

500Mbps の IPsec 処理を実現する組み込み Linux システムの実装と評価

福田 伸彦[†], 下見 淳一郎[†], 下國 治[†], 都筑 俊秀[‡], 河合 純[†], 陣崎 明[†]

† 株式会社富士通研究所 コピキタシステム研究センター

〒211-8588 川崎市中原区上小田中 4-1-1

E-mail: {nfukuda,shitami,osamus,jkawai,zinzin}@labs.fujitsu.com

‡ 株式会社富士通コンピュータテクノロジーズ

〒211-8588 川崎市中原区上小田中 4-1-1

E-mail: tsuzuki@ctec.fujitsu.com

Implementation and Evaluation of Embedded Linux System Realizing 500Mbps IPsec Processing

Nobuhiko Fukuda[†], Junichiro Shitami[†], Osamu Shimokuni[†], Toshihide Tsuzuki[‡],
Jun Kawai[†], Akira Jinzaki[†]

† Ubiquitous System Research Center, Fujitsu Laboratories Ltd.

211-8588 Kamikodanaka 4-1-1, Nakahara-ku, Kawasaki, Kanagawa, Japan

E-mail: {nfukuda,shitami,osamus,jkawai,zinzin}@labs.fujitsu.com

‡ Fujitsu Computer Technologies Ltd.

211-8588 Kamikodanaka 4-1-1, Nakahara-ku, Kawasaki, Kanagawa, Japan

E-mail: tsuzuki@ctec.fujitsu.com

小型・低消費電力でありながら、高速なセキュア通信処理が可能なプロセッサ Trusted Network Processor (TNP)を利用して、Gigabit Ethernet 上で高いIPsec ESP 処理性能を実現するシステムの開発を行った。TNP はプログラマブルなセキュア通信処理エンジン Stream Processor (SP)と ARM プロセッサコア、2 つの Gigabit Ethernet コントローラを搭載する。このTNP 上で、高い転送性能を実現する組み込みファームウェアと汎用機能を実現するLinux を両立して動作させることに成功した。この時のIPsec パケットの転送性能は最大で 591Mbps を達成した。また、必要なコア消費電力は 620mW と小さい。本稿では、TNP の概要とLinux のTNP 対応、SP によるIPsec ESP 処理性能の測定結果と評価を示す。

1. はじめに

今日、インターネットへのアクセスラインは光回線によるブロードバンド化が進みつつある。その高速性を利用し、各家庭へ高品質の動画を配信するサービスが提供され始めている。現在のブロードバンド回線の主流は 10 ~ 100Mbps 程度の帯域であるが、2005 年 7 月より 1Gbps のサービスが開始された[1]。今後、Gigabit クラスの帯域を持つ家庭向け回線が増えていくであろう。

また最近では、ネットワークに対応した AV 機器や家電などのホームアプライアンス機器が登場し始めている[2]。これらの機器によりホームネットワークが構成され、音声や動画などのコンテンツ配信や、エアコンや冷蔵庫などの制御がネットワークを通じて行われるようになる。

そのような状況において、ブロードバンド環境におけるセキュア通信への要求はますます高まっている[3]。家庭にいながらにして会社や学校の内部ネットワークに安全にアクセスできる VPN の利用をはじめ、著作権保護のためのコンテンツ配信、インターネットストレージサービスの利用などにおいて、通信の暗号化は欠かせない。このようなセキュア通信の実現には、セキュアゲートウェイ機器が利用される。

しかし、セキュア通信に必要な、データの暗号化や復号化といった処理は非常に重く、ブロードバンド環境で高いスループットのセキュア通信を行うためには、高性能なプロセッサを必要とする。高性能な汎用プロセッサは消費電力が数十 W と大きく、また数千 ~ 数万円程度と高価なため、小型で廉価なホームルータなどのゲートウェイ機器では利用しにくかった。

