4Y-6

IXP425 における暗号処理をオフロードする Web サーバのパフォーマンス計測

天野桂輔† 渥美裕太† 笠原竜大‡ 村上智祐‡ 齋藤孝道†

†明治大学 ‡明治大学大学院

1. はじめに

IPsec, SSL (Secure Socket Layer) などのセキュリティプロトコルでは，通信に高負荷な演算処理を伴うため，システム全体のパフォーマンスを低下させてしまう．

その解決策の 1 つとして，暗号処理に最適化されたモジュールを搭載したネットワークプロセッサ（以下，NP と呼ぶ）を利用し，暗号処理をオフロードする方法がある．

NP では，暗号処理を専用モジュール（以下，暗号モジュールと呼ぶ）で処理することにより，汎用コアの負荷を下げ，システム全体のパフォーマンスの向上が期待できる．

本論文では，Intel 社の NP である IXP425 [1] の暗号モジュールを利用し SSL-Web サーバにおける暗号処理を暗号モジュールにオフロードする実装を行い，そのパフォーマンスを計測し，その評価を行った．

2. 計測環境

2.1 IXP425

IXP425 は，主に，XScale アーキテクチャを基にした汎用プロセッサ と 2 つのパケット処理専用モジュールから構成されている．

このパケット処理専用モジュールのことを NPE (Network Processor Engine) と呼び，IXP425 では，その内 1 つの NPE に暗号モジュールを内蔵している．この暗号モジュールは，共通鍵暗号化方式の DES, 3DES と AES に対応しており，その利用モードとして，CBC (Cipher Block Chaining) と ECB (Electronic Code Book) が利用可能である．また，ハッシュアルゴリズムとして SHA-1 と MD5 に対応している．

計測環境の構築には，IPX4xx シリーズ向けのデバイスドライバ開発用の API である Access Library を用いている．OS は uClinux を利用．

2.2 SSL サーバ

本論文では，パフォーマンス計測を行う Web サーバとして，OpenSSL の s_server コマンドを用いた．

s_server コマンドにおいて，-engine オプションの指定により，ハードウェアのデバイスを使用可能である．今回，指定するハードウェアとしては，我々が導入した OCF 経由での IXP425 の暗号モジュールとした．

これにより，SSL サーバで暗号処理をする際に，その暗号処理をオフロードする．

2.3 OCF と OpenSSL

IXP425 の暗号モジュールで暗号処理を行うため，OpenSSL が提供する ENGINE API を利用する．ENGINE API は，暗号モジュール毎に用意されているライブラリを呼び出し，暗号モジュールへのオフロードを可能にする．

本論文では，IXP425 の暗号モジュールに対する透過的なアクセスを提供するため，様々なハードウェアアクセラレータ向けの API を搭載したミドルウェアである OCF (OpenBSD Cryptographic Framework) を改修し用いた [2]．

OpenSSL から OCF 経由で，暗号モジュールへのオフロードを実現した．SSL サーバ，OCF 及び暗号モジュールの関係を図 1 に示す．

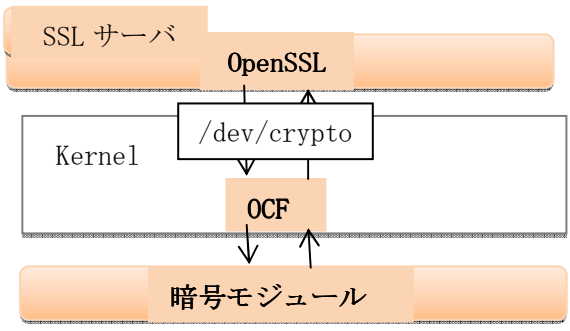

図 1:オフロード概要

A Performance Evaluation of Web server Off-loading Hardware Cryptographic Module on IXP425
† Amano Keisuke, Atsumi Yuta, Saito Takamichi
‡ Kasahara Ryuta, Murakami Tomosuke
Meiji University(†), Graduate School of Meiji University(‡)
1-1-1 Higashimita, Tama-ku, Kawasaki-shi, Kanagawa, 214-8571, Japan(†)(‡), ee77036@cs.meiji.ac.jp, ee77093@cs.meiji.ac.jp, ce06013@meiji.ac.jp, ce06039@jcom.ac.jp, saito@cs.meiji.ac.jp

