DEWS2006 4C-oi5

# 高遅延環境における IP-SAN を用いた 暗号処理最適化ミドルウェアの実装と性能評価

神坂紀久子† 山口 実靖†† 小口 正人†

† お茶の水女子大学 〒 112–8610 東京都文京区大塚 2–1–1
†† 東京生産技術研究所 〒 153–8505 東京都目黒区駒場 4–6–1
E-mail: †kikuko@ogl.is.ocha.ac.jp, ††sane@tkl.iis.u-tokyo.ac.jp, †††oguchi@computer.org

**あらまし** iSCSI, TCP/IP プロトコルと Ethernet を使用して構築する IP-SAN が提案され，普及し始めている．IP-SAN の登場で，専用のハードウェアを使用し，接続距離に限界がある FC-SAN と比較して，ストレージ管理コストの大幅な削減と大規模広域 SAN の実現が可能になった．iSCSI を用いてストレージにアクセスする際には，オープンなネットワークを介するため安全に通信を行うことが重要であるが，データの暗号化による通信性能の低下が懸念されている．そこで本稿では，従来の暗号化通信方式である IPsec に代わり，暗号化を効率的に行う暗号処理最適化ミドルウェアシステムを実装し，評価を行った．その結果，提案システムは高遅延環境において IPsec より高い性能を示し，非常に有効であることがわかった．

**キーワード** ストレージシステム，セキュリティ，性能評価，iSCSI

# Performance Evaluation and Implementation of IP-SAN based Middleware Optimized in a Long-Latency Environment

Kikuko KAMISAKA†, Saneyasu YAMAGUCHI††, and Masato OGUCHI†

† Ochanomizu University Otsuka 2–1–1,Bunkyo-Ku, Tokyo 112–8610 Japan
†† Institute of Industrial Science, The University of Tokyo Komaba 4–6–1, Meguro-ku, Tokyo, 153–8505 Japan
E-mail: †kikuko@ogl.is.ocha.ac.jp, ††sane@tkl.iis.u-tokyo.ac.jp, †††oguchi@computer.org

**Abstract** IP-SAN constructed using an iSCSI over TCP/IP protocols and Ethernet has been proposed and become accessible to the public. Traditional SAN (FC-SAN) is built using storage-specific hardware and has distance limitation. On the contrary, IP-SAN has reduced storage management costs and provided a possibility to realize a larger scale system on wide area networks, compared with FC-SAN. In the case of accessing remote storage, because IP-SAN is built on open networks, it is important to communicate securely. However, performance degradation by inefficient encryption processing on IP-SAN is a serious issue. In this paper, we have implemented middleware optimized for encryption processing more efficiently compared with traditional method, IPsec. Furthermore, we have evaluated the performance of the middleware. As a result of the experiments, our system achieves higher performance than a system using IPsec does, especially in a long-latency environment.

**Key words** Storage System, Security, Performance Evaluation, iSCSI

## 1. はじめに

ブロードバンドネットワークの急速な普及により，企業や組織で大容量のデータをネットワークを通じてストレージに保存，管理することが多くなってきた．地震などの自然災害に備え，非常災害対策のために大規模なデータを遠隔地にバックアップすることも日常的に行われている．現在，ストレージ分野で大きな課題となっているのが，急速なデータ容量の増加とそれにに伴うストレージの管理コストである．その問題に対応するため，多くの企業ではストレージシステムとサーバを高速なネットワークで接続するネットワークストレージである SAN (Storage Area Network) を導入している [1]．従来のストレージ接続形態である，サーバにストレージシステムを直接接続する DAS (Direct Attached Storage) と比較して，SAN はストレージを統合することによりディスクの効率的な利用と集中管理を可能にし，バックアップやキャパシティプランニングなどに伴う管理負荷を大幅に削減できる．

従来の SAN では，高速なネットワーク技術である Fibre Chan-

nel (FC) を使用して構築されてきた．しかし，FC を用いる SAN (FC-SAN) では，FC 用のスイッチやインタフェースなどのハードウェアが高価であること，FC を管理する技術者が少ないことなど SAN を新たに導入するには障害があった．また FC-SAN は，接続距離が 10km 程度に制限されてしまうため，大規模な広域 SAN を容易に実現することが困難であった．

