**LITERATURE SURVAY**

**1.Secure Deduplication And Data Security With Efficient And Reliable CEKM**

**AUTHOR:**

**J. Pepitone, “Dropbox’s password nightmare highlights cloud risks,” June 2011.**

Secure deduplication is a technique for eliminating duplicate copies of storage data, and provides security to them. To reduce storage space and upload bandwidth in cloud storage deduplication has been a well-known technique. For that purpose convergent encryption has been extensively adopt for secure deduplication, critical issue of making convergent encryption practical is to efficiently and reliably manage a huge number of convergent keys. The basic idea in this paper is that we can eliminate duplicate copies of storage data and limit the damage of stolen data if we decrease the value of that stolen information to the attacker. This paper makes the first attempt to formally address the problem of achieving efficient and reliable key management in secure deduplication. We first introduce a baseline approach in which each user holds an independent master key for encrypting the convergent keys and outsourcing them. However, such a baseline key management scheme generates an enormous number of keys with the increasing number of users and requires users to dedicatedly protect the master keys. To this end, we propose Dekey, User Behaviour Profiling and Decoys technology. Dekey new construction in which users do not need to manage any keys on their own but instead securely distribute the convergent key shares across multiple servers for insider attacker. As a proof of concept, we implement Dekey using the Ramp secret sharing scheme and demonstrate that Dekey incurs limited overhead in realistic environments. User profiling and decoys, then, serve two purposes: First one is validating whether data access is authorized when abnormal information access is detected, and second one is that confusing the attacker with bogus information. We posit that the combination of these security features will provide unprecedented levels of security for the deduplication in insider and outsider attacker.

**2. A Secure Data Deduplication Scheme for Cloud Storage**

**AUTHOR:**

**Open Security Foundation: DataLossDB (http://datalossdb.org/).**

As more corporate and private users outsource their data to cloud storage providers, recent data breach incidents make end-toend encryption an increasingly prominent requirement. Unfortunately, semantically secure encryption schemes render various cost-effective storage optimization techniques, such as data deduplication, ineffective. We present a novel idea that differentiates data according to their popularity. Based on this idea, we design an encryption scheme that guarantees semantic security for unpopular data and provides weaker security and better storage and bandwidth benefits for popular data. This way, data deduplication can be effective for popular data, whilst semantically secure encryption protects unpopular content. We show that our scheme is secure under the Symmetric External Decisional Diffie-Hellman Assumption in the random oracle model.

**3. ClouDedup: Secure Deduplication with Encrypted Data for Cloud Storage**

**AUTHOR:**

**Amazon EC2. http://aws.amazon.com/ec2/.**

With the continuous and exponential increase of the number of users and the size of their data, data deduplication becomes more and more a necessity for cloud storage providers. By storing a unique copy of duplicate data, cloud providers greatly reduce their storage and data transfer costs. The advantages of deduplication unfortunately come with a high cost in terms of new security and privacy challenges. We propose ClouDedup, a secure and efficient storage service which assures block-level deduplication and data confidentiality at the same time. Although based on convergent encryption, ClouDedup remains secure thanks to the definition of a component that implements an additional encryption operation and an access control mechanism. Furthermore, as the requirement for deduplication at block-level raises an issue with respect to key management, we suggest to include a new component in order to implement the key management for each block together with the actual deduplication operation. We show that the overhead introduced by these new components is minimal and does not impact the overall storage and computational costs.