DATE :　　2016/01/08

Department of Electric and Electronic Information Engineering		ID	M145204
Name	Ryo Yamauchi		

Supervisor	Shuichi Ichikawa Naoki Fujieda

Abstract

Title	A Proposal of Bandwidth Reduction Methods on Path ORAM

Oblivious RAM（ORAM）is a technique that hides memory access patterns by dummy accesses and data relocations. ORAM incurs a large bandwidth overhead that should be reduced for practical applications.

This study proposes the bandwidth reduction methods on Path ORAM, one of the lightweight ORAM algorithms. First, a Path ORAM simulator was developed and verified. Then, the storage cost and the bandwidth reduction of the proposed methods were evaluated.

In Path ORAM, data blocks are placed on external memory as a binary tree. Nodes from a leaf to the root are called *Path* and each block is assigned into a Path. All blocks along a Path are required to be read and written for each memory request from the CPU. Therefore the bandwidth overhead of Path ORAM is twice as many as the number of resident blocks on a Path. Furthermore, internal memory is additionally required to store blocks temporarily.

This study focuses on redundant memory accesses on Path ORAM to reduce its bandwidth. If successive memory requests are close in Path, some blocks may often be read immediately after being written. This study proposes two methods, called *Reuse* and *Delay*, to reduce such redundant memory access. Furthermore, I propose another method called *Shuffle* that improves the efficiency of bandwidth reduction for the both methods. When a Path is read in Reuse, the blocks contained in the previously written Path are reutilized by setting a valid flag in the internal memory. In Delay, writing back of blocks is postponed to the next Path access. The redundant accesses are reduced because the blocks read in the last Path access are present in the internal memory. In the both methods, the number of memory accesses can be reduced if two consecutive Paths are in close. Shuffle is a technique to reorder memory requests from the CPU by the proximity of Paths for the further reduction of the bandwidth.

In this study, an evaluation environment was developed to simulate Path ORAM algorithm. To examine its validity, I compared the simulation results to those shown in the preceding studies. The result of the comparison showed that the simulator is sufficiently accurate for our study.

The proposed methods were evaluated with the memory trace files provided in a DRAM simulator called USIMM. The bandwidth reduction rate and storage costs were evaluated based on the case where the memory access has not been reduced.

According to my simulation results, the combination of Shuffle and Delay achieved the highest bandwidth reduction rates, which were 27.0% at a maximum and 24.1% on average. Given the same storage cost, it achieved 10.3% higher than an existing method proposed by Maas *et al.*