Note:

1. PK denotes Public Key.
2. SK denotes Secret Key (or Private Key)
3. DO denotes Data Owner
4. QU denotes Query User
5. CS denotes Cloud Server
6. E1 denotes Enclave E1 (Used for Data Storage and KNN computation)
7. E2 denotes Enclave E2 (Used for interfacing to decouple QU's identity)
8. Secret Keys are generated inside enclaves and will never leave the enclave. All decryption will happen inside the enclaves and not in the untrusted apps or cloud.

Key's Details

Key	Type of Key	Key Made By Whom	For What Purpose
K_{DO}	AES	Data Owner (1 time work)	1. Encryption of database by DO (to improve performance). 2. Decryption of database inside Enclave E1. 3. Encryption of ML code (e.g. KNN) by DO if required (for code confidentiality). 4. Decryption of ML code (e.g. KNN) inside Enclave E1 if required (for code confidentiality).
(PK_{E1}, SK_{E1})	RSA	Enclave E1 (1 time work)	**1. Secure communication between Enclave E1 and DO:** DO uses PK_{E1} to send AES Key K_{DO} in encrypted form to Enclave E1 who alone can decrypt it using SK_{E1}. With K_{DO}, E1 can decrypt DataBase.

			2. QU uses PK_{E1} to send AES Key K_{QU} in encrypted form to Enclave E1 through Enclave E2. Only E1 alone can decrypt and get key K_{QU} using SK_{E1}. So, the query is hidden from E2.
(PK_{E2}, SK_{E2})	RSA	Enclave E2 (1 time work)	**1. Secure communication between Enclave E2 and QU:** QU uses PK_{E2} to send the entire block (containing encrypted query, encrypted query's decryption key, payment info or credential, and another key (PK_Q)) in encrypted form to Enclave E2 who alone can decrypt it using SK_{E2}. So E1, DO, and CS cannot see and use credential or IP address or PK_Q to link QU's identity to the query. a. Why encrypted query and encrypted query's decryption key are encrypted again by E2's public key? Reason: Encrypted query and Encrypted query's decryption key are sent to Enclave E1 by Enclave E2. If encryption with E2's public key is not used, then E1 can link the Encrypted query to its owner by tapping the line between QU and E2 as it will know the IP address from where the encrypted query is coming. b. Why are credentials or payment info encrypted by E2's public key? Reason: To hide credentials or payment info from other parties. c. Why is PK_Q encrypted by E2's public key? Reason: The final KNN result will be returned to the query user by E1 through E2. E2 uses PK_Q to encrypt

			the encrypted result returned by E1. If encryption with PK_Q is not used, then E1 can link the encrypted returned result to its owner by tapping the line between QU and E2 as it will know the IP address to where the encrypted result is going.
K_{QU}	AES	Query User (per query)	1. Encryption of query by QU (to improve performance). 2. Decryption of query inside Enclave E1. 3. Encryption of KNN results by Enclave E1 so that only QU can decrypt it (Hiding Access Pattern from other parties). 4. Decryption of KNN results by QU.
(PK_Q, SK_Q)	RSA	Query User (1 time work)	1. Encryption of returned (encrypted) knn computation result (from Enclave E1) by Enclave E2. Reason: The final KNN result will be returned to the query user by E1 through E2. E2 uses PK_Q to encrypt the encrypted result returned by E1. If encryption with PK_Q is not used, then E1 can link the encrypted returned result to its owner by tapping the line between QU and E2 as it will know the IP address to where the encrypted result is going. 2. SK_Q used by QU to decrypt KNN results.