こうした問題に対応するため、セキュア処理をハードウェアで高速に行う機能を持つ組み込み向けのネットワークプロセッサが数多く登場している。しかし、パケット処理については、これまでのネットワークプロセッサでは制約が大きかった。なぜなら、パケット処理をハードウェアで行う場合、高いスループット性能を得られるが、プログラマビリティがなく、新しいプロトコルへの対応が困難である。一方、ソフトウェアで処理を行う場合、柔軟性は高いが、スループット性能を得られにくいという問題がある。このように、プログラマビリティとスループット性能の両立は困難であった。

そこで我々は Trusted Network Processor (TNP)[4]

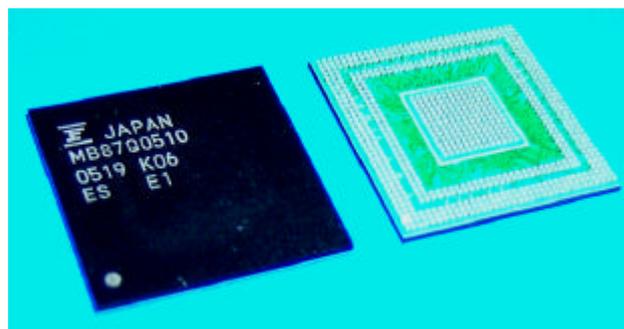


図 1: TNP チップ外観

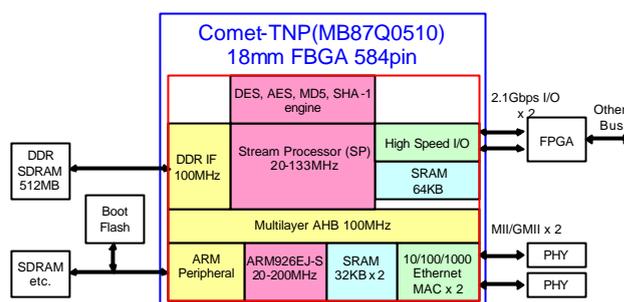


図 2: TNP ブロック図

を開発した。TNP はプログラマブルな Stream Processor (SP)[5,6] を搭載し、IPsec ESP (Encapsulating Security Payload) 処理をはじめ、さまざまなパケット処理を高速に実現できる。TNP は Gigabit Ethernet 上で 600Mbps 以上のスループットを実現する。この時のコア消費電力は 620mW である。また、高い転送性能を実現する組み込みファームウェアと、汎用機能を提供する組み込み Linux を共に動作させることで、性能と機能性の両立を目指した。

本稿では、TNP の概要と Linux の TNP 対応、SP による IPsec ESP 処理性能の測定と評価について述べる。TNP を用いることで、低消費電力で高いスループット性能を持ったセキュアゲートウェイ機器を実現可能である。

2. Comet TNP の概要

TNP はプログラマブルなセキュア通信処理エンジン SP と 2 つの Gigabit Ethernet コントローラ 制御用の ARM926EJ-S プロセッサおよび周辺デバイスをパッケージ化したネットワークプロセッサである。TNP のチップ外観写真を図 1 に、ブロック図を図 2 に示す。また、TNP の諸元を表 1 に示す。

表 1: TNP 諸元

パッケージ	FBGA 584pin, 18mm 角
消費電力(typ.)	2.3W
クロック周波数	20 ~ 200MHz
通信処理エンジン	Stream Processor (SP)
暗号エンジン	AES128/192/256, DES/3DES
ハッシュエンジン	MD5, SHA-1
制御プロセッサ	ARM926EJ-S (Cache I-32KB/D-32KB, MMU 有)
バッファ用メモリ インタフェース	DDR SDRAM 100MHz (最大 512MB, パリティ付き)
ネットワーク	Gigabit Ethernet MAC x2
周辺デバイス	UART, タイマ, 割り込み, DMA, 64KB SRAM, ETM9, JTAG

本節では、TNP の構成要素について説明する。

2.1 Stream Processor (SP)

SP はプログラマブルで高速なパケット処理が可能なセキュア通信処理エンジンである。暗号エンジンは AES128/192/256 (数字は鍵の bit 長を示す) および DES/3DES、ハッシュエンジンは MD5 および SHA-1 をハードウェアで搭載している。また、送受信パケットのバッファリング用に DDR SDRAM を最大 512MB まで接続できる。