論文 [2]を，s_server コマンド用に改変したものを利用した．

3. 評価

3.1 評価方法

本論文では，IXP425 上に構築した SSL サーバに対して，過負荷をかけることでパフォーマンスの評価を行う．負荷として HTTPS リクエストを SSL サーバに送る．ここで，HTTPS リクエストを生成する負荷生成器として，Spirent 社の Avalanche2007 モデル B（以下，Avalanche と呼ぶ）を用いた．Avalanche により多数のユーザをエミュレートし，各ユーザが HTTPS リクエストをそれぞれ送信するかのような負荷を生成する．毎秒のリクエスト数を徐々に増やしていくことにより，SSL サーバの正常に処理可能なリクエスト数を計測した．

計測では，負荷生成開始から 60 秒間，一定の割合でリクエスト数を増やしていき，最終的に毎秒 50 個のリクエストを生成するようにした．各リクエストでは，共通鍵暗号化方式に 3DES，その利用モードに CBC，ハッシュアルゴリズムとして SHA-1 を用いた．また，Avalanche からリクエストとして，SSL サーバへ送信するファイルサイズは，平均的な Web ページのサイズである 50Kbytes [3]と 1Kbyes とした．

最終的に，対象の SSL サーバが，正常に処理可能なリクエスト数を調べ，オフロードを行わない状態の SSL サーバとの比較を行う．

3.2 計測結果

SSL サーバが実際に処理したリクエストの数を表 1，オフロードを行わない場合と行った場合の SSL サーバのグラフを図 2 に示す．ただし，図 2 のページサイズは 50Kbytes のものである．

表 1:処理したリクエスト数

ファイルサイズ	50Kbytes		1Kbytes	
暗号モジュール	未使用	使用	未使用	使用
Attempted	1513	1510	1511	1510
Successful	843	1104	1406	1379
Unsuccessful	670	406	105	131

表 1 より，リクエスト全体の量を Attempted，SSL サーバが処理に成功したリクエスト数を Successful，処理に失敗したリクエスト数を Unsuccessful とした．

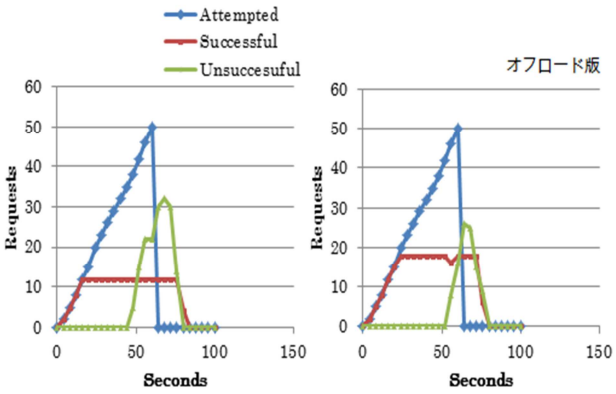

図 2:計測結果

図 2 より，本論文の条件下で計測を行うと，暗号モジュールにオフロードを行う場合（図 2 右）の SSL サーバのリクエスト処理数が，オフロードを行わない場合（図 2 左）と比べて，上回っていることがわかる．しかし，表 1 より，本論文の計測では，クライアントから SSL サーバへ送信するファイルのサイズが 1Kbytes の場合，オフロード時に発生するオーバーヘッドの関係で，暗号モジュールにオフロードを行う場合の SSL サーバのリクエスト処理数の方が下回る結果となった．

4. まとめ

本論文では，IXP425 における暗号処理をオフロードする Web サーバの実装と評価を行った．

計測の結果から，リクエストのファイルサイズが 50Kbytes の場合，IXP425 の暗号モジュールにオフロードを行うことで，SSL サーバがより多くのリクエストを処理できるようになったと言える．今後の課題としては，他の Web サーバでのパフォーマンスの計測，オフローディング方式の改善などが挙げられる．

謝辞 本研究の成果の一部は，科学研究費補助金（課題番号 22700086）の助成を受けたものである．

参考文献

[1] http://www.intel.com/design/network/products/npfamily/ixp425.htm.

[2] 齋藤，大釜，羅，杉浦，IXP425 における暗号処理の効率的なオフロード方式の実装と評価，情報処理学会論文誌，Vol.51 No.9 (2010)，pp1530-1541.

[3] L.Badia，Real World SSL Benchmarking，Rainbow Techologies Whitepaper，Sept. 2001.