RGC Ref. No.:
UGC/FDS14/E05/14
(please insert ref. above)

RESEARCH GRANTS COUNCIL
COMPETITIVE RESEARCH FUNDING SCHEMES FOR THE LOCAL SELF-FINANCING DEGREE SECTOR

FACULTY DEVELOPMENT SCHEME (FDS)

Completion Report
(for completed projects only)

Submission Deadlines:
1. *Auditor's report with unspent balance, if any: within **six** months of the approved project completion date.*
2. *Completion report: within **12** months of the approved project completion date.*

Part A: The Project and Investigator(s)

1. Project Title

Improving selection query processing speed of secure cloud database systems by tuple pruning on desensitized data

2. Investigator(s) And Academic Department(s) / Unit(s) Involved

Research Team	Name / Post	Unit / Department / Institution
Principal Investigator	Dr. Wong Wai Kit, Assistant Professor	Department of Computing, Hang Seng Management College
Co-Investigator(s)	Dr. Yue Ho Yin, Senior Lecturer	Department of Computing, Hang Seng Management College

3. Project Duration

	Original	Revised	Date of RGC / Institution Approval *(must be quoted)*
Project Start Date	1 Jan 2015		
Project Completion Date	31 Dec 2016	30 Jun 2017	5 May 2016
Duration *(in month)*	24	30	5 May 2016
Deadline for Submission of Completion Report	31 Dec 2017	30 Jun 2018	5 May 2016

Part B: The Final Report

5. Project Objectives

5.1 Objectives as per original application

1. Define a proper security definition that is suitable for the pruning scenario and desensitization.

2. Study how the security protection is affected when we apply our security definition on existing secure database systems.

3. Develop data desensitization technique to generate auxiliary data, that is secure under our defined security definition.

4. Develop query transformation technique to transform a plain query to a query on desensitized data.

5. Develop efficient algorithms to process a query on desensitized data.

6. Implement the developed techniques on an existing secure database system and carry out performance study.

5.2 Revised objectives

N/A

5.3 Realisation of the objectives

(Maximum 1 page; please state how and to what extent the project objectives have been achieved; give reasons for under-achievements and outline attempts to overcome problems, if any)

Objective 1:

The original plan was to identify a security model for the desensitization scenario, which aimed to balance between information leakage and efficiency of the optimization method. Intuitively, if more information is leaked, the optimization method is more efficient. In this project, we investigated existing secure cloud database systems and identified an oracle model to describe the query processing mechanisms of these systems. We also observed that the oracle model leaks certain information. We defined the security models such that the idea behind these models is that no more information is leaked in our optimization method. We remark that our (ideal version of) security model is the best any secure database system can achieve unless the system does not follow the oracle model.

Objective 2:

See description of Objective 1. In our project, the security analysis on existing database systems and the security definition mutually affect each other. The security analysis covered all major secure database systems that we were aware of.

Objective 3:

We developed two approaches to build the index (the auxiliary data).

The first approach is called SanTable. Recall that oracle model leaks certain information. When an attacker gathers such information, the attacker can derive an estimated value of each data item with bounded error. In this method, the user actively carries out such attack and computes the estimated range of data. The estimated data ranges form SanTable and it is sent to the server. Since the server can perform the attack anyway, sending SanTable to server does not leak additional information to the server. This method require user's involvement to generate the SanTable. Details of the method can be found in our paper in DEXA 2017.

The second one is to use the information observed in oracle model to build a Past Result Knowledge Base (PRKB). Unlike SanTable, PRKB is generated solely by the database server. It accumulates knowledge from observed "leaked" information (according to oracle model) of each query. As the server observes more queries, PRKB grows. PRKB can then provide input to query processing and prune certain processes and thus saves the overall query time. Details of the mechanism of PRKB can be found in our paper in EDBT 2018.

Objective 4 and objective 5:

Relevant algorithms were developed to achieve the objectives. Details of individual algorithms can be found in our published papers (DEXA 2017, EDBT 2018).

Objective 6:

The results of the performance study can be found in our published papers (DEXA 2017, EDBT 2018).