これらの制限を解消するために，FC よりも安価に導入や管理が可能である IP-SAN が提案され，注目を集めている．IP-SAN は FC に代わり，TCP/IP プロトコルと Ethernet で構築される SAN であり，ハードウェアが安価で管理技術者も多いため，導入・管理コストを抑えることができる．また既存のネットワークとのシームレスな統合ができ，接続距離に制限がないことなどから，大規模広域 SAN を実現することが可能となり，近年重要視される非常災害対策を目的とした遠距離バックアップなども期待されている．IP-SAN は TCP/IP を使用しているため，FC-SAN と比較すると通信速度や CPU の負荷が問題であると指摘されている．しかし，Gigabit/10 Gigabit Ethernet の登場など性能向上のペースが比較的速いことから，今後，互換性を保ったまま性能が向上されることを期待できる．

IP-SAN で使用される技術で代表的なものに，2003 年 2 月に IETF により正式承認された iSCSI (Internet SCSI) プロトコルがある [2]．iSCSI では，TCP/IP パケットの中に SCSI コマンドをカプセル化することによって，サーバなどの計算機 (イニシエータ) とストレージシステム (ターゲット) 間で，ブロックレベルのデータ転送を行う通信プロトコルである．これにより，iSCSI は標準の SCSI 命令体系を使用したまま，IP ネットワークを介して遠隔地にあるストレージへのデータ転送を可能にする．

一方，IP-SAN ではアクセスしやすいオープンなネットワークを介するため，セキュリティが新たな課題として持ち上がっている．iSCSI では，そのデータセキュリティ対策の方法として，IP パケットに対して強固な暗号化と認証機能を提供する IPsec をサポートしている．しかし，大量のデータがバースト的に発生し，長距離間でデータ転送される場合のある SAN では，下位の IP 層で暗号化する IPsec は，効率的な処理を行っているとは言いがたい．IP-SAN において安全性と高性能はトレードオフの関係にあるため，双方を考慮した上で適切なストレージアクセスを行う必要がある．

そこで本稿では，IPsec を用いる代わりに暗号化を効率的に行う暗号処理最適化ミドルウェアシステムを構築した．また，低遅延，高遅延環境において構築したシステムのシーケンシャルライト性能を評価した．その結果，IPsec 使用時と比較して，実装した暗号処理最適化ミドルウェアシステムは高遅延環境において高い性能を示し，非常に有効であることがわかった．さらに本稿では，スループットモデリングによる性能の解析を行い，測定結果と比較して評価した．

本稿の構成は以下の通りである．まず，2 章において暗号処理最適化手法，3 章でその実装について説明し，4 章で実装したシステムの低遅延と高遅延環境における性能評価実験と考察について述べる．5 章で性能評価実験の解析，6 章で関連研究について述べ，最後に 7 章でまとめる．

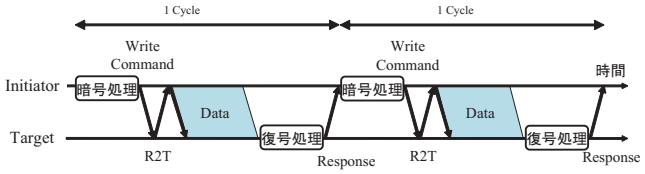


図 1　上位層で暗号化した場合のシーケンシャルライトアクセス

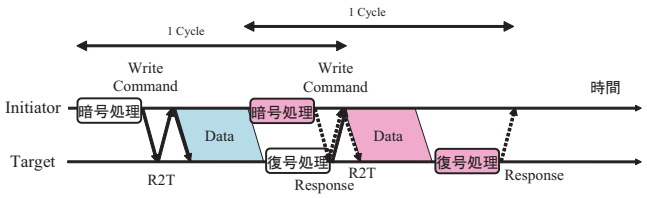


図 2　暗号処理最適化手法

## 2. 暗号処理最適化手法

### 2.1 IPsec を用いた暗号化通信方式の問題点

TCP/IP 技術に基づく iSCSI プロトコルを用いて安全にストレージへデータ転送を行う際には，IP ネットワークで広く普及している IPsec を利用することが可能である．IPsec は IP パケットを暗号化するため，上位のソフトウェアを変更する必要がなく透過的に暗号化を行うことができる．一方，IP-SAN において IPsec を使用する場合，暗号化処理によって大幅にストレージアクセス性能が低下する．

