DYNAMIC DATA INTEGRITY AND CHECKPOINT RECOVERY USING PUBLIC AUDITING IN CLOUD STORAGE

Saranya Jothi C, Usha V, Alex David S
Veltech Dr. RR & Dr. SR University, Chennai, India

ABSTRACT

Cloud computing brings great challenges in the field of data security and security assurance. In this paper, we are analyzing the different types of cloud security issues. The key and signature are generated using the elliptic curve cryptography algorithm and Ron Rivest, Adi Shamir and Leonard Adleman algorithm to provide efficient security. Dynamic Merkle hash tree is constructed for providing dynamic data integrity. Public auditing is performed using Third party auditor (TPA). Checkpoint information is used for recovering the lost data blocks in the cloud storage. The main objective is to provide public auditing and data recovery to recover the lost data from the cloud storage and finally compare the key size, time taken for key generation and signature generation of ECC and RSA encryption algorithm.

Key words: Checkpoint recovery, Data integrity, RSA, ECC, Auditing.

http://www.iaeme.com/IJCIET/issues.asp?JType=IJCIET&VType=8&IType=9

1. INTRODUCTION

1.1. Cloud Computing

Cloud computing is developing as a key innovation for sharing assets by means of web. The fundamental goal of cloud computing is to improve utilization of circulated assets and tackle vast scale calculation issues. For instance, cloud computing can centre the energy of thousands of PCs on one issue, empowering scientists to do their work quicker than any time in recent memory. Separated from these, cloud computing faces a few security issues. Some of the issues are data security, data trustworthiness, Network security, data isolation and data classification.

1.2. Data Recovery

The way toward dealing with the information from harmed, fizzled, debased, or distant auxiliary stockpiling media when it can't be gotten to typically. Recuperation might be required because of physical harm to the capacity gadget or intelligent harm to the document framework that keeps it from being mounted by framework (OS). A portion of the procedures...
utilized for information recuperation are information reinforcement, checkpoint recuperation, and so forth.

1.3. Data Integrity
Information honesty infers that information ought to be sincerely put away on cloud servers, and any infringement (e.g., information is lost, adjusted, or bargained) are to be identified. In cloud storage, applications convey capacity as an administration. Servers keep a lot of information that have the capacity of being gotten to on uncommon events. The cloud servers are discomfited regarding both securities also, unwavering quality, which implies that information, might be lost or changed noxiously or, on the other hand inadvertently. Organization blunders may cause information misfortune (e.g., reinforcement and re- establish, information movement, and changing enrolments in P2P frameworks. Also, enemies may start assaults by exploiting information proprietor’s loss of control over their own information. Keeping in mind the end goal to tackle honesty checking issue in cloud storage, a great deal of works have been finished. Instruments giving information Integrity are merkle hash tree (MHT), bilinear isolation signature, third gathering reviewer (TPA), regenerating codes, Certificate less public examining.

1.4. Auditing
To check the honesty in cloud, different plans have been proposed [10], [11]. They all fall into two classifications: private auditability and open audit ability [9]. Private auditability bolsters the client to confirm the put away information respectability what's more, accessibility. In Public audit ability clients might be not ready to manage the cost of the overhead to verify the information uprightness oftentimes in light of the fact that TPA will regularly play out the confirmation of information trustworthiness in the information stockpiling. It has been demonstrated viable in check of trustworthiness and accessibility. So in functional application, the plan with open audit ability is more sensible.

