DeyPoS: Deduplicatable Dynamic Proof of Storage for Multi-User Environments

C .Madhu[1], Dr. G.V.Ramesh Babu[2]
[1] *Student,* MCA, M.Tech, Ph.D., *Dept. of MCA SVU College of CM&CS*
[2] *Assistant Professor, Dept. of MCA SVU College of CM&CS*

Abstract- **In this Paper, Dynamic Proof of Storage (PoS) is a useful cryptographic primitive that enables a user to check the integrity of outsourced files and to efficiently update the files in a cloud server. Although researchers have proposed many dynamic PoS schemes in singleuser environments, the problem in multi-user environments has not been investigated sufficiently. A practical multi-user cloud storage system needs the secure client-side cross-user deduplication technique, which allows a user to skip the uploading process and obtain the ownership of the files immediately, when other owners of the same files have uploaded them to the cloud server. To the best of our knowledge, none of the existing dynamic PoSs can support this technique. In this paper, we introduce the concept of deduplicatable dynamic proof of storage and propose an efficient construction called DeyPoS, to achieve dynamic PoS and secure cross-user deduplication, simultaneously. Considering the challenges of structure diversity and private tag generation, we exploit a novel tool called Homomorphic Authenticated Tree (HAT). We prove the security of our construction, and the theoretical analysis and experimental results show that our construction is efficient in practice.**

DEDUPLICATION

Data deduplication is a specialized data compression technique for eliminating duplicate copies of repeating data. Related and somewhat synonymous terms are intelligent (data) compression and single-instance (data) storage. This technique is used to improve storage utilization and can also be applied to network data transfers to reduce the number of bytes that must be sent. In the deduplication process, unique chunks of data, or byte patterns, are identified and stored during a process of analysis.

EXISTING SYSTEM

To the best of our knowledge, none of the existing dynamic PoSs can support this technique.In most of the existing dynamic PoSs, a tag used for integrity verification is generated by the secret key of the uploader. Thus, other owners who have the ownership of the file but have not uploaded it due to the cross-user deduplication on the client-side, cannot generate a new tag when they update the file. In this situation, the dynamic PoSs would fail. If we take dynamic PoS and cross-user deduplication on the client-side as orthogonal issues, we may simply combine the existing dynamic PoS schemes and deduplication techniques. Then, structure diversity is solved via deduplication scheme.all existing techniques for cross-user deduplication on the client-side were designed for static files. Once the files are updated, the cloud server has to regenerate the complete authenticated structures for these files, which causes heavy computation cost on the server-side.

DISADVANTAGES:

Cross-user deduplication on the client-side, cannot generate a new tag when they update the file. In this situation, the dynamic PoSs would fail.As a summary, existing dynamic PoSs cannot be extended to the multi-user environment.Whenever data is transformed, concerns arise about potential loss of data. By definition, data deduplication systems store data differently from how it was written. As a result, users are concerned with the integrity of their data.One method for deduplicating data relies on the use of cryptographic hash functions to identify duplicate segments of data. If two different pieces of information generate the same hash value, this is known as a collision. The probability of a collision depends upon the hash function used, and although the probabilities are small, they are always non zero.

PROPOSED SYSTEMS

Proposed many dynamic PoS schemes in singleuser environments, the problem in multi-user environments has not been investigated sufficiently. A practical multi-user cloud storage system needs the secure client-side cross-user deduplication technique, which allows a user to skip the uploading process and obtain the ownership of the files immediately, when other owners of the same files have uploaded them to the cloud server. proposed a client-side deduplication scheme for encrypted data, but the scheme employs a deterministic proof algorithm which indicates that every file has a deterministic short proof. Thus, anyone who obtains this proof can pass the verification without possessing the file locally.We proposed the comprehensive requirements in multi-user cloud storage systems and introduced the model of deduplicatable dynamic PoS. We designed a novel tool called HAT which is an efficient authenticated structure Based.

ADVANTAGES

Storage-based data deduplication reduces the amount of storage needed for a given set of files. It is most effective in applications where many copies of very similar or even identical data are stored on a single disk—a surprisingly common scenario. In the case of data backups, which routinely are performed to protect against data loss, most data in a given backup remain unchanged from the previous backup. Common backup systems try to exploit this by omitting (or hard linking) files that haven't changed or storing differences between files. Neither approach captures all redundancies, however. Hard-linking does not help with large files that have only changed in small ways, such as an email database; differences only find redundancies in adjacent versions of a single file (consider a section that was deleted and later added in again, or a logo image included in many documents).

STORAGE outsourcing is becoming more and more attractive to both industry and academia due to the advantages of low cost, high accessibility, and easy sharing. As one of the storage outsourcing forms, cloud storage gains wide attention in recent years.

Data deduplication is used to reduce the number of bytes that must be transferred between endpoints, which can reduce the amount of bandwidth required. See WAN optimization for more information. Virtual servers benefit from deduplication because it allows nominally separate system files for each virtual server to be coalesced into a single storage space. At the same time, if a given server customizes a file, deduplication will not change the files on the other servers—something that alternatives like hard links or shared disks do not offer. Backing up or making duplicate copies of virtual environments is similarly improved.

IMPLEMENTATION

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and it's constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

MODULES

In this project we have following Three modules .

i).Cloud storage

ii).Dynamic proof of storage

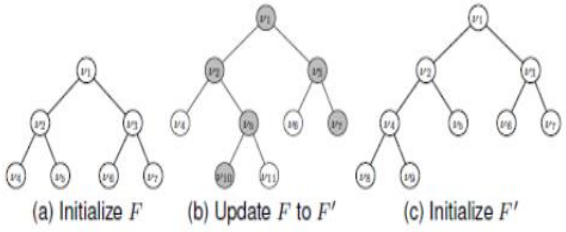
iii).Deduplication.