暗号化処理によって性能が低下するのは当然であるが，それに加え，IP-SAN における IPsec の暗号化処理方法が性能に大きな影響を及ぼしていると考えられる．つまり，IPsec は下位層に位置しているため，大規模なデータをシーケンシャルにアクセスする場合であっても，上位層において細分化されたデータセグメントに対して，下位層で逐次的に暗号化処理を繰り返すのみである．そのため，IPsec は SCSI 層や TCP 層など上位層の処理内容を把握した上で柔軟な処理を行うことができない．さらに，IPsec の暗号処理による性能低下を改善する場合には，性能向上機能はカーネル内部に存在する下位層 IP の改良を余儀なくされるため容易には実現できない．よって，広帯域かつ遠距離の環境でデータ転送されることの多い SAN においては，暗号化の逐次処理を繰り返すのみの IPsec は効率的な暗号化を行うことが困難である．

### 2.2 上位層における暗号化手法

我々は IP-SAN において，IPsec の逐次的な暗号化処理の問題点を解決するために，IPsec を使用する代わりに，IP 層より上位層で暗号化処理を行う手法を提案した [3]．上位層で暗号化を行うことによって，下位の IP 層の実装に変更を加えることなく，アプリケーションや SCSI 層，TCP 層などにおける上位層の処理に柔軟に対応することができる．ユーザが選択したアプリケーションにおいて暗号化処理が最適になるように，上位層のソフトウェアで機能を自由に作成して組み込むことができ，データの機密性を保ったまま，ストレージへのアクセス性能の向上が実現できる．

### 2.3 暗号処理最適化手法

前節で述べた上位層において暗号化を行う手法により，暗号化処理を効率的に行い，性能を向上させる機能を容易に適用することが可能になった．そこで本節では，IPsec の暗号化による性能低下に対する解決策として提案した暗号処理最適化手法について述べる [4] [5] [6].

通常のシーケンシャルライトアクセスでは，まず SCSI write コマンドがイニシエータからターゲットへ転送される．次に，受信側の準備が整うのを待ってデータを送信するために，ターゲットは受信可能となったときに Ready to Transfer (R2T) を送信する．これを受けたイニシエータはデータを送信し，ターゲットがディスクにデータを書き込んだ後に Response コマンドを送信する．その際，データの転送が 1 つのパケットに収まらない場合は，複数に分けてデータの転送を行う．

図 1 は，上位層で暗号化する場合の iSCSI シーケンシャルライトアクセスのシーケンスである．上位層で暗号化する場合は，イニシエータでデータの暗号化を行ってからネットワーク上に転送し，ターゲットで復号を行った後にディスクにデータの書き込みを行う．その際，相手側がデータの暗号化や復号を行っている間，あるいは SCSI コマンドを転送している間などに通信の待ち時間が発生する．

これに対して図 2 は，暗号処理最適化手法を適用した場合におけるシーケンシャルライトアクセスのシーケンスの一例である．同図のように，暗号処理最適化手法では，そのような通信の待ち時間を利用して次のデータを先読みし，暗号化の先処理を連続的に行う．それにより，CPU 処理の空き時間を有効に使用することができ，暗号化通信の性能を向上させる．

## 3. 暗号処理最適化ミドルウェアの実装

本稿では，上位層で暗号化処理を適用する手法と暗号処理最適化手法に基づくミドルウェアを実装した．本節では，実装したミドルウェアの概要 (図 3) について述べる．

本稿においてイニシエータとターゲットには，オープンソースのオペレーティングシステムである Linux を使用した．また iSCSI にも，ソースコードの可用性とコストの安さを考慮し，オープンソースのソフトウェアを使用している．

通常，iSCSI を用いた場合の階層構造は，イニシエータでは，上位層からアプリケーション，ファイルシステム，ブロックデバイス，SCSI/iSCSI，TCP/IP のようになっている．一方，ターゲットでは SCSI 層の上はディスクになっており，両者は異なる階層構造になっている．本システムでは，上位層における暗号化機能と暗号処理最適化機能は，同じミドルウェアとして構築する必要があるため，イニシエータ側ではユーザ空間で動作するミドルウェアとして実装した．本稿で実装したイニシエータのミドルウェアでは，ファイルシステムより上位で実装を行っているが，暗号化機能はファイルシステムやブロックデバイスに組み込むことも可能である．

