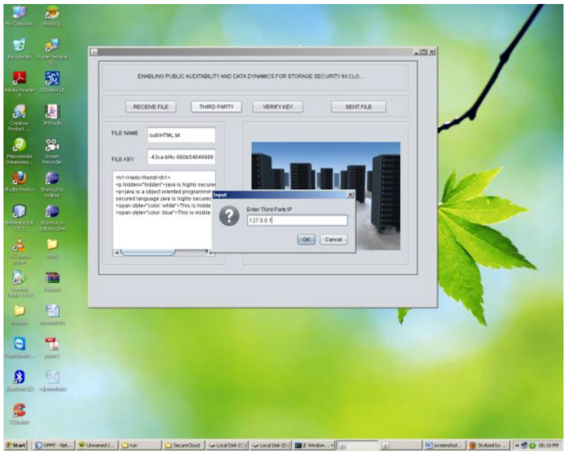
Fig 3.6 File received to server file key generated and displayed along with file



SERVER IP 127.0.0.1
SERVER PORT 1000

Message: File Key=3b53c486-cff2-43ca-bf4c-660b54646888

Fig 3.7 File key is sent to client


ENABLING PUBLIC AUDITABILITY AND DATA DYNAMICS FOR STORAGE SECURITY IN CLO...
RECEIVE FILE
THIRD PARTY
VERIFY KEY
SENT FILE
OK
Cancel

FILE NAME outHTML.txt
FILE KEY -43ca-bf4c-660b54646888

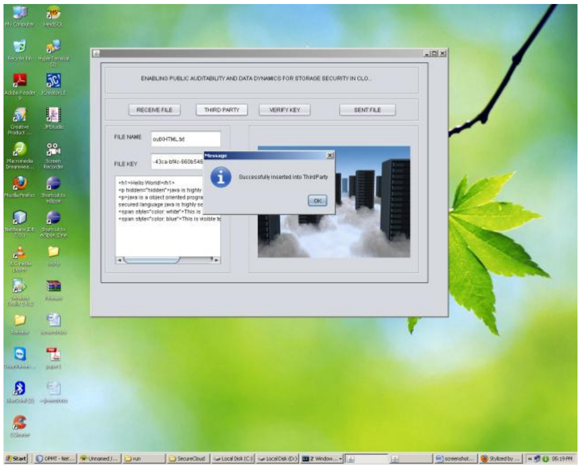
```
<h1>Hello World!</h1>
<p hidden="hidden">java is highly secure
<p>java is a object oriented programmin
secured language java is highly secured
<span style="color: white">This is hidde
<span style="color: blue">This is visible
```

Input: Enter Third Party IP 127.0.0.1

Fig 3.8 Enabling public auditability and data dynamics for data storage security

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 12, December 2013

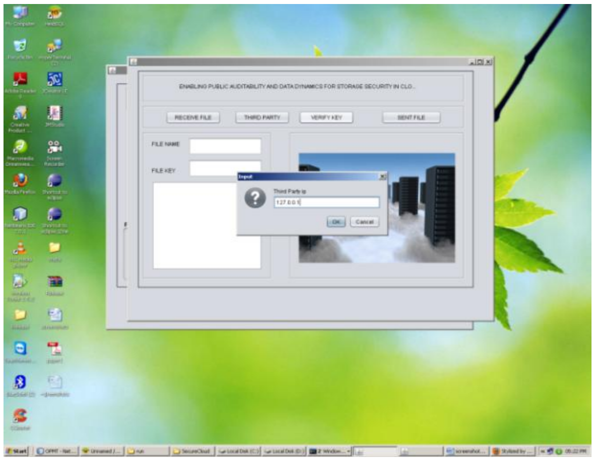

ENABLING PUBLIC AUDITABILITY AND DATA DYNAMICS FOR STORAGE SECURITY IN CLO...
RECEIVE FILE
THIRD PARTY
VERIFY KEY
SENT FILE
Message
OK

FILE NAME: outXHTML.txt
FILE KEY: -43ca-bf4c-660b546

```
<h1>Hello World!</h1>
<p hidden="hidden">java is highly
<p>java is a object oriented progra
secured language java is highly se
<span style="color: white">This is
<span style="color: blue">This is visible te
```

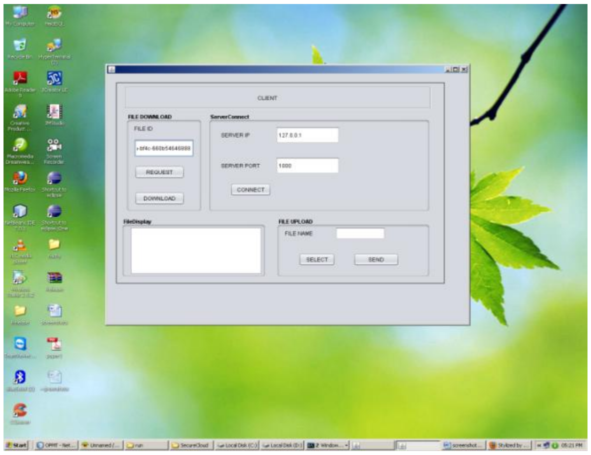
Successfully inserted into ThirdParty

Fig 3.9 File key stored in third party system:


ENABLING PUBLIC AUDITABILITY AND DATA DYNAMICS FOR STORAGE SECURITY IN CLO...
RECEIVE FILE
THIRD PARTY
VERIFY KEY
SENT FILE
FILE NAME
FILE KEY
Input
OK
Cancel

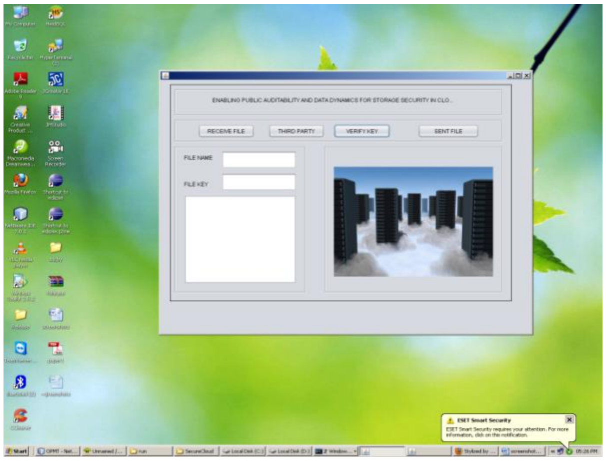
Third Party ip: 127.0.0.1

Fig 3.12 File key verified in server



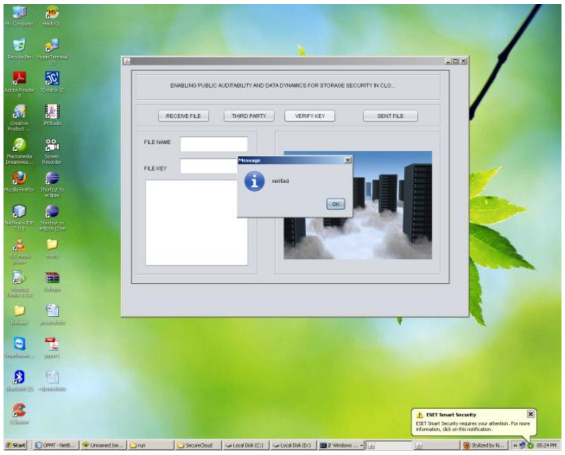
FILE ID: -bf4c-660b54646888
SERVER IP: 127.0.0.1
SERVER PORT: 1000

Fig 3.10 Enter file key and click request button



ESET Smart Security requires your attention. For more information, click on this notification.

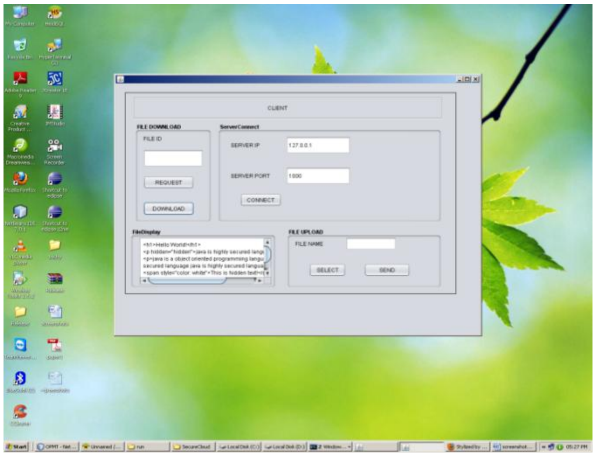
Fig 3.13 Click send file button in server:



verified

ESET Smart Security requires your attention. For more information, click on this notification.

Fig 3.11 Click verify key button and enter third party



SERVER IP: 127.0.0.1
SERVER PORT: 1000

```
<h1>Hello World!</h1>
<p hidden="hidden">java is highly secured langu
<p>java is a object oriented programming langu
secured language java is highly secured langua
<span style="color: white">This is hidden text</
```

Fig 3.14 File received to client and displayed

VIII. Advantages of Proposed system:

1. Compared to many of its predecessors, which only provide binary results about the storage state across the distributed servers, the challenge-response protocol in our work further provides the localization of data error.
2. Unlike most prior works for ensuring remote data integrity, the new scheme supports secure and efficient dynamic operations on data blocks, including: update, delete and append.
3. Extensive security and performance analysis shows that the proposed scheme is highly efficient and resilient against Byzantine failure, malicious data modification attack, and even server colluding attack.