DYNAMIC PROOF OF STORAGE

Dynamic Proof of Storage (PoS) is a useful cryptographic primitive that enables a user to check the integrity of outsourced files and to efficiently update the files in a cloud server.we introduce the concept of deduplicatable dynamic proof of storage and propose an efficient construction called DeyPoS, to achieve dynamic PoS and secure cross-user deduplication, simultaneously. Considering the challenges of structure diversity and private tag generation, we exploit a novel tool called Homomorphic Authenticated Tree (HAT).we observe that our scheme is the only one satisfying the cross-user deduplication on the client-side and dynamic proof of storage simultaneously. Furthermore, the asymptotic performance of our scheme is better than

the other schemes except which only provides weak security guarantee.

HAT(Homomorphic Authenticated Tree):-



CLOUD STORAGE

which allows a user to skip the uploading process and obtain the ownership of the files immediately, when other owners of the same files have uploaded them to the cloud server.provide their own cloud storage services, where users can upload their files to the servers, access them from various devices, and share them with the others. Although cloud storage services are widely adopted in current days, there still remain many security issues and potential threats.Data integrity is one of the most important properties when a user outsources its files to cloud storage. Users should be convinced that the files stored in the server are not tampered. Traditional techniques for protecting data integrity, such as message authentication codes (MACs) and digital signatures, require users to download all of the files from the cloud server for verification, which incurs a heavy communication cost. These techniques are not suitable for cloud storage services where users may check the integrity frequently, such as every hour.

DEDUPLICATION

A practical multi-user cloud storage system needs the secure client-side cross-user deduplication technique, which allows a user to skip the uploading process and obtain the ownership of the files immediately.dynamic PoS remains to be improved in a multi-user environment, due to the requirement of cross-user deduplication on the client-side. This indicates that users can skip the uploading process and obtain the ownership of files immediately, as long as the uploaded files already exist in the cloud server.Thus, other owners who have the ownership of the file but have not uploaded it due to the cross-user deduplication on the client-side, cannot generate a new tag when they update the file. In this situation, the dynamic PoSs would fail.dynamic PoS and cross-user deduplication on the client-side as orthogonal issues, we may simply combine the existing dynamic PoS schemes and deduplication techniques.

ALGORITHMS

The deduplication proving algorithm

- Path search algorithm
- Sibling search algorithm

The deduplication proving algorithm:

Data deduplication is used to reduce the number of bytes that must be transferred between endpoints, which can reduce the amount of bandwidth required. See WAN optimization for more information. Virtual servers benefit from deduplication because it allows nominally separate system files for each virtual server to be coalesced into a single storage space. At the same time, if a given server customizes a file, deduplication will not change the files on the other servers—something that alternatives like hard links or shared disks do not offer. Backing up or making duplicate copies of virtual environments is similarly improved.

1.procedureDEDUPPROVE(αs, kc, αc, {c1, . . . , cn}, I,Q)
2: c ←0, t ←∅, ζ←1, l ←1
3: while ζ≤n do
4: δ←0
5: while ζ<ıjldo
6: δ←δ+ c_ ,ζ←ζ+ 1
7: pop the first element in Q
8: t ←t ∪ {fkc(iklikvi) + αcαsδ}
9: c ←c + c_
10: l ←l + 1, ζ←ζ+ 1
11: return c, t

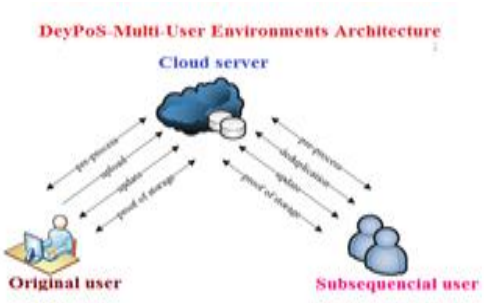
Path search algorithm:

It is clear that both the path search algorithm and the sibling search algorithm have the same computation complexity O(b log(n)), where b is the number of block indexes (i.e., the size of I) and n is the number of leaf nodes.

Sibling search algorithm:-

We define the sibling search algorithm $\psi \leftarrow$ Sibling(ρ) as Algorithm 2. It takes the path ρ as input, and outputs the index set of the siblings of all nodes in the path ρ. Note that, the output of the sibling search algorithm is not an ordered list. It always outputs the leftmost one in the remaining siblings.

ARCHITECTURE DIAGRAMS



System Configuration:

HARDWARE REQUIREMENTS:

Hardware	-	Pentium
Speed	-	1.1 GHz
RAM	-	1GB
Hard Disk	-	20 GB
Floppy Drive	-	1.44 MB
Key Board	-	Standard Windows Keyboard
Mouse	-	Two or Three Button Mouse
Monitor	-	SVGA

SOFTWARE REQUIREMENTS:

Operating System	: Windows
Technology	: Java and J2EE
Web Technologies	: Html, JavaScript, CSS
IDE	: My Eclipse
Web Server	: Tomcat
Tool kit	: Android Phone
Database	: My SQL
Java Version	: J2SDK1.5

CONCLUSION

The comprehensive requirements in multi-user cloud storage systems and introduced the model of deduplicatable dynamic PoS. We designed a novel tool called HAT which is an efficient authenticated structure. Based on HAT, we proposed the first practical deduplicatable dynamic PoS scheme called DeyPoS and proved its security in the random oracle model. The theoretical and experimental results show that our DeyPoS implementation is efficient, especially when the file size and the number of the challenged blocks are large.