ターゲットではすべての階層処理がカーネル空間で動作しているため，上位層における暗号化機能を SCSI 層の上位に位置するミドルウェアのカーネルモジュールとして作成した．カーネルモジュールでは，SCSI デバイスドライバで発行している write 命令を横取りしてデータを復号し，ディスクに平文として書き込まれる．また本システムの暗号化機能として，現在広く使われており，安全性が高く，IPsec でもデフォルトで使用されている 3DES (Triple Data Encryption Standard) 暗号化アルゴリズムを用いている．その暗号化処理実装コードは，比較のために本稿の性能評価実験で使用している IPsec と同じものである．

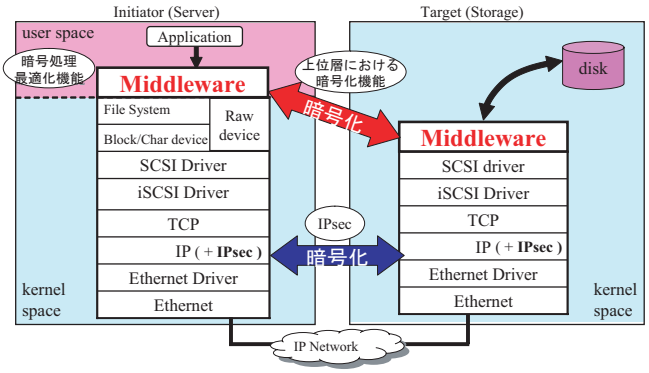


図 3　暗号処理最適化ミドルウェアの実装

イニシエータ側で実装した暗号化の先処理を連続的に行う最適化は，ミドルウェア中でプロセスを複数に分岐し，並行して処理を実行している．たとえば，まずイニシエータの単一プロセスで write 命令が発行され，ターゲットが復号処理を行っている間などの待ち時間に，ミドルウェアでその次に write すべきデータセグメントの暗号処理を行う．それによって，暗号化の先処理が可能になり，全体の処理時間が短縮される．実装したミドルウェアは，アプリケーションが単一プロセスによってストレージアクセスを行う場合に，連続的に暗号化の先処理を実行することができる．このように暗号処理最適化機能をミドルウェアとして独立させることにより，簡単に性能を向上させる手法を適用し，機能を追加，改良することができる．

## 4. 性 能 評 価

効率的に暗号化を行う暗号化処理最適化ミドルウェアの性能を評価するため，実装したシステムを用いて性能評価実験を行った．本実験ではストレージアクセス性能のみを評価するため，キャッシュによる性能への影響を排除することを目的として，raw デバイスを使用したシーケンシャルライトアクセスを実行してスループットと CPU 使用率を測定し，IPsec 使用時の性能と比較している．また，本実験では，IP-SAN が非常災害対策などのために比較的遠距離で使用されることを想定し，低遅延，高遅延環境において提案システムの性能を評価した．

### 4.1 実 験 環 境

性能評価実験環境を図 4 に示す．iSCSI イニシエータと iSCSI ターゲットを Gigabit Ethernet で接続し，TCP/IP 接続を確立した後，高遅延環境を構築するために，イニシエータとターゲットの間に人工的な遅延装置として FreeBSD Dummynet を設置した [7]．イニシエータとターゲット間の片道遅延時間は “0ms”〜“64ms” と設定している．たとえば日米間のデータ転送の場合には片道遅延時間 50ms 以上となるが，国内における長距離バック

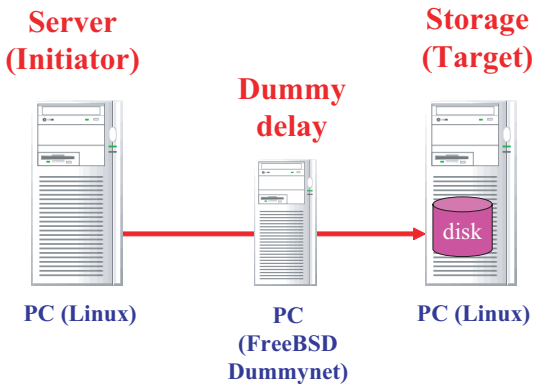


図 4　高遅延環境における実験環境

表 1　性能評価実験環境 1：使用計算機

| | |
|---|---|
| OS | Initiator : Linux 2.4.18-3<br>Target : Linux 2.4.18-3<br>Dummynet : Free BSD 4.9 - RELEASE |
| CPU | Intel Xeon 2.4GHz |
| Main Memory | 512MB DDR SDRAM |
| HDD | 36GB SCSI HD |
| NIC | Initiator, Target :<br>Intel PRO/1000XT Server Adapter<br>Dummynet :<br>Intel PRO/1000MT Server Adapter |