図 3 に SP 処理のフローを示す。SP は入力パケットのデータを状態遷移表のインデックスとして扱う一種の有限状態機械であり、暗号エンジンやハッシュエンジンを SP プログラムから並列利用することができる。SP プログラムを開発することで、IPsec ESP 処理をはじめ、TLS/SSL (Transport Layer Security/Secure Socket Layer) アクセラレータ iSCSI (SCSI over IP) によるストレージサービス、そして DTCP-IP (Digital Transmission Content Protection over IP)[7] によるコンテンツ保護といった、多様な機能を実現できる。

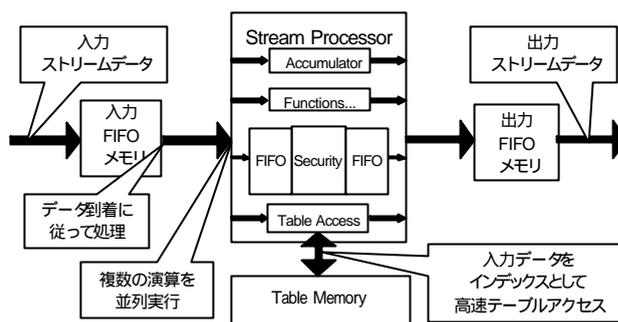


図 3: SP 処理フロー

2.2 Gigabit Ethernet MAC

TNP は Gigabit Ethernet MAC (Media Access Control) コアを 2 つ搭載している。PHY チップは外部に接続する。フレームデータの送受信は、メモリ上に制御情報を記述したディスクリプタを用意し、DMA 転送で送受信を行うため、プロセッサに負荷をかけずに処理できる。

2.3 ARM コア

TNP は Gigabit Ethernet MAC や UART といった周辺デバイスの制御プロセッサとして、ARM926EJ-S コアを搭載している。クロック周波数は最大 200MHz である。メモリは高速アクセスが可能な 64KB のオンチップ SRAM に加え、システム用に SDRAM を最大 128MB まで接続できる。この ARM コア上で Linux や NetBSD、ITRON といった組み込み用 OS を動作させることも可能である。

2.4 Comet TNP 評価ボード

TNP の動作検証と評価のために、我々は Comet TNP 評価ボードを製作した。本稿の性能評価はこの評価ボード上で行った。評価ボードの写真を図 4 に、諸元を表 2 に示す。

2.5 搭載メモリ

評価ボードは以下の 3 種類の RAM を搭載している。

- オンチップ SRAM (64KB)
- SDRAM (32MB)
- DDR SDRAM (512MB)

ARM コアからそれぞれのメモリのワードアクセスにかかる時間を表 3 に示す。オンチップ SRAM のアクセスは最も速いが、容量が非常に小さい。DDR SDRAM は、バース



図 4: Comet TNP 評価ボード

表 2: Comet TNP 評価ボード諸元

プロセッサ	TNP (MB87Q0510)
システム用メモリ	SDRAM 32MB
バッファ用メモリ	DDR SDRAM 256MB
Flash ROM	8MB
Ethernet PHY	Marvell 88E1111 (2 ポート)
シリアル	UART (1 ポート)

トランスポート時は SDRAM より高速であるが、ワード単位のアクセスの繰り返しではあまり性能が出ないことを示している。このような性質の違いにより、パケット用バッファ、Gigabit Ethernet MAC のディスクリプタ、ARM 用ファームウェアの使用するバッファ管理データをどのメモリに配置するかによって、性能が変化する。

事前の実験より、アクセスはデータごとに異なる種類のメモリに分散させた方が良い性能が得られることがわかっている。そこで、原則としてメモリを以下のように利用することにした。

- オンチップ SRAM: Gigabit Ethernet MAC のディスクリプタ (非キャッシュ)
- SDRAM: バッファ管理データ (キャッシュ)
- DDR SDRAM: パケット用バッファ (非キャッシュ)

