平成27年年度　卒業研究報告書概要

課程，学籍番号，氏名	課程: 電気・電子情報工学課程，学籍番号: B153257，氏名: 藤田 浩輝
工学分野名: 情報通信システム	指導教員名: 市川 周一，藤枝 直輝

題　　目: 和　Path ORAMにおけるバンド幅削減手法の実装

（英　Implementation of the reduction of bandwidth on Path ORAM　）

Abstract

Secure processor provides advanced security such as data encryption to prevent information leakage. Goldreich proposed Oblivious RAM (ORAM) that hides memory access pattern by dummy requests and data relocation. Path ORAM was recently proposed by Stefanov et al. as a lightweight ORAM protocol. To reduce the bandwidth overhead of Path ORAM, Fujieda et al proposed two methods called Reuse and Delay to remove the redundancy of Path ORAM , while they were only confirmed as effective by simulation. The purpose of this study is to implement these methods in hardware to verify their effectiveness. Evaluation measures are the number of LUTs and Flip Flops, along with the maximum operating frequency. According to evaluation results of Reuse, LUTs increased by 46%, Flip Flops increased by 21%. In Delay, LUTs increased by 25%, Flip Flops increased by 14%.The maximum frequency decreased by 2%. In the evaluation of bandwidth, about 12% of reduction was shown in read requests of the Reuse and both read and write requests of the Delay.

概　　要

情報漏洩を防ぐため，メモリの暗号化など高度なセキュリティを提供するプロセッサ（セキュアプロセッサ）の研究が進んでいる．Oblivious RAM (ORAM) は Goldreich (1987) が提案したセキュアプロセッサ技術の一つで，ダミーリクエストやデータの再配置によってメモリへのアクセスパターンを秘匿する．特に軽量な ORAM プロトコルとして Stefanov ら (2013) は Path ORAM を提案した．Path ORAM は ORAM Tree と呼ばれる高さ L+1 のバイナリツリーによってデータを保持する．Leaf から root までの経路を Path と呼び，アクセスはこの Path 単位で行われる．Path ORAM はメモリアクセスにおけるバンド幅オーバーヘッドが大きいため，藤枝ら (2016) はバンド幅オーバーヘッドを削減する手法として Reuse と Delay を提案した．これらの手法はシミュレーションにより有効性が確認されているが，ハードウェアによる実装，評価は未だ行われていない．そこで本研究では Path ORAM における Reuse と Delay のハードウェア実装を行った．研究室で開発された ORAM コントローラーを用い，それぞれの手法を再現し実装，評価を行った．オーバーヘッド削減のため，Path ORAM のアクセスの冗長性に着目する．連続した ORAM アクセスにおいて，書き戻した直後のデータを再び読み出すのは無駄である．これを解消するために，直前に書き戻したデータを記憶する領域を ORAM コントローラーに実装した．記録されたデータと Stash 内のデータを比較し，読み書きの必要性を判断することによって冗長性を削減する．Reuse は重複していると判断したデータを Stash 内で有効化することでデータの読み出し (Read リクエスト) の冗長性を削減する．Delay は書き戻し (Write リクエスト) を遅らせ，Stash 内に直前にアクセスしたデータを残すことにより，Read/Write 両方の冗長性を削減する．実装評価では，LUT 数，Flip Flop 数，最大動作周波数を評価項目とした．評価結果では，Reuse は LUT 数が 46%，Flip Flop 数が 21% 増加．Delay は LUT 数が 25%，Flip Flop 数が 14% の増加となった．最大動作周波数は 2% 悪化した．バンド幅削減率を測定した結果，Reuse は Read リクエストで約 12%，Delay は Read/Write リクエスト共に約 12% の削減が見られた．これは，ORAM Tree の上位 2 段のデータが削減されている事と等しい．連続したアクセスにおいて重複するデータの期待値は

$$\lim_{n\to\infty} \sum_{i=1}^{\infty} 2^{-(i-1)} \fallingdotseq 2$$

となるため，理論値とも一致する結果となった．

発表する際の課程を記入　電気・電子情報工学　課程　発表番号 70

（学籍が他課程所属の学生も発表する課程を記入すること）