表 2　性能評価実験環境：使用実装

| | |
|---|---|
| iSCSI | UNH-iSCSI Initiator and Target for Linux ver. 1. 5. 3 |
| IPsec | FreeS/WAN ver. 2.01 |

アップを想定するには十分な遅延時間である．遅延時間 “0ms” は，遅延装置を介するが人工的な遅延は発生しない場合である．また受信側の TCP Window Size は，実験に影響を与えないよう十分に大きい 8MB としている．

実験に用いたシステム環境を表 1，2 に示す．iSCSI の実装には，ニューハンプシャー大学 InterOperability Lab [8] が提供しているオープンソースの実装 (UNH-iSCSI) を用い，IPsec の実装には，Linux において広く利用されているオープンソースの FreeS/WAN [9] を用いた．IPsec の設定に，ホスト間の通信を暗号化するトランスポートモードおよび ESP プロトコルを使用している．

### 4.2　基 本 性 能

まず iSCSI を用いたシーケンシャルライトアクセスの暗号化を行わない場合の基本性能を，片道遅延時間を 0ms から 64ms の環境において測定した．図 5 はスループット，図 6 は CPU 使用率である．以降の性能評価実験において，横軸のブロックサイズは，OS に対して raw デバイスの write システムコールを発行する際にアプリケーションが指定したデータサイズである．また CPU 使用率は，Linux の iostat コマンドを用いてターゲット側で測定した．

全体の傾向として遅延が大きくなるほどスループットが大幅に低下し，CPU の負荷が減少する．また，スループットはブロックサイズが大きくなると高くなるが，遅延が 0ms の場合にはブロックサイズ 32KB 付近で，遅延が 1ms 以上の場合にはブロックサイズ 256〜512KB 付近でほぼ頭打ちとなりそれ以上は増加しない．CPU 使用率は全体的に低い値であり，最も高い値でも遅延 0ms で 23%，それ以外はほとんどの場合が 20%以下である．

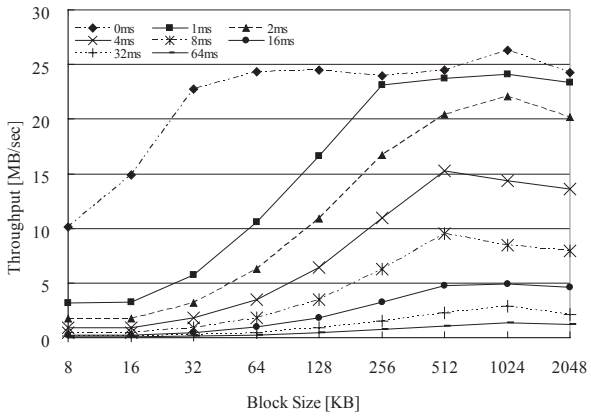


図 5　iSCSI シーケンシャルライトアクセスの基本性能（スループット）

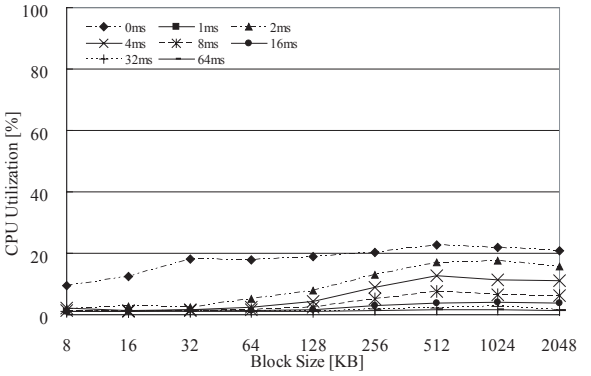


図 6　iSCSI シーケンシャルライトアクセスの基本性能（CPU 使用率）

### 4.3　スループット測定結果と考察

次に，iSCSI シーケンシャルライトアクセスで暗号化を行った場合の性能を測定した．片道遅延時間を 0ms から 64ms まで変化させ，本稿で実装したシステムと IPsec のスループットを比較している．また本実験のスループットと CPU 使用率の測定結果では，暗号処理最適化において，連続して暗号化を先処理するために分岐する処理数を 2，4，6，8，10 まで変化させ，それらを “OP-2”，“OP-4”，“OP-6”，“OP-8”，“OP-10” として表している．