2.6. ソフトウェアモデル

TNP のソフトウェアは、全体の制御のための ARM ソフトウェアと、パケット処理を行う SP ソフトウェアで構成される。

TNP のソフトウェアモデルとして、高性能で実時間性が要求される用途に向けた組み込みソフトウェアモデルと、性能よりも汎用性や機能性を重視した Linux モデルがある。いずれのモデルにおいても、同じ SP プログラムを共通に使用できる。

表 3: メモリアクセス時間

メモリの種類	リード(ns)	ライト(ns)
オンチップ SRAM	60.00	25.00
SDRAM	120.57	70.72
DDR SDRAM	250.24	122.62

CPU キャッシュはオフ

3. TNP Linux

当初開発を行った TNP ソフトウェアは、チップの評価検証の目的もあり、専用の組み込みファームウェアとして実装した。この組み込みファームウェアは、入力パケットに対して IPsec ESP 処理を行い、出力するというシンプルな機能を持つものである。

シンプルな組み込みファームウェアは、性能の高いものを作りやすい。その反面、一般的なホーム Appliances 機器などが備える、Web や TELNET による設定機能といった汎用的なネットワークサービスの組み込みは、実装のコストが大きい。また、実用的な IPsec 利用のためには、IKE (Internet Key Exchange) 処理の組み込みも別途必要になる。

そこで、既存のアプリケーションやサービスを容易に利用できるよう、TNP で動作する組み込み用の Linux 環境の構築を行った。Linux の実行によるオーバーヘッドをできるだけなくし、高速な IPsec ESP 処理を実現するために以下のアプローチを採った。

- 専用の組み込みファームウェアを最小限の変更で Linux のドライバモジュールに移植
- ファームウェアをできるだけ実行させつつ、Linux の実行時間を確保

本節では、Linux カーネルの TNP 対応と実験用ユーザ環境の構築、そしてファームウェアの Linux ドライバ化とその動作について説明する。

3.1 Linux カーネルの TNP 対応

Linux カーネルの TNP 対応は、kernel.org[8]で配布している標準のカーネルソースをベースに行った。使用したカーネルのバージョンは 2.4.31 および 2.6.12.3 である。

これらのバージョンの Linux は標準で ARM プロセッサ

に対応しており TNP の周辺デバイス (割り込みコントローラやタイマ) に依存する部分の処理を実装するだけで動作させることができた。追加または変更したファイルの数は 25 個程度である。

シリアルコンソールは TNP 内蔵の UART を使用する。UART のドライバは、Linux のシリアルドライバをそのまま使用できた。

カーネルは評価ボードの Flash ROM (8MB) 上に載せるため、実験評価に必要な最小限の構成で構築した。基本機能と IPv4 スタックのみのカーネル(無圧縮)のファイルサイズは、2.4.31 で約 1.3MB、2.6.12.3 では約 1.94MB である。また、圧縮時(zImage)はそれぞれ約 490KB、約 723KB であった。

3.2 実験用ユーザ環境

実験評価用のユーザ環境には BusyBox[9]を使用した。BusyBox は組み込み向けのコマンドツール群で、コンパクトなサイズながら数多くのコマンドを実装している。

ファイルシステムは 4MB の RAM ディスクとした。評価ボードの Flash ROM に初期 RAM ディスクイメージ (initrd) を書き込み、ルートファイルシステムとして使用できるようにした。

3.3 組み込みファームウェアの Linux ドライバ化

組み込みファームウェアでは、割り込みを使わずにポーリングで処理を行っている。そこで、TNP の持つ 2 個のタイマデバイスの 1 つを使用して、割り込みによって定期的にファームウェアを起動するようにした。

Linux 上でのファームウェアの実行の概念を図 5 に示す。ファームウェアがタイマ割り込みによって実行を開始すると、Gigabit Ethernet MAC によるパケットの送受信処理、SP による処理データの DMA 転送が行われる。1 回のファームウェアの実行でパケットの送受信処理がインタフェースごとにそれぞれ最大 16 回、SP への入力 DMA が最大 16 回、SP からの出力 DMA が最大 32 回行われる。

