組み込み機器における最適な IPsec 実装方式の検討

海老名 明弘1 永井 靖1 水谷 美加1
1日立製作所 システム開発研究所

1 はじめに

ADSL や FTTH のサービス展開に後押しされ，家庭のブロードバンド化が進み，ネットワークへの定額常時接続環境が整いつつある．インターネット通信では，通信内容の傍受，データ改ざんの危険性があり，今後様々な機器(ノン PC 端末)がインターネットに接続されることを考慮すると IPsec を標準実装する IPv6 の利用が必然となる．一方，ノン PC 端末に実装される組み込み系プロセッサは処理能力が低く，IPsec 通信による認証・暗号処理に多くの処理時間を費やし，スループットが 6 分の１に低下するといわれている(1)．本稿では，認証・暗号処理を軽減するハードウェアを OS 内部の通信処理部に組み込む際の課題と実装方式について述べる．

2 認証・暗号ハードウェアの概要

認証・暗号演算を行うハードウェア処理をする場合，(ⅰ)メインメモリ上の暗号データのハードウェアへの転送，(ⅱ)暗号データの復号処理，(ⅲ)復号データのハードウェアからメインメモリへの転送，という一連の手順で暗号/復号データ処理することができる．この際データ転送は，メインメモリと暗号ハード間で２度行われることから，効率よくハードウェアにデータを転送することが必要となる．そこで，暗号ハードウェアには(１) CPU が逐次データコピーを行う PIO(Program I/O transfer)方式に加えて(２) CPU を介さず DMA(Direct Memory Access)コントローラ(DMAC)がデータコピーを行う DMA 方式対応する I/F を備えた．このハードウェア適用時のシステム構成図を図 1に示す．暗号ハードは，IPsec 通信で規格化されている認証・暗号アルゴリズムである DES/3DES, SHA1, MD5 の認証・暗号演算部，DMA 転送時のデータを格納する２面バッファ，暗号ハード内部での DMA 転送を司る内蔵 DMAC で構成した．以下，PIO 方式と DMA 方式におけるデータ復号化の流れを示す．

(1) PIO 方式でのデータ処理

PIO 方式では，(a)CPU がメインメモリ上にある 4byte 分の暗号データを読み込み，(b)CPU が暗号データを認証・暗号演算部のレジスタへ書き込み，演算結果の復号データを読み取り，(c)CPU が復号データをメインメモリ上に書き込む処理を行う.(a)から(c)の流れをデータ長/4byte(回)行うことで暗号データ全てを復号することができる．

(2) DMA 方式でのデータ処理

DMA 方式では，(d)データ転送を行う前に CPU キャッシュ上のデータをメインメモリ上へ展開し，DMAC,内部 DMAC の設定を行う，(e)メインメモリ上のデータをバッファ 1 へ転送する，(f)内蔵 DMAC がバッファ 1 上にある暗号データの復号処理を行いバッファ 2 に格納する，(g)バッファ 2 上にあるデータをメインメモリ上に転送することで暗号データの復号化処理が完了する．DMA 転送は 16byte のブロックでデータ転送するため PIO 方式に比べてより高速なデータ転送が実現できる．また，復号後のデータのメインメモリへの転送無しに続けて認証の処理を行えること，さらに(f)の動作と(e)の動作を並列処理することを可能とし，復号処理時のオーバヘッドを隠蔽することで高速化できる特徴を持つ．

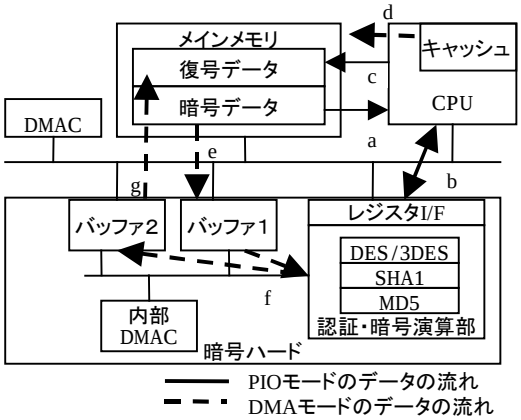

図 1 システム構成とデータの流れ

3 ハード制御用ソフトウェアの実装

3.1 実装時の課題

ソフトウェアで処理する通信プロトコル部にハードウェアを適用するためには，以下を考慮する必要がある．

(1) 既存システムへの容易な組み込み

通信プロトコル処理は，一般に OS の一部に組み込まれており，多くのモジュール部から構成される．暗号モジュール部を制御ソフトウェアに容易に置き換えるために，既存関数の活用と，データの受け渡し手順

Proposing IPsec Implementation for an embedded system
Akihiro Ebina[1] Yasushi Nagai[1] Mika Mizutani[1]
[1]Systems Development Laboratory, Hitachi,Ltd.
292, Yoshida, Totsuka, Yokohama, Kanagaya, 2440817 Japan
E-mail: ebina@sdl.hitachi.co.jp