図 7 は片道遅延時間 1ms の場合のスループットであり，図 8 は 16ms の場合，図 9 は 64ms の場合のスループットである．これらの図より，暗号処理最適化手法によって暗号化の先処理をした場合は，IPsec のスループットと比較して，大幅に性能が向上していることがわかる．IPsec のスループットはこれらの遅延時間において，ブロックサイズが 512KB 以上で飽和していることが確認される．

遅延時間が小さい 1ms においてブロックサイズが小さい場合

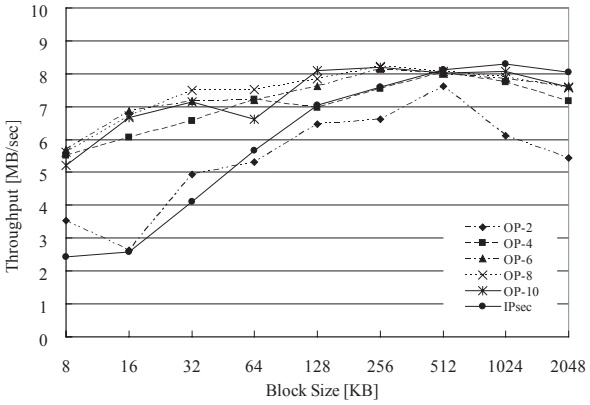


図 7　片道遅延時間 1ms におけるスループット

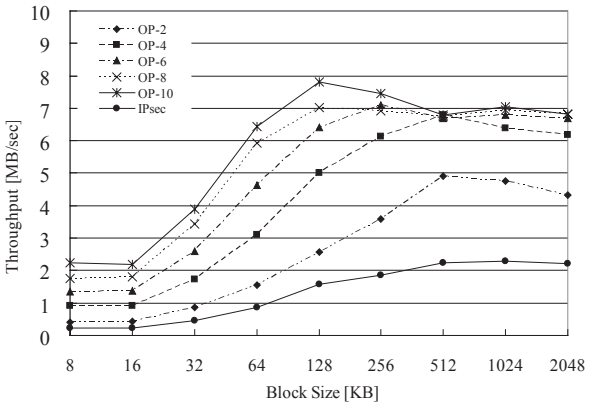


図 8　片道遅延時間 16ms におけるスループット

には，処理数が 4 から 10 まで (OP-4〜OP-10) の提案システムのスループットは IPsec より大幅に高い．ブロックサイズが大きくなるにつれ，IPsec と提案システムの性能差は小さくなる．

一方，遅延時間が大きい 16ms，64ms において (図 8，9)，ブロックサイズが大きい場合は，すべての処理数において提案システムの方が IPsec よりも大幅にスループットが増加する．ブロックサイズが比較的小さい場合も，IPsec よりスループットが高くなり，処理数による性能向上への影響は，ブロックサイズが大きい場合よりも大きくなることがわかった．全体性能として，片道遅延時間 16ms において IPsec の場合には最大でも 2.3MB/sec であるが，提案システムでは最大で 7.8MB/sec のスループットを達成した．片道遅延時間 64ms の場合でも，IPsec は最大で 0.6MB/sec であるのに対し，提案システムでは 2.8MB/sec を達成した．基本性能実験におけるスループットが，片道遅延時間 16ms の場合は最大で 4.9MB/sec，64ms の場合には 1.2MB/sec であるため，基本性能以上のスループットが得られている．

表 3 は，IPsec を 1 とした場合の IPsec に対するスループット向上比率を各ブロックサイズごとに計算し，それらの向上比率を全ブロックサイズで平均したものである．同表より，片道遅延時間 0ms である低遅延環境では，提案システムの全処理数におけるスループットは IPsec よりもやや低いが，片道遅延時間 1ms 以上の環境では提案システムの方が IPsec よりも高くなる．

