**Data table**

|  |  |
| --- | --- |
| **Field** | **Description** |
| MessageId | Unique identifier [primary key] |
| Message | Cyphertext |

**Index table**

|  |  |
| --- | --- |
| **Field** | **Description** |
| KeywordRefId | Unique identifier [primary key] |
| MessageId | A reference to Data.MessageId [foreign key] |
| KeywordHash | SHA256(H’(kwi)) Keyword hash as defined by LSABE-MA algorithm [indexed] |

Let l1 be the number of keywords for the message as defined in the paragraph D.2 of section V. Construction of LSABE-MA

For each message there will be l1 records in the index table – one per keyword defined for this message

**Extension for Encryption**

(Item 4 of Paragraph D.2 of section V. Construction of LSABE-MA)

When cyphertext is outsourced to the cloud it is extended with blind indexes SHA256(H’(kwi)) , i =0..l1-1

**Prelude for Search**

(Paragraph E of section V. Construction of LSABE-MA)

The search request is extended with blind indexes SHA256(H’(kwj)) , j =0..l2-1

**Search algorithm**

Set of data records: S  
Set of index records: R  
Integer: j

S = IndexTable  
j=0

while (number of records in S is greater than 1) and (j is less than l2)  
 R = subset of IndexTable such that (R.KeywordHash is equal to SHA256(H’(kwj)))   
 S = subset of S such that (S.MessageId is equal to one of R.MessageId)  
 j = j+1

if (number of records in S is greater than 1) is 0 then no records match give keyword set  
else trapdoor method is applied to all records in S

If binary search is used complexity of that search would be O(K \* log (M)) where K is the number of keywords and M is the number of messages in the storage