を同様とすることで従来の関数 I/F に影響を与えず置換え可能である．

(2) 最適なデータ転送

PIO 方式では，アルゴリズムと鍵の設定に関して認証・暗号演算部のレジスタ設定を 12 項目行うのに対し，DMA 方式では PIO 方式でのレジスタ設定箇所に加えて(e)(g)の動作を行うための DMAC の設定に 19 項目，(f)の動作を行うための内部 DMAC の設定に 25 項目必要である．しかしながら，DMA 方式では PIO 方式に比べて設定項目数は多い半面，CPU を介さない 16byte 単位のデータ転送により高速処理が可能であることから，データ長が長い場合に有効である．以上のことから, データ長により PIO 方式と DMA 方式を使い分けることで，暗号ハードウェアとメインメモリ間のデータ転送を効率的に行いシステム性能の高速化することが可能である．

(3) 演算処理とデータ転送の並列化

通信プロトコル部はシングルスレッドのタスクで動作するため，現在処理を行っているパケットと並列して次パケット処理を行えるよう通信プロトコル部の実装を変更することにより DMA 方式時の並列化が可能である．

3.2 制御ソフトウェアの構成

並列化に対応するためには，OS のタスク/割込み処理への影響も多く複雑な構成となる．そこで，通信プロトコル部のみの変更で容易に実現し，性能を引き出すことを行った．これを実現する制御ソフトウェアの構成と復号時のデータの流れを図 2に示す．制御ソフトウェアは，ソフトウェアによる復号処理 I/F と同様にデータの受け渡しを実現する暗号/復号 I/F 部を介して通信プロトコル部から起動する．方式選択部はデータ長を判別し，PIO 方式と DMA 方式の選択，ハードウェア設定部は通信プロトコル部より通知されたアルゴリズムと鍵を設定するものである．また，DMA 方式使用時には DMAC の設定を行う．

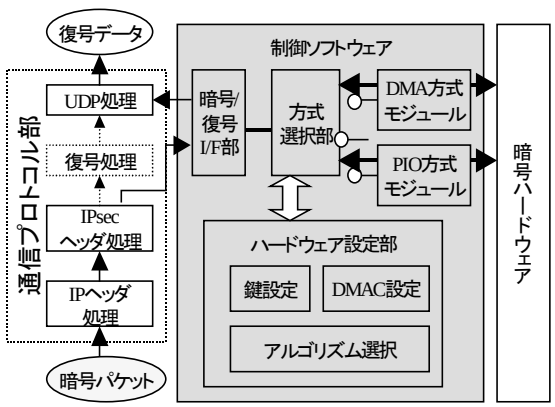

図 2　制御ソフトウェア構成と復号の流れ

4　性能評価

組み込み系 CPU（133MHz）と IPv6 対応 OS（Usagi-Linux(*1)）環境下で，UDP/IP 通信時の処理時間の測定を行うことで検証した(2)．本環境では制御ソフトウェアへの暗号データ入力から復号データ出力までの処理時間は，(1) 64byte データ処理 : DMA モードに比べて PIO モードで 32%，(2) 1024byte データ処理 : PIO モードに比べて DMA モードで 9%高速化可能である．各モードの測定結果より方式選択のスレッシュホールドデータサイズは 512byte である．この場合のハードウェア適用効果を図 3に示す．3DES の復号処理ではソフトウェア処理に比べ，1024byte では 1/16 に短縮でき，UDP/IP 全体のスループットはソフト処理の 7 倍の効果が得られた．暗号/復号処理の高速化により，組み込み機器における IPsec 通信を実現することが可能となる．

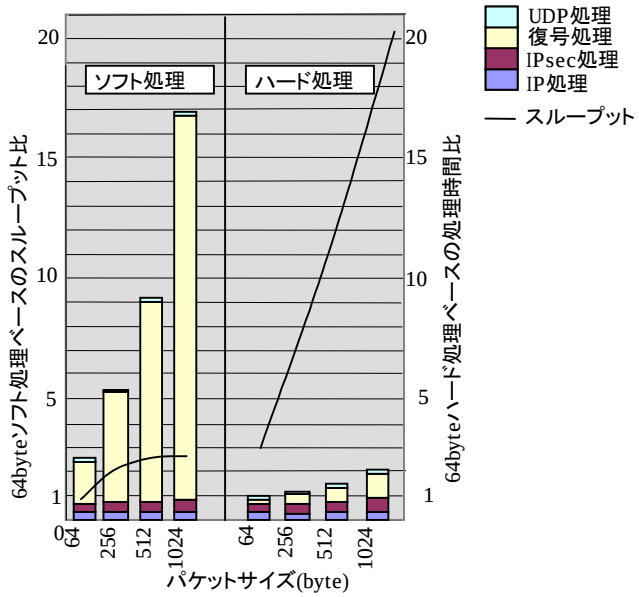

図 3　プロトコル処理時間の内訳

5　おわりに

認証・暗号演算処理を行うハードウェアをソフトウェアで処理する通信プロトコル部から利用するための，制御ソフトウェア実装を検討し，組み込み機器における認証・暗号ハードウェアの適用時の効果を明らかにした．

6　参考文献

1) 永井　靖 : IPsec のハードウェア実装方式の提案, 情報処理学会第 65 回全国大会，2003,3

2) 日立製作所　半導体事業部　半導体グループカスタマサービス本部 : SH7709A　ハードウェアマニュアル　第 5 版，2001,9

3) 馬場達也 : マスタリング IPsec，オライリー・ジャパン，2001

*1) Linux は,Linus Torvalds の米国及びその他の国における登録商標あるいは商標である