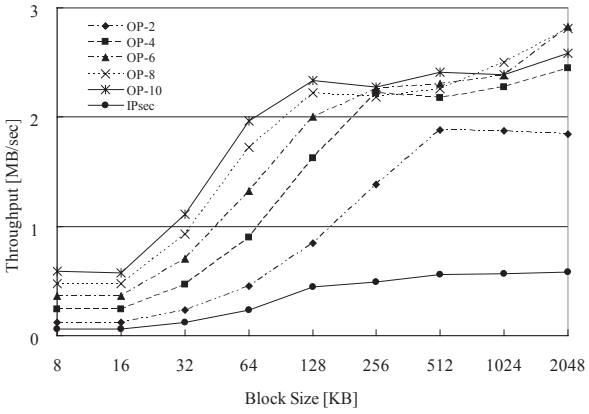


図 9　片道遅延時間 64ms におけるスループット

表 3　IPsec に対するスループットの向上比率 (片道遅延時間 0ms〜64ms)

| | IPsec | OP-2 | OP-4 | OP-6 | OP-8 | OP-10 |
|---|---|---|---|---|---|---|
| 0ms | 1.00 | 0.83 | 0.91 | 0.93 | 0.92 | 0.87 |
| 1ms | 1.00 | 0.97 | 1.37 | 1.45 | 1.46 | 1.42 |
| 2ms | 1.00 | 1.08 | 1.73 | 2.00 | 2.04 | 1.98 |
| 4ms | 1.00 | 1.34 | 2.21 | 2.79 | 3.20 | 3.24 |
| 8ms | 1.00 | 1.49 | 2.67 | 3.52 | 4.20 | 4.63 |
| 16ms | 1.00 | 1.90 | 3.35 | 4.36 | 5.18 | 5.85 |
| 32ms | 1.00 | 2.22 | 3.90 | 4.57 | 5.42 | 6.22 |
| 64ms | 1.00 | 2.50 | 4.00 | 5.08 | 5.96 | 6.64 |

また遅延時間が増加するにしたがって，提案システムと IPsec の性能差が大きく広がる．その理由として，低遅延環境においては，通信時間はデータセグメントの暗号化・復号時間と比較して相対的に短いため，提案手法における暗号化の処理数を増加させてもそれほどの効果は見られない．しかし，高遅延環境においては，通信時間はデータセグメントの暗号化・復号時間と比較して相対的に長くなる．そのため，通信の待ち時間，つまり CPU 処理の空き時間が長くなることにより，1 つの暗号化サイクルが終了しないうちに，連続的に次のデータセグメントを暗号化する暗号処理最適化手法の効果が高くなるため，性能向上比率が大きくなったと考えられる．

片道遅延時間が 64ms で，処理数が 10 回の場合には，提案システムは IPsec よりも 2.5 倍から 6.6 倍スループットが向上したことを確認できた．

### 4.4　CPU 使用率測定結果と考察

図 10，11，12 は，片道遅延時間を 1ms，16ms，64ms の環境において，本稿で実装したシステムを用いた際の CPU 使用率と IPsec を使用した際の CPU 使用率である．

表 4 は，各遅延時間で測定したそれぞれの CPU 使用率を全ブロックサイズで平均したものである．図 10〜12 および表 4 より，全遅延時間において提案システムは IPsec よりも CPU の負荷が高く，提案システムの処理数を増加させると CPU 負荷も大きくなる．一方，遅延時間が増加した場合，全体的に CPU の負荷が小さくなる．これは通信時間が長いために CPU 処理の空き時間が長くなり，その間に次のデータの暗号化を進める