それぞれの処理でこれ以上行いう処理がないか、最大回数の処理が行われると、次回のファームウェア起動

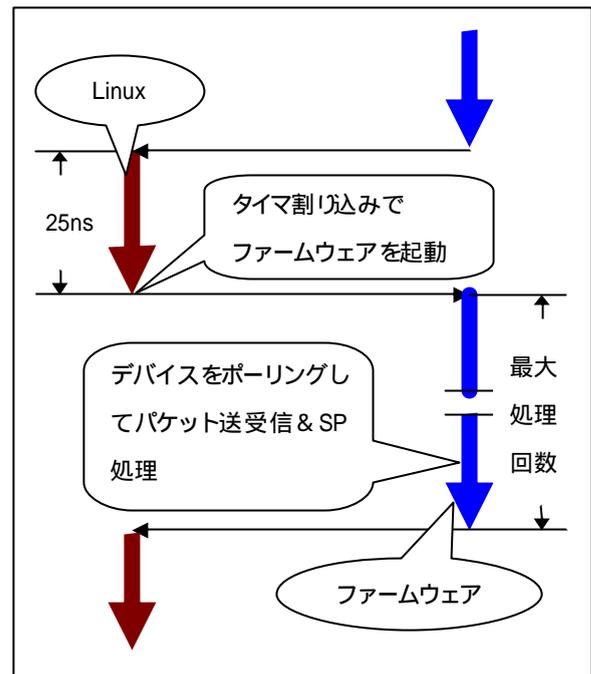


図 5: Linux 上でのファームウェアの実行

のためのタイマ割り込みを設定して割り込みを終了する。Linux カーネル 2.6.13.2 の場合、Linux の処理時間を最低 25ns 分確保しないと Linux が正常に動作しなかったため、この値が設定可能な処理時間の最小値となる。

3.4 Linux からのパケット送受信

ファームウェアで行っているパケット送受信処理や IPsec ESP 処理は、Linux カーネルのネットワークスタックを一切使用していない。そのため、そのままでは TNP の Gigabit Ethernet を経由して、Linux 上で動作するネットワークサービスにアクセスすることができない。

そこで、Linux から送受信したいパケットをファームウェアとの間で受け渡すモジュールを用意した。それにより、TNP の 2 つの Gigabit Ethernet インタフェースを、Linux 上から通常のネットワークインタフェースとして利用できる。

ファームウェア上では、パケット受信時に入力パケットの宛先を調べて、Linux が受信すべき IP アドレスであったら Linux のネットワークバッファに DMA 転送を行う。また、Linux からの送信時も同様にファームウェア

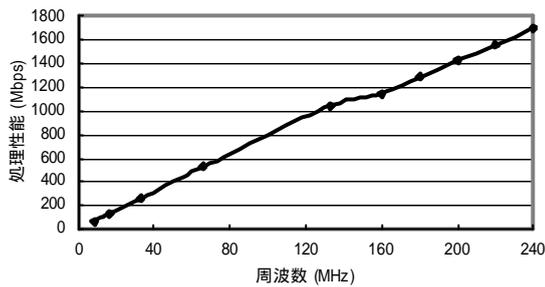


図 6: SP の IPsec ESP 処理性能

ムウェアが DMA 転送で出力パケットを受け取り、送信を行う。

Linux カーネルはファームウェア上のパケット用バッファを直接管理できないため、Linux のシステム用メモリとの間でデータ転送のオーバーヘッドが生じる。オーバーヘッドが生じる。しかし、本機能は設定・管理用の Web サーバや TELNET サーバなど、低スループット用途での利用を想定しており、性能は求めない。

4. 性能評価

評価ボードを用いて、TNP による IPv4 のパケット処理性能を測定した。スループットの測定には Agilent Router Tester[10]を使用した。

