**Achieving A Data Secure Data Shared in Encrypted with Verification in Database**

**ABSTRACT**:

Secure search techniques over encrypted cloud data allow an authorized user to query data files of interest by submitting encrypted query keywords to the cloud server in a privacy-preserving manner. However, in practice, the returned query results may be incorrect or incomplete in the dishonest cloud environment. For example, the cloud server may intentionally omit some qualified results to save computational resources and communication overhead. Thus, a well-functioning secure query system should provide a query results verification mechanism that allows the data user to verify results. In this paper, we design a secure, easily integrated, and fine-grained query results verification mechanism, by which, given an encrypted query results set, the query user not only can verify the correctness of each data file in the set but also can further check how many or which qualified data files are not returned if the set is incomplete before decryption. The verification scheme is loose-coupling to concrete secure search techniques and can be very easily integrated into any secure query scheme. We achieve the goal by constructing secure verification object for encrypted cloud data. Furthermore, a short signature technique with extremely small storage cost is proposed to guarantee the authenticity of verification object and a verification object request technique is presented to allow the query user to securely obtain the desired verification object. Performance evaluation shows that the proposed schemes are practical and efficient.

|  |  |
| --- | --- |
| **EXSISTING SYSTEM** | **PROPOSED SYSTEM** |
| * Recently, with the growing popularity of cloud computing, how to securely and efficiently search over encrypted cloud data becomes a research focus. * Wang et al. applied hash chain technique to implement the completeness verification of query results by embedding the encrypted verification information into their proposed secure searchable index. * Sun et al. used encrypted index tree structure to implement secure query results verification functionality. | * In this paper, we extend and reinforce our work to make it more applicable in the cloud environment and more secure to against dishonest cloud server. * We formally propose the verifiable secure search system model and threat model and design a fine-grained query results verification scheme for secure keyword search over encrypted cloud data. * We propose a short signature technique based on certificateless public-key cryptography to guarantee the authenticity of the verification objects themselves. |

|  |  |
| --- | --- |
| **EXISTING ALGORITHM**  Security of Verification Object | **PROPOSED ALGORITHM:-**  Attribute-based encryption technique (ABE) |
| **ALGORITHM DEFINITION:-**  In this paper, compared with the previous works, animportant distinction about the threat model is that thecloud is considered to be an untrusted entity. More  specifically, first of all, the cloud server tries to gain somevaluable information from encrypted data files, secureindexes, and verification objects (e.g., a misbehavingcloud administrator aims at obtaining these informationfor possible monetary profits). Then, the cloud server  would intentionally return false search results for savingcomputation resource or communication cost. | **ALGORITHM DEFINITION:-**  Theencrypted verification information into their proposedsecure searchable index. In [24], Sun et al. used encryptedindex tree structure to implement secure query resultsverification functionality. In this scheme, when thequery ends, the cloud server returns query results alongwith a minimum encrypted index tree, then the datauser searches this minimum index tree using the samesearch algorithm as the cloud server did to finish resultverification. |
| **DRAWBACKS:-**   * cloud data based on attribute-based encryption technique (ABE) in the public-key setting * less security | **ADVANTAGES:-**   * honest-but-curious” entity * keeps robust and secure software/hardware environments. |

**SYSTEM REQUIREMENTS:**

**HARDWARE REQUIREMENTS**:

System : Pentium i3 Processor

Hard Disk : 500 GB.

Monitor : 15’’ LED

Input Devices : Keyboard, Mouse

RAM : 2 GB

**SOFTWARE REQUIREMENTS:**

Operating system : Windows 7.

Coding Language : Java.

Tool : Eclipse

Database : MYSQL