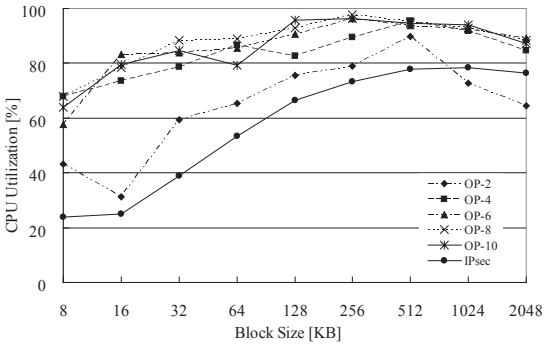


図 10　片道遅延時間 1ms における CPU 使用率

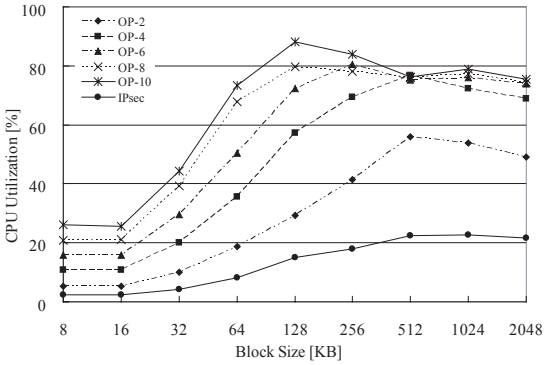


図 11　片道遅延時間 16ms における CPU 使用率

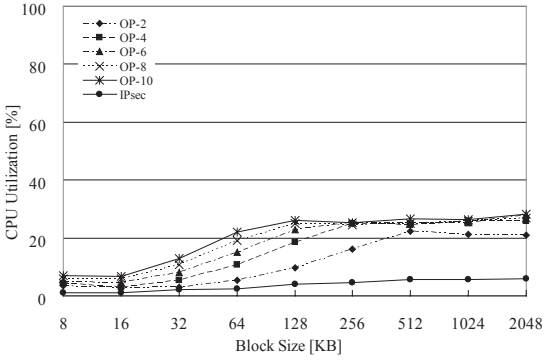


図 12　片道遅延時間 64ms における CPU 使用率

余裕が大きくなるためと考えられる．また，片道遅延時間が 64ms の場合には，最大でも 20%と低い値であるため，高遅延環境において提案システムを用いる有効性が高いといえる．

図 13，14 は，片道遅延時間 16ms，64ms の場合において，各ブロックサイズにおける CPU 使用率をスループットで割ることによって正規化したグラフである．これらの図より，遅延時間 64ms のブロックサイズ 8KB～32KB の場合を除いて，提案システムと IPsec はほぼ同じ値を示している．このことから，表 4 に見られるように，提案システムの方が IPsec より CPU の負荷が高いが，高いスループットを示していることを考慮すると，提案システムの CPU 使用率は妥当なものであると考えられる．

### 4.5　セキュリティの観点からの考察

セキュリティの観点から IPsec との比較について考察すると，

表 4　各遅延時間における CPU 使用率の平均 (%)

| | IPsec | OP-2 | OP-4 | OP-6 | OP-8 | OP-10 |
|---|---|---|---|---|---|---|
| 0ms | 77.9 | 78.4 | 86.4 | 88.5 | 87.1 | 83.0 |
| 1ms | 57.0 | 64.5 | 83.4 | 85.8 | 87.8 | 86.1 |
| 2ms | 47.4 | 55.1 | 77.5 | 85.6 | 87.2 | 86.0 |
| 4ms | 39.7 | 48.1 | 69.7 | 78.7 | 83.7 | 84.8 |
| 8ms | 22.5 | 37.2 | 57.2 | 67.0 | 72.2 | 74.3 |
| 16ms | 13.0 | 29.9 | 46.9 | 54.4 | 59.4 | 63.5 |
| 32ms | 7.2 | 20.3 | 31.1 | 30.8 | 33.9 | 37.0 |
| 64ms | 3.6 | 11.7 | 15.9 | 17.7 | 18.8 | 20.1 |

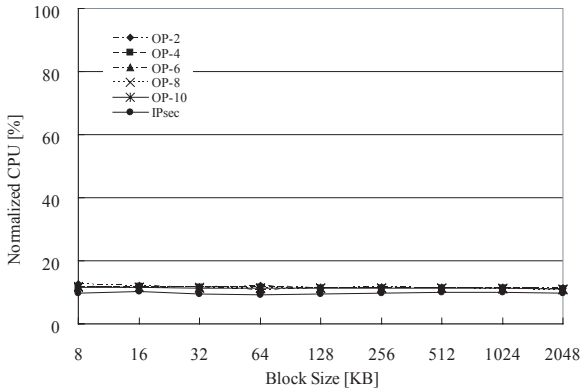


図 13　片道遅延時間 16ms におけるスループットで正規化した CPU