本節では、専用の組み込みファームウェアを使用した場合と Linux ドライバ化したファームウェアを使用した場合について、それぞれ結果を示す。

4.1 SP の IPsec ESP 処理性能

SP 単体での IPsec ESP 処理性能を図 6 に示す。暗号化アルゴリズムは AES128、認証アルゴリズムは HMAC-MD5 を使用した。SP コアのクロック周波数を変化させながら測定したところ、133MHz 動作時に 1Gbps のスループットが得られた。

4.2 組み込みファームウェアによる転送性能

専用の組み込みファームウェアを使用して、単方向のパケット転送時におけるスループットの測定を行った。ARM コアのクロック周波数は 200MHz、SP のクロック周波数は 133MHz である。

測定項目を以下に示す。

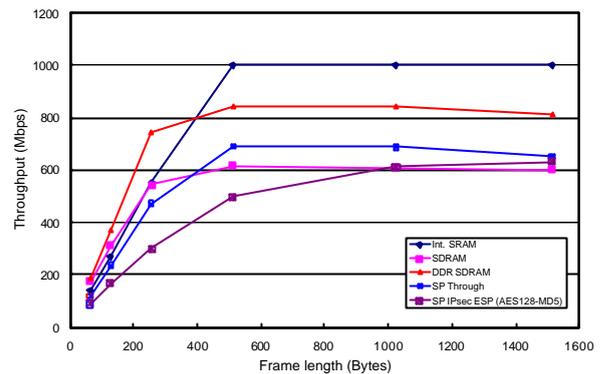


図 7: 組み込みファームウェアによるパケット転送性能

- SP を経由しないスルー転送
 - オンチップ SRAM 上のバッファ
 - SDRAM 上のバッファ
 - DDR SDRAM 上のバッファ
- SP を経由したスルー転送
- SP を経由した IPsec ESP 転送

これらの測定結果を図 7 に示す。以降、それぞれの詳細について説明する。

4.2.1 SP を経由しないスルー転送

まず、TNP の Gigabit Ethernet コントローラの性能を調べるため SP を経由しないでスルー転送を行った。この時、バッファに使用するメモリの種類を変えてスループットの測定を行った。Gigabit Ethernet MAC によるパケットの転送は、指定のメモリに対する DMA 転送によって行われるため、ARM コアがバッファ用メモリにアクセスする必要はない。

オンチップ SRAM の場合では、フレーム長が 512B 以上の時にワイヤスピードの 1Gbps が得られた。しかし、ショートパケット時の性能低下が著しかった。これは、64KB しかないオンチップ SRAM ではバッファの個数が 31 個(1 バッファあたり 2KB、残りの領域はディスクリプトとして使用)と非常に少なく、MAC からの転送時に空きバッファが枯渇してしまうためと考えられる。

SDRAM および DDR SDRAM の場合では、ほぼメモリの性能に比例する結果となった。DMA 転送はバースト的に行われるため、DDR SDRAM においてピーク性能が 845Mbps と高い性能が得られた。

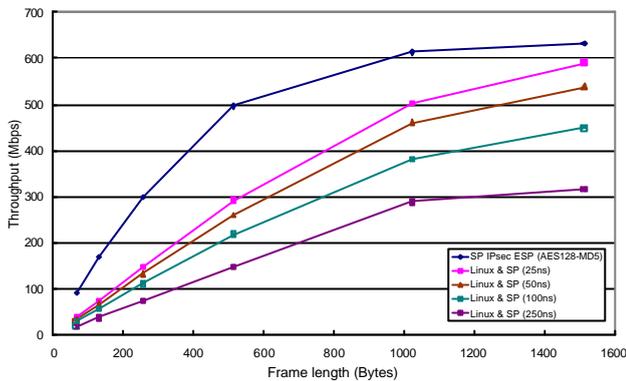


図 8: Linux 用ファームウェアによる
パケット転送性能

4.2.2 SP を経由したスルー転送

次に、パケットを SP に通じた場合のスループットを測定した。この測定では受信したパケットを SP に入力し、なにも処理せずにそのまま出力する。SP の入出力は DDR SDRAM に対する DMA 転送で行われる。

