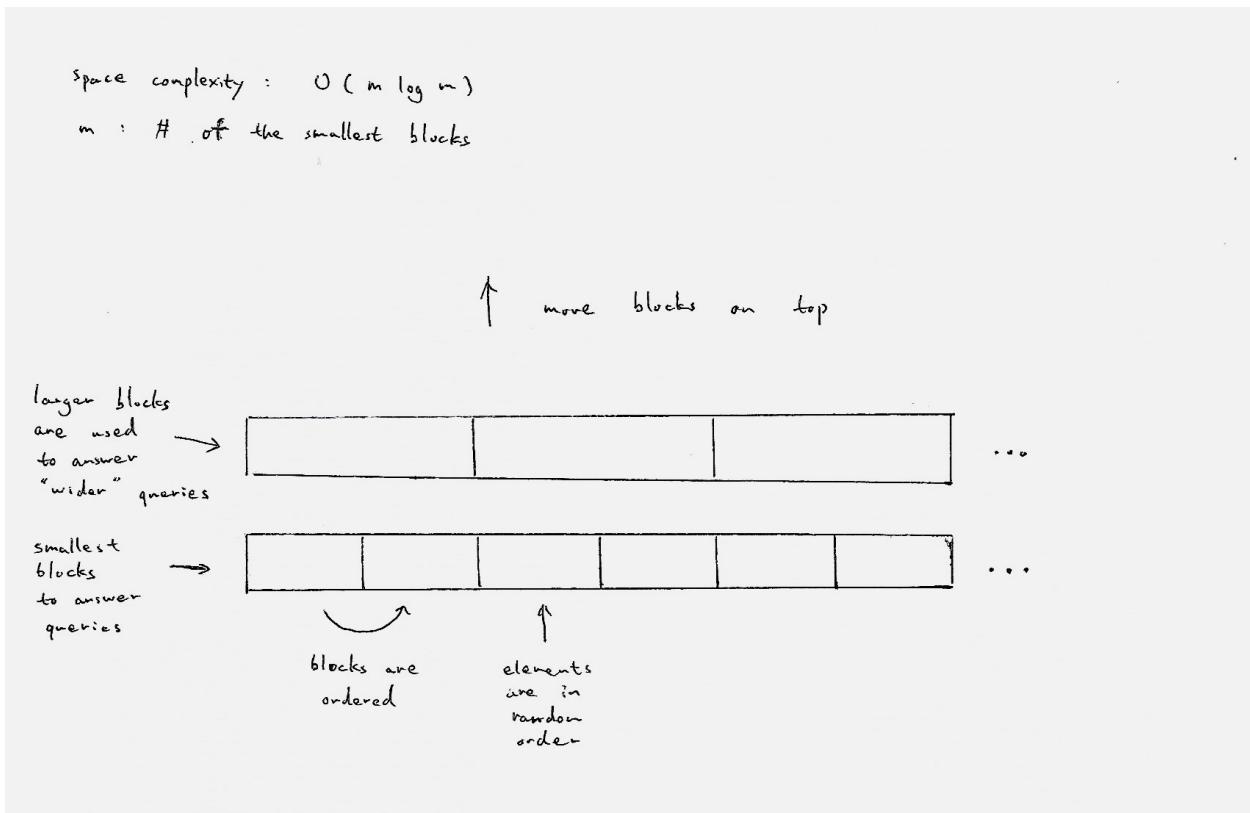


1 Papers referred to

1. Marie-Sarah Lacharite, Brice Minaud. **Improved Reconstruction Attacks on Encrypted Data Using Range Query Leakage**
2. **K-Nearest-Neighbour Search on Encrypted Data** (Not published yet)
3. Rafail Ostrovsky. **Software Protection and Simulation on Oblivious RAMs**
4. Data structure and algorithms I have learnt during undergraduate.

2 A new OPE scheme

- The idea is to use ‘searchable’ data structures to pre-process the queries so that the queries do not leak the sets used in 1.
- Sketch of the scheme:



- Instead of lookup table to identify the block to retrieve, one can use PRF on the endpoints (in terms of range covered by the block) of the block.
- Claim: the block returned contains at most twice number of items than the actual items.
- Problems with the scheme:
 - The claim is not quite true: consider the range spanning the second and third block in the bottom layer, there is no single block to return from the layer on top. However, there is a simple fix to this, and does not change space complexity (**To be discussed in the next meeting**).

- If the range is 1 to 2^k then this scheme is not ideal in terms of space complexity. I claim that the scheme can be modified to achieve a very low overhead (**To be discussed in the next meeting**).
- To think: what does the scheme leak?

3 ORAM

- Confirms that the lower-bound of overhead is indeed $\mathcal{O}(\log n)$.
- Question to be answered: is ORAM any similar to OPE in the sense that, in order to hide access pattern, how much overhead does it cost?

4 Syntax of encrypted database

- We can think of database in three levels: a string of zeros and ones in the most abstract form, a database as a set of strings, and a database with all the details.
- We will formalize syntax on the most abstract form first.
- Notation (need better names):
 - **Initialise**: $1^\lambda \rightarrow \text{params}^*$
 - **KeyGen**: $1^\lambda \rightarrow \text{sk}$
 - **Enc**: $1^\lambda \times \text{sk} \times \mathbf{DB} \rightarrow \mathbf{EDB}$
 - **Dec**: $1^\lambda \times \text{sk} \times \mathbf{EDB} \rightarrow \mathbf{DB}$
 - **Q**: $\mathbf{DB} \times Q \rightarrow \mathbf{DB}$
 - **encQ**: $1^\lambda \times \text{sk} \times Q \rightarrow \mathbf{eQ} \times \mathbf{iQ}$
 - **EQ**: $1^\lambda \times \text{sk} \times \mathbf{EDB} \times \mathbf{eQ} \rightarrow \mathbf{EDB}$
 - **DQ**: $1^\lambda \times \text{sk} \times \mathbf{EDB} \times \mathbf{iQ} \rightarrow \mathbf{DB}$
- Additional notation:
 - Π_i : projection of i -th component of the arguments
 - id : identity function
- Commutative diagram:

