CLAIMS:

1. A method for querying a symmetrically encrypted database for conjunctive keywords comprising:

symmetrically decrypting the encrypted database and associating each keyword from the decrypted database with a file identity;

generating an encrypted index table using keywords from the decrypted database, the encrypted index table comprising symmetrically encrypted keywords and their associated index pairs, wherein each encrypted keyword that is encrypted using secret key K_T is associated with at least an index pair, and wherein each index pair comprises an *e* parameter and a *y* parameter;

populating a Bloom filter with elements that each contain a symmetrically encrypted keyword from the encrypted database and a symmetrically encrypted file identity that is associated with the keyword;

receiving queried conjunctive keywords, wherein the queried conjunctive keywords comprises at least one disjunction;

selecting a shortest clause from the conjunctive keywords, wherein the shortest clause is a disjunction;

generating trapdoors for all keywords contained in the selected shortest clause and retrieving corresponding index pairs for the keywords from the encrypted index table using the generated trapdoors;

computing a group of search tokens for each of the keywords in the shortest clause wherein each group of search tokens is associated with an index pair retrieved for a keyword from the selected shortest clause and with keywords from other clauses in the conjunctive keywords;

determining that the conjunctive keyword permutations are available in the encrypted database by evaluating a group of computed search tokens with operators contained in the other clauses; and

generating a decryption key for a keyword from the shortest clause that is associated with the group of computed search tokens using the following equation:

$$K_E = F(K_S, q_i)$$

where q_i is the i^{th} selected keyword from the shortest clause, Ks is a secret key and F is a pseudorandom function; and

decrypting an *e* parameter using the following equation to obtain a file identity associated with the queried conjunctive keywords:

$$\text{file ID} = \text{Dec}(K_E, e)$$

where file ID is the file identity associated with q_i and Dec is a symmetric decryption function.

2. The method according to claim 1 wherein the e parameter in the index table is calculated using the following equation:

$$e = \text{Enc}(K_E, \text{file ID})$$

where $K_E = F(K_S, w_i)$, F is a pseudorandom function, w_i represents a keyword from the database, file ID is a file identity associated with w_i, K_S is a secret key and Enc is a symmetric encryption function.

3. The method according to claim 1 wherein the y parameter in the index table is calculated using the following equation:

$$y = \frac{\text{xid}_{file\ ID}}{F_P(K_Z, w_i || c)}$$

where $\text{xid}_{\text{file ID}} = F_P(K_{ID}, \text{file ID})$, F_P is a pseudorandom function, K_Z and K_{ID} are secret keys, w_i represents a keyword from the database, file ID is a file identity associated with w_i, and c is a counter to track the number of index pairs created for w_i.

4. The method according to claim 1 wherein each element in the Bloom Filter is defined by the following equation:

$$g^{F_p(K_X, w_i) \cdot xid_{file\ ID}}$$

where $\text{xid}_{\text{file ID}} = F_P(K_{ID}, \text{file ID})$, F_P is a pseudorandom function, K_X and K_{ID} are secret keys, w_i represents a keyword from the database, and file ID is a file identity associated with w_i.

5. The method according to claim 1 wherein the computing the group of search tokens for each of the keywords in the shortest clause comprises computing each search token using the following equation:

$$\text{token}[c][j] = g^{F_p(K_X, q_j) \cdot F_p(K_Z, q_i || c) \cdot y}$$

where q_j is the j^{th} selected keyword from the other clauses, q_i is the i^{th} selected keyword from the shortest clause, c is a counter showing the number of an index pair retrieved for q_i; j is

the number of the selected keyword from the other clauses, F_P is a pseudorandom function, K_X, K_Z are secret keys and *y* is a parameter obtained from the selected index pair.

6. The method according to claim 1 wherein the generating of trapdoors for all keywords contained in the selected shortest clause comprises:

encrypting each keyword in the shortest clause using secret key K_T.

7. The method according to claim 1 wherein the determining that the conjunctive keyword permutations are available in the encrypted database by evaluating a group of computed search tokens with operators contained in the other clause comprises:

creating a test array for storing an array of TRUE or FALSE results;

comparing each search token from the computed group of search tokens with elements contained in the Bloom filter to determine whether there is a match, and populating the test array with the results from the comparison;

evaluating the populated test array with an operator array, wherein the operator array contains operators from the other clauses; and

providing a positive or negative result depending on the evaluation outcome.