5.4 Summary of objectives addressed to date

Objectives *(as per 5.1/5.2 above)*	**Addressed** *(please tick)*	**Percentage Achieved** *(please estimate)*
1. Define a proper security definition that is suitable for the pruning scenario and desensitization.	✓	100%
2. Study how the security protection is affected when we apply our security definition on existing secure database systems.	✓	100%
3. Develop data desensitization technique to generate auxiliary data, that is secure under our defined security definition.	✓	100%
4. Develop query transformation technique to transform a plain query to a query on desensitized data.	✓	100%
5. Develop efficient algorithms to process a query on desensitized data.	✓	100%
6. Implement the developed techniques on an existing secure database system and carry out performance study.	✓	100%

6. Research Outcome

6.1 Major findings and research outcome

(Maximum 1 page; please make reference to Part C where necessary)

The query processing cost of secure database system because data is encrypted. It requires complex techniques to compute query on encrypted data. This project aimed to develop optimization method to improve the efficiency. We developed two major techniques:

1. Sanitized Table (SanTable) – DEXA 2017

Summary of the idea:
The user simulates the scenario where the server computes a fixed number of queries and sees the corresponding (encrypted) results. (The plain records are not observable to the server, but the server knows which encrypted record is in the result). The server can then make a rough estimation about the plain values. The user sends such estimations to the server as "sanitized data". This forms the sanitized table (SanTable). SanTable, containing estimation of plain values, can be used in query processing. For example, if WHERE clause of the query is "WHERE X < 10" and it is estimated that a data item falls in 50-500. The data item must not be in the query result without processing its corresponding encrypted data. Query processing cost can be saved.

Performance:
We measured security and query processing speed in our experiments. If an attacker uses inference attack to recover the plain data, the attacker accuracy for SanTable is at most 10% in our experiments. The attack accuracy of is more or less the same as the case the attacker attacks the database system without implementing SanTable.
SanTable has an average improvement of at least an order of magnitude over the secure cloud database system without implementing SanTable.

2. Past Result Knowledge Base (PRKB) – EDBT 2018

Summary of the idea:
Unlike SanTable, PRKB does not require user's involvement. The server accumulates knowledge from computing queries and uses such knowledge to facilitate query processing of new queries. For example, if the server knows that (i) t_1 and t_2 (but not t_3) are in the query result of query Q1; (ii) t_2 and t_3 (but not t_1) are in the query result of Q2; and (iii) t_1 and t_3 are in the result of Q3 and t_2 is not processed yet. It is further known that Q1, Q2 and Q3 are all queries with single comparison condition, i.e., the condition is in the form of "X op c" where X is the attribute in the table, op is a comparison operator (>, <, >=, <=) and c is a numeric value unknown to the server. Although the server does not see the plain values of t_1, t_2, t_3 and the plain query, the server can conclude that t_1 and t_3 are the largest and smallest value, or vice versa. If both t_1 and t_3 are in the result of Q3, so is t_2. As a result, the server can simply add t_2 to the Q3's result without processing t_2's encrypted data.

Performance:
As the server accumulates more query results in PRKB, the efficiency improves. At the point when the server has observed only 250 query results, query time of PRKB is about two orders of magnitude faster than the case PRKB is not implemented in the system.

6.2 Potential for further development of the research and the proposed course of action *(Maximum half a page)*

Our optimization is designed for selection processing. There are other different database operations in standard query processing, e.g., join, aggregate query. These are also important components in a database system. Due to limited time and resources, such operations are beyond the scope of this project. In addition, there are also more advanced but trendy applications on database systems, e.g., deep learning. The same efficiency challenge is faced, because these operations/applications also need to process a large amount of encrypted data. Our developed techniques are designed for selection processing. It is not obvious to modify the techniques and apply them in these operations. A potential extension of the project is then to study optimization methods for these applications/database operations.

On the other hand, we observed that there were very few systematic security analyses done on secure cloud database systems. Existing analyses were done on separate techniques, but not on entire systems. As a result, it is not clear whether some of the security settings make sense in some scenarios. A thorough study on security strength on existing secure database systems can help the research community understand better what security level each of the existing database system can achieve, e.g., what information is leaked in each system.

7. Layman's Summary

(Describe in layman's language the nature, significance and value of the research project, in no more than 200 words)

Database-as-a-service is an emerging data management model in which the database is hosted and managed by a third party (cloud) service provider. Security is a concern as the cloud server has access to user's private data. Encryption is required to protect the data, and so secure encrypted database systems were developed. Since data is encrypted, complex algorithms were developed to facilitate query processing on encrypted data. The query processing speed is slow and the gain from using cloud database diminishes significantly after encryption is applied. This project aimed to optimize the query processing speed of selection operations in a secure database system. We developed two methods, namely SanTable and Past Result Knowledge Base (PRKB). SanTable is initiated by the user and PRKB is solely done by the cloud server. Both techniques build an auxiliary structure on top of the system and uses it to prune unnecessary processing of encrypted data. Experiments showed that both methods were very effective, with at least an order of magnitude faster than the case our methods were not used.

Part C: Research Output

8. Peer-Reviewed Journal Publication(s) Arising Directly From This Research Project

(Please attach a copy of the publication and/or the letter of acceptance if not yet submitted in the previous progress report(s). All listed publications must acknowledge RGC's funding support by quoting the specific grant reference.)

The Latest Status of Publications				**Author(s)** *(denote the corresponding author with an asterisk**)*	**Title and Journal / Book** *(with the volume, pages and other necessary publishing details specified)*	**Submitted to RGC** *(indicate the year ending of the relevant progress report)*	**Attached to this Report** *(Yes or No)*	**Acknowledged the Support of RGC** *(Yes or No)*	**Accessible from the institutional repository** *(Yes or No)*
Year of Publication	**Year of Acceptance** *(For paper accepted but not yet published)*	**Under Review**	**Under Preparation** *(optional)*						
2015	N/A	N/A	N/A	Zhian He*, Wai Kit Wong*, Ben Kao, David Wai Lok Cheung, Rongbin Li, Siu Ming Yiu, Eric Lo	"SDB: A Secure Query Processing System with Data Interoperability", Proceedings of the VLDB Endowment, Vol. 8, No. 12	Yes, 2015	Yes	Yes	Yes

9. Recognized International Conference(s) In Which Paper(s) Related To This Research Project Was / Were Delivered

(Please attach a copy of each conference abstract)

Month / Year / Place	**Title**	**Conference Name**	**Submitted to RGC** *(indicate the year ending of the relevant progress report)*	**Attached to this Report** *(Yes or No)*	**Acknowledged the Support of RGC** *(Yes or No)*	**Accessible from the institutional repository** *(Yes or No)*
2017	Non-order-preserving index for Encrypted Database Management System	International Conference on Database and Expert Systems Applications (DEXA)	Not yet	Yes	Yes	Yes

2018	Optimizing Selection Processing for Encrypted Database using Past Result Knowledge Base	International Conference on Extending Database Technology (EDBT)	Not yet	Yes	Yes	Yes

10. Whether Research Experience And New Knowledge Has Been Transferred / Has Contributed To Teaching And Learning

(Please elaborate)

The research experience enabled the team to learn and generate new knowledge in the field of secure database system. Such knowledge can be transferred to students in future lectures so that students can know the latest development of new technologies in secure database systems.

Besides, the experience in research methodology was shared with students in some lectures to let them understand the importance of formal security analysis. Security is not a simple yes/no question and we should define clearly the security model and/or information leakage of the algorithm(s)/system(s).

11. Student(s) Trained

(Please attach a copy of the title page of the thesis)

Name	**Degree Registered for**	**Date of Registration**	**Date of Thesis Submission / Graduation**
N/A			

12. Other Impact

(e.g. award of patents or prizes, collaboration with other research institutions, technology transfer, teaching enhancement, etc.)

An EDBMS was implemented to serve as the test bed for the developed algorithms. The source code of the system is open-source for the general public. The source codes can be found in https://github.com/andyhehk/SecureDB.

13. Public Access Of Completion Report

(Please specify the information, if any, that cannot be provided for public access and give the reasons.)

Information that Cannot Be Provided for Public Access	**Reasons**
N/A	

RESEARCH GRANTS COUNCIL
COMPETITIVE RESEARCH FUNDING SCHEMES FOR THE LOCAL SELF-FINANCING DEGREE SECTOR

FACULTY DEVELOPMENT SCHEME (FDS)

Completion Report - Attachment

(for completed projects only)

RGC Ref. No.: UGC/FDS14/E05/14

Principal Investigator: Wong Wai Kit

Project Title: Improving selection query processing speed of secure cloud database systems by tuple pruning on desensitized data

Statistics on Research Outputs

	Peer-reviewed Journal Publications	Conference Papers	Scholarly Books, Monographs and Chapters	Patents Awarded	Other Research Outputs (Please specify)
No. of outputs arising directly from this research project [or conference]	1	2	0	0	Open source system prototype[1]

[1] https://github.com/andyhehk/SecureDB