IX. CONCLUTION

To ensure cloud data storage security, it is critical to enable a TPA to evaluate the service quality from an objective and independent perspective. Public auditability also allows clients to delegate the integrity verification tasks to TPA while they themselves can be unreliable or not be able to commit necessary computation resources performing continuous verifications. In this paper, we explored the problem of providing simultaneous public auditability and data dynamics for remote data integrity check in Cloud Computing. To achieve efficient data dynamics, we must improve the existing proof of storage models by manipulating the classic Merkle Hash Tree construction for block tag authentication. To support efficient handling of multiple auditing tasks, we further explore the technique of bilinear aggregate signature to extend our main result into a multiuser setting, where TPA can perform multiple auditing tasks simultaneously. Extensive security and performance analysis show that the proposed scheme is highly efficient and provably secure.

X. REFERENCES

[1] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in *Proc. of CCS'07*. New York, NY, USA: ACM, 2007, pp. 598-609.

[2] K. D. Bowers, A. Juels, and A. Oprea, "Hail: A high-availability and integrity layer for cloud storage," in *Proc. of CCS'09*. Chicago, IL, USA: ACM, 2009, pp. 187-198.

[3] D. Brackney, T. Goan, A. Ott, and L. Martin, "The Cyber Enemy within Countering the Threat from Malicious Insiders," Proc.Ann. Computer Security Applications Conf. (ACSAC).pp. 346-347, 2004.

[4] Cong Wang, Qian Wang, KuiRen, Wenjing Lou (2010),"Privacy Preserving Public Auditing for Data Storage Security in Cloud Computing".

[5] A. Conry-Murray, "The Threat from within. Network Computing (Aug. 2005),"http://www.networkcomputing.com/showArticle.jhtml?articleID=166400792, July 2009.

[6] A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica,M. Zaharia, "Above the clouds: A berkeley view of cloud computing," University of California, Berkeley, Tech. Rep,2009.

[7] R. Mogull, "Top Five Steps to Prevent Data Loss and Information Leaks. Gartner Research (July 2006)," http://www.gartner.com,2010.

[8] C. Wang, Q. Wang, K. Ren, and W. Lou, "Ensuring data storage security in cloud computing," in *Proc. of IWQoS'09*, Charleston, South Carolina, USA, 2009.

[9] Q. Wang, C. Wang, J. Li, K. Ren, W. Lou (2009), "Enabling public verifiability and data dynamics for storage security in cloud computing", in Proc. of ESORICS'09, Saint Malo.

[10] J. Widom and S. Ceri, Active Database Systems: Triggers and Rules for Advanced Database Processing. Morgan Kaufmann, 1995.

BIOGRAPHIES

K.Gayathri received the B.E. degree in Electronics and communication Engineering in 2008 followed by M.E. in Computer & communication Engineering in 2013 from Anna University, Guindy, Chennai. She is currently pursuing Ph.D., in the field of Computer and Communication Engineering. She is working as an Assistant Professor for various engineering colleges, since 2011. She published several international/national conferences. Her research interest involves in network security, image processing, extended up to cloud computing techniques. She is guiding many projects for UG, PG students. She is member of IEEE, IET and ISTE.

P.Umamaheswari received the B.E. degree in Electronics and Communication Engineering from the Anna University in 2010. She received the M.E. degree in Applied Electronics from Anna University, Chennai in 2013 and secured a reward of university rank holder among 350 Engineering colleges from Tamilnadu. She is currently working as an Assistant Professor in the department of Electronics and Communication Engineering in Jay Shriram Group of Institutions, Tirupur. She published several international/ national conferences. Her research interest involves in Wireless networks , Image processing, Microprocessor, Digital Electronics. She is guiding projects for UG, PG students.

P.Senthilkumar obtained his B.E. degrees in electrical and electronics engineering from Bharathiyar University in 1999 and received M.E. degree in Power System Engineering from CEG, Anna University in 2008. He currently is pursuing as research Scholar in the field of Power System Engineering. He specializes in harmonics, DC systems and digital signal systems. He is a member of IET, IEEE and life member in ISTE. From 2000 to 2010, he has been working as faculty for various engineering colleges and presently he is working as Assistant professor in ASTU. Ethiopia. He guided many projects for students and revolved around transistorized drives, induction heating converters and power flow control devices for power systems.