8. A system for querying a symmetrically encrypted database for conjunctive keywords comprising:

a processing unit; and

a non-transitory media readable by the processing unit, the media storing instructions that when executed by the processing unit, cause the processing unit to:

decrypt the encrypted database and associate each keyword from the decrypted database with a file identity;

generate an encrypted index table using keywords from the decrypted database, the encrypted index table comprising symmetrically encrypted keywords and their associated index pairs, wherein each encrypted keyword that is encrypted using secret key K_T is associated with at least an index pair, and wherein each index pair comprises an *e* parameter and a *y* parameter;

populate a Bloom filter with elements that each contain a symmetrically encrypted keyword from the encrypted database and a symmetrically encrypted file identity that is associated with the keyword;

receive queried conjunctive keywords, wherein the queried conjunctive keywords comprises at least one disjunction in the query;

selecting a shortest clause from the conjunctive keywords, wherein the shortest clause is a disjunction;

generate trapdoors for all keywords contained in the selected shortest clause and retrieving corresponding index pairs for the keywords from the encrypted index table using the generated trapdoors;

compute a group of search tokens for each of the keywords in the shortest clause wherein each group of search tokens is associated with an index pair retrieved for a keyword from the selected shortest clause and with keywords from other clauses in the conjunctive keywords;

determine that the conjunctive keyword permutations are available in the encrypted database by evaluating a group of computed search tokens with operators contained in the other clauses; and

generate a decryption key for a keyword from the shortest clause that is associated with the group of computed search tokens using the following equation:

$$K_E = F(K_S, q_i)$$

where q_i is the i^{th} selected keyword from the shortest clause, Ks is a secret key and F is a pseudorandom function; and

decrypt an e parameter using the following equation to obtain a file identity associated with the queried conjunctive keywords:

$$\text{file ID} = \text{Dec}(K_E, e)$$

where file ID is the file identity associated with q_i and Dec is a symmetric decryption function.

9. The system of claim 8 wherein the instructions to generate the encrypted index table comprise:

instructions for directing the processing unit to:

compute the e parameter in the index table using the following equation:

$$e = \text{Enc}(K_E, \text{file ID})$$

where $K_E = F(K_S, w_i)$, F is a pseudorandom function, w_i represents a keyword from the database, file ID is a file identity associated with w_i, K_S is a secret key and Enc is a symmetric encryption function.

10. The system of claim 8 wherein the instructions to generate the encrypted index table comprise:

instructions for directing the processing unit to:

compute the y parameter in the index table using the following equation:

$$y = \frac{\text{xid}_{file\ ID}}{F_P(K_Z, w_i || c)}$$

where $\text{xid}_{\text{file ID}} = F_P(K_{ID}, \text{file ID})$, F_P is a pseudorandom function, K_Z and K_{ID} are secret keys, w_i represents a keyword from the database, file ID is a file identity associated with w_i, and c is a counter to track the number of index pairs created for w_i.

11. The system of claim 8 wherein the instructions to populate the Bloom Filter comprise:

instructions for directing the processing unit to:

compute each element in the Bloom Filter using the following equation:

$$g^{F_p(K_X, w_i) \cdot xid_{file\ ID}}$$

where $\text{xid}_{\text{file ID}} = F_P(K_{ID}, \text{file ID})$, F_P is a pseudorandom function, K_X and K_{ID} are secret keys, w_i represents a keyword from the database, and file ID is a file identity associated with w_i.

12. The system of claim 8 wherein the instructions to compute the group of search tokens for each of the keywords in the shortest clause comprise:

instructions for directing the processing unit to:

compute each search token using the following equation:

$$\text{token[c][j]} = g^{F_p(K_X, q_j) \cdot F_p(K_Z, q_i || c) \cdot y}$$

where q_j is the j^{th} selected keyword from the other clauses, q_i is the i^{th} selected keyword from the shortest clause, c is a counter showing the number of an index pair retrieved for q_i; j is the number of the selected keyword from the other clauses, F_P is a pseudorandom function, K_X, K_Z are secret keys and *y* is a parameter obtained from the selected index pair.

13. The system of claim 8 wherein the instructions to generate trapdoors for all keywords contained in the selected shortest clause comprise:

instructions for directing the processing unit to:

encrypt each keyword in the shortest clause using secret key K_T.

14. The system of claim 8 wherein the instructions to determine that the conjunctive keyword permutations are available in the encrypted database by evaluating a group of computed search tokens with operators contained in the other clause comprise:

instructions for directing the processing unit to:

create a test array for storing an array of TRUE or FALSE results;

compare each search token from the computed group of search tokens with elements contained in the Bloom filter to determine whether there is a match, and populating the test array with the results from the comparison;

evaluate the populated test array with an operator array, wherein the operator array contains operators from the other clauses; and

provide a positive or negative result depending on the evaluation outcome.

15. The method according to claim 1, wherein the shortest clause is without a NOT operator.

16. The system of claim 8, wherein the shortest clause is without a NOT operator.