測定結果は、ピーク性能が 690Mbps、ショートパケット時の性能が 118Mbps であった。SP を経由した場合の若干の性能低下は、SP と DDR SDRAM 間の DMA 転送によるオーバーヘッドが生じるためと考えられる。

4.2.3 SP を経由した IPsec ESP 転送

SP で IPsec ESP 処理(トンネルモードを行った場合)のスループットを測定した。暗号化アルゴリズムは AES128、認証アルゴリズムは HMAC-MD5 を使用した。この測定では、2 台の評価ボードをチェーン接続し、1 つのパケットに対して暗号化と復号化の処理を連続して行っている。

測定結果はピーク性能が 631Mbps、ショートパケット時の性能が 89Mbps であった。フレーム長 64B 時の、毎秒あたりの転送パケット数は 17 万 pps 程度である。

Intel Xeon 2.4GHz 上で動作する Linux カーネル 2.4.20 および FreeS/Wan での IPsec ESP 処理性能の測定[5]では、100Mbps を下回る程度の性能であった。このような汎用プロセッサによるソフトウェア処理に比べると、TNP は小型で低消費電力であるが、非常に高い性能が得られることがわかった。

なお、Gigabit Ethernet MAC はジャンボフレームに対

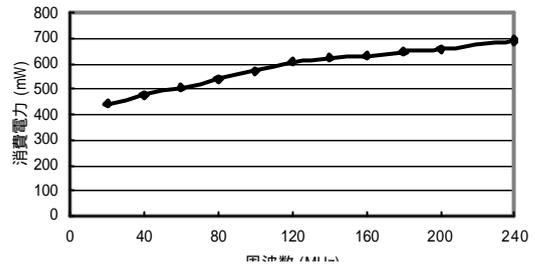


図 9: TNP コア消費電力

応しており、ソフトウェア対応によりフレーム長を大きくすれば最大性能はより上がると考えられる。

4.3 Linux 用ファームウェアによる転送性能

Linux kernel 2.6.12.3 上で、Linux ドライバ化したファームウェアを使用した場合の IPsec ESP パケット転送性能を測定した。SP プログラムで使用した暗号化アルゴリズムは AES128、認証アルゴリズムは HMAC-MD5 である。

測定結果を図 8 に示す。比較のため、組み込みファームウェアでの測定結果を併記した。Linux の実行単位時間が 25ns の時、スループットはピーク性能で 591Mbps、ショートパケット時で 37.4Mbps という結果になった。組み込みファームウェアと比較すると、フレーム長が大きい場合、性能低下は 10% 程度であるが、フレーム長が小さくなると 50~60% 程度まで性能が低下した。

また、Linux の実行単位時間を増加させると、25ns の場合に対して 25ns ごとに約 10% ずつスループットが低下した。Linux の実行時間単位は、Linux 上で動作させたいサービスの種類によって決定する必要があるが、25ns の時でファームウェアによる転送が高負荷の状態であっても、シリアルコンソールによるシェル操作の応答性は得られた。

4.4 消費電力

SP のクロック周波数を変化させながら IPsec ESP パケット転送を行った時のコア(1.2V)の消費電力を測定した。結果を図 9 に示す。133MHz 動作時の消費電力は 620mW 程度である。20MHz あたりの処理効率はスループットが 100Mbps、消費電力の増加が 35mW である。

4.5 相互接続性試験

TNP と他の IPsec 対応機器との相互接続性の確認を行った。調査環境は Linux 用の IPsec 実装である Openswan[11] 2.3.1 を用意した。

IPsec ESP 処理の暗号化アルゴリズムに AES128 と 3DES、認証アルゴリズムに HMAC-MD5 と HMAC-SHA-1 をそれぞれ使用して、トンネルモードで通信を行ったところ、問題なく通信を行うことができた。

4.6 他 OS への適用

Linux 用ファームウェアはパケットの受信、IPsec ESP 処理、送信の一連の作業を、基本的に OS の機能を使わずに行っている。また、その実行は割り込みコンテキスト内で行われる。