2. RELATED WORKS
In enhancing reliability [1], ideal checkpoint strategy with edge switches disappointment mindful. The edge switch disappointment mindful checkpoint technique incorporates two calculations. The principal calculation utilizes the server farm topology and correspondence trademark for checkpoint picture stockpiling server determination. The second calculation utilizes the checkpoint picture stockpiling trademark and in addition the server farm topology to choose the recuperation server. Recreation tests are performed to show the adequacy of the proposed technique. Elliptical curve cryptography (ECC) [2], is an open key encryption capability in view of elliptic bend hypothesis that can be utilized to make quicker, littler, and more productive cryptographic keys. ECC creates keys through the properties of the elliptic bend condition rather than the conventional strategy acting of augmentation as the result of substantial prime numbers. The ECC stockpiling security conspire comprises of the accompanying four algorithmic projects. Key generation (Key-Gen) calculation is utilized to produce general society and basic trooper key for the signature melody tune. The Signature-era (Sig-Gen) calculation creates the mark for every datum record square. The Cogent confirmation - era (Proof-Gen) calculation creates the evidence for the test send. The last one Verify-confirmation (Ver-Proof) is completed to check the verification produced. RSA [3], is an open key cryptography strategy, broadly utilized for giving solid safety efforts. In RSA stockpiling security scheme, the information stockpiling accuracy is guaranteed and recognizable proof of getting disruptive server with high possibility is accomplished utilizing six calculations to be specific Key-Gen(), Sig-Gen(), Gen-Proof(), Verify Proof(), Execute-
Update() and Verify-Update(). Unique Provable Data Possession (PDP) The first PDP model[4] requires that the data is pre-processed in the setup stage keeping in mind the end goal to abandon some metadata on the client side for check purposes. Once the customer wanting to check the data dependability at a later time, he/she sends a test to the cloud server, which will respond with a substance in perspective of the data content. Subsequent to joining the answer and the neighbourhood meta-information, the client can demonstrate whether the honesty of the information is disregarded. It is like PDP [5]. The client stores just a input, which is utilized to encode the document F so as to get the scrambled record F.A set of haphazardly esteemed check pieces called sentinels. The undertaking is that an arrangement of guard esteems is inserted into F, and the server just stores F without knowing where the sentinel might be since the sentinels are vague from normal information squares. The verifier challenges the proved by indicating the places of a gathering of sentinels and requesting that the proved restore the related sentinel esteems. In case the proved has adjusted or eradicated a noteworthy piece of F, by then with high probability it will moreover have smothered various sentinels. It is in this way outlandish to respond precisely to the verifier. In this manner, a customer has proof to demonstrate that the server has ruined the document. POR embraces blunder remedying codes to recuperate the record with just a little part being undermined. A Merkle Hash Tree (MHT) [6], is a typical twofold hierarchy, which aid document trustworthiness confirmation. The thought is to split the document up into various little pieces, hash those piece, and after that frequentative consolidate and repeat the subsequent hashes in a tree-like form until the point when a solitary root hash is made. Honesty confirmation is finished with the base of the hash tree, base of the hash tree is marked with clients private key. So it can effectively and safely demonstrate that an arrangement of components is undamaged and unaltered.

3. PROPOSED FRAMEWORK
In dynamic data integrity, public auditability and checkpoint recovery is applied and under implementation. The detailed structural design is shown in the Figure 1.

![Proposed Framework](http://www.iaeme.com/IJCIET/index.asp)
In the proposed framework data are collected from the client and the collected data are split into number of fixed sized blocks. The splitted files need to be encrypted before storing in the cloud. Encryption of data blocks using public key of ECC and RSA. In the mean time, using sha-1 hash algorithm, hashes for the encrypted data blocks are found. Using the hash values, dynamic merkle hash tree is constructed. The root of the hash is sent to the TPA. TPA maintains the file recipe which contains the filename and the different data blocks that make up the file with its root hash. Encrypted data blocks are sent to the cloud for storage and they are stored in machines that are found in cloud. Commits are taken for each data blocks in cloud. For a group of commits, checkpoints are taken at regular intervals. When there is a system crash in the cloud storage, there is a possibility of failure in integrity checking. To overcome this problem, checkpoint recovery is used to know which data blocks are stored in the cloud and not stored. Resending of those lost data blocks to the cloud storage is achieved. To check the honesty of a customer's record, customer ask for the TPA to check the honesty of the record by determining its document name. TPA forwards the client’s request to the cloud storage. In response, cloud storage sends the encrypted data blocks of the requested file to the TPA. TPA constructs the DMHT for the encrypted data blocks to check the integrity. After the integrity checking, TPA informs the integrity status to client.

3.1. Encryption of data blocks using ECC and RSA

In this encryption method, first data are collected from client and they are split-up into several data blocks of fixed size 1.4 kb. ECC and RSA algorithms are used for encryption of data blocks. Public and Private keys are generated using the ECC and RSA algorithm. Using the public key of ECC algorithm, signature tag for data blocks are created. After encrypting the data blocks, hash algorithm (SHA-1) is used to construct the hash values for the encrypted data blocks. DMHT is to be constructed and hash value of the root is to be sent to TPA. Figure 2 depicts the entire flow.

Figure 2 Encryption of data blocks

Key Generation using ECC

- Select one distinct prime number R and S.
- And compute N=RS, Where N is used as the modules for both private and the public key.
- And its key length is expressed in bits.
- Choose an integer t such that 1 it if (N) and gcd (t, f (N)) =1; t and f (N) are co-prime.
- And d is the multiplicative inverse of t(modulo(N)), d is the private key.
3.2. Storing Encrypted Data in Cloud Storage
This module describes how the data blocks are getting stored in the cloud [8]. At First, we have to setup a private cloud storage using Eucalyptus. After the installation of cloud, the encrypted data blocks are sent to the cloud storage using APIs like S3curl, S3cmd and S3fs.

3.3. DMHT Construction
DMHT can expressly and effectivily handle completely powerful information operations counting information change (M), information addition (I) and information erasure (D) for cloud information stockpiling [12]. Sample hash tree Figure 3.3 depicts the sample hash tree, where h c = h(h(m1)k h(m2)) (i.e.) H(c) is constructed by the hash values of H(m1) and H(m2). In the way h f =h(h(m7)k h(m8)), h a = h(h c k h d), h b = h(h e k h f) and h R = h(h a k h b).

![Sample hash tree](image)

**Figure 3** Sample hash tree

3.3.1. Insert operation
Information inclusion alludes to embeddings new squares after some predefined positions in the information record. It doesn’t change the rationale structure of customer’s information record. In the figure 4, h (n_) is embedded after h (n2). Hash tree is like a paired tree, each root hub ought to have two youngsters. To accomplish this property hub c is shaped, h (n_) and h (n2) are its tyke.

![Data Insertion](image)

**Figure 4** Data Insertion
3.3.2. Modify Operation

Information alteration, which is a standout amongst the most every now and again, utilized operations in cloud information stockpiling. An essential information change operation alludes to the supplanting of indicated hinders with new ones. It doesn't change the rationale structure of customer’s information record. In Figure 5, h (n2) is supplanted by h (n0 2), with the goal that the parent hash is likewise changed.

![Figure 5 Data Deletion](image)

3.4. Checkpoint Recovery of Lost Blocks

Buckets and volumes are the persistent storage within Eucalyptus. The Storage Controller (SC) is the Eucalyptus component implementing the versatile Piece Storage [7], which enables occasions to utilize volumes. Volumes can be made, erased, joined to running cases, segregated, and snapshotted utilizing the EC2 interface, thus the euca tools has all the needed command such as euca-create-volume, euca-create-snapshot to operate on the volumes. Data blocks are stored in the systems in the cloud [8]. Cloud-output.log is a log file in the /var/log/eucalyptus path, maintains the log record for every action that takes place in the cloud. For each data block, cloud-output log file maintains the status of the stored data block, this status is considered as commit point. At regular intervals of time, an image of the path/var/lib/eucalyptus/bukkits/bucket name is created, run the instance of the image, attach a block of storage to that running instance, then take a snapshot of the running virtual machine. This snapshot is referred as checkpoint. When there is an accidental data lost in the cloud storage, there is a possibility of failure in integrity checking. To overcome this problem, commits are taken for each data block and checkpoints are taken for group of commits at regular intervals to know which data blocks are stored or not stored in the cloud [9]. If the system crashes or data gets lost or accidentally deleted the cloud server will roll back to the most recently taken checkpoint. As shown in figure 6, the lost encrypted data blocks need to be resend to the cloud storage.
3.5. Data Integrity Check using Public Auditing

The data integrity of the file by specifying stored in the cloud, client request the TPA to check the integrity of the file by specifying its file name [13]. TPA forwards the request to the cloud storage. In response, cloud sends the requested encrypted data blocks to the TPA. Then the TPA constructs the DMHT for the requested file. TPA matches the constructed root of DMHT with the hash value sent by the client to the TPA. If matches, client forwards the encrypted data block to the client, then the client using the private key decrypts the received file. If the root of DMHT doesn’t match the hash value, then the encrypted data block sent by the storage is modified one. The figure 7 depicts the process flow of data integrity checking using public auditing.
4. PERFORMANCE COMPARISON OF RSA AND ECC

The time taken for key generation and signature generation of RSA and ECC algorithm are compared. ECC algorithm consumed less time period for generating Public and private key and to encrypt the data. Therefore, the process is continued with data blocks encrypted using ECC. The below table 1 shows the obtained results.

<table>
<thead>
<tr>
<th>Time taken</th>
<th>RSA</th>
<th>ECC</th>
</tr>
</thead>
<tbody>
<tr>
<td>For key generation</td>
<td>6 seconds</td>
<td>1 second</td>
</tr>
<tr>
<td>For signature generation</td>
<td>15 seconds</td>
<td>1 second</td>
</tr>
</tbody>
</table>

![Figure 8 Analysis Chart](image)

5. CONCLUSIONS

The goal of this plan is to implement dynamic information respectability and checkpoint recovery. In this, integrity of the encrypted data, stored in the cloud storage is verified by TPA using dynamic merkle hash tree and checkpoint recovery is to restore the lost data blocks at the cloud storage due to system crash. Integrity of the data stored in the cloud is verified using public auditing. Checkpoint recovery is implemented by taking the snapshot of the memory state of a running VM of current instance. A block of volume or storage is attached to the running instance. Later the snapshot of the memory state is stored in the block storage. Future improvement of the development can be incorporating replication technologies to recover lost data instead of client re-sending encrypted data to cloud storage.

REFERENCES


Dynamic Data Integrity and Checkpoint Recovery Using Public Auditing in Cloud Storage