OS の機能に依存する部分は、メモリ空間や割り込み資源のアロケーション、設定のためのインタフェース、OS 側ネットワークへの橋渡し処理である。これらの機能は、汎用的な OS が備えるカーネル内プログラムインタフェースを使用することで実現できるため、他 OS への適用は容易である。

ファームウェアの実行性能は OS の実行単位時間に依存する。必要な実行単位時間は OS の構造に依存すると考えられるため、個別に調整する必要がある。

5. おわりに

TNP は小型・低消費電力でありながら、柔軟なプログラミングが可能なセキュア通信処理エンジン SP によって、高速なパケット暗号化処理が可能なネットワークプロセッサである。

高性能な専用の組み込みファームウェアによる IPsec ESP 処理性能測定では、SP が 133MHz 動作時に 620mW というコア消費電力で最大 631Mbps という高いスループット性能が得られることがわかった。また、汎用的なネットワークサービスを実現するために、Linux の TNP 対応を行った。組み込みファームウェアの Linux ドライバ化を行い、IPsec ESP 処理性能を測定したところ、最大 591Mbps のスループット性

能が得られた。

最近、Gigabit Ethernet コントローラとセキュア通信処理エンジンを搭載したネットワークプロセッサが登場し始めている[12]。今後、ブロードバンド環境が広く普及するにつれて、高速で安全な通信を実現できる TNP の需要はますます高まっていくであろう TNP は SP プログラムによって、TLS/SSL や DTCP-IP、iSCSI といったさまざまなプロトコルに柔軟に対応することができる。

今後は、TNP を搭載した実用評価機器として開発した Comet Gate を用いて、Comet TCP[13]や Comet DVIP/SXGA[14]といった既存アプリケーションの移植、TNP を利用した新しいアプリケーションの開発、そして TNP 搭載機器の製品展開を目指す。

謝辞

TNP の開発では、SP プログラムおよびその開発ツールを担当した富士通関西中部 ネットテック株式会社の田中淳介氏をはじめ、関連部署の諸氏に多大なるご協力をいただきました。ここに感謝の意を表します。

本研究の一部は新エネルギー・産業技術総合開発機構 (NEDO) 基盤技術研究促進事業委託研究 02004216-0 「トラステッドネットワークプロセッサ基盤技術の研究開発」によって行った。

参考文献

- [1] <http://www.k-opti.com/press/2005/press27.html>
- [2] http://www.dhf.gr.jp/JP/committee/contents/DHF_White_Paper040715.pdf
- [3] 山田勇、長谷部高行、足利靖: クライアントセキュリティ技術, 雑誌 FUJITSU 2004-1 月号, Vol.55, No.1, pp.62-67, 2004.
- [4] 陣崎明、都筑俊秀、鈴木英好: Gigabit ネットワーク向け低消費電力セキュリティ LSI の開発, 組込技術とネットワークに関するワークショップ ETNET2005, Mar. 2005.
- [5] 下國治、河合純、陣崎明、山澤昌夫、中村修、村井純: Security Network Processor による低消費電力 IPsec ESP の実装と評価, インターネットコンファレンス 2003 論文集, pp.51-58, 2003.

- [6] 陣崎明: Stream Processor、並列処理シンポジウム
JSPP2000, IPSJ Symposium Series Vol. 2000,
No.6, pp. 205-212, 2000.
- [7] <http://www.dtcp.com/>
- [8] <http://www.kernel.org/>
- [9] <http://www.busybox.net/>
- [10] <http://advanced.comms.agilent.com/n2x/>
- [11] <http://www.openswan.org/>
- [12] http://japan.renesas.com/media/company_info/news_and_events/press_releases/2005/0801/20050801j.pdf
- [13] 下見淳一郎、河合純、下國治、陣崎明、中村誠、稲葉真理、平木敬: 長距離 TCP 高速化機構の開発、インターネットコンファレンス2004 論文集, pp.83-91, 2004.
- [14] <http://www.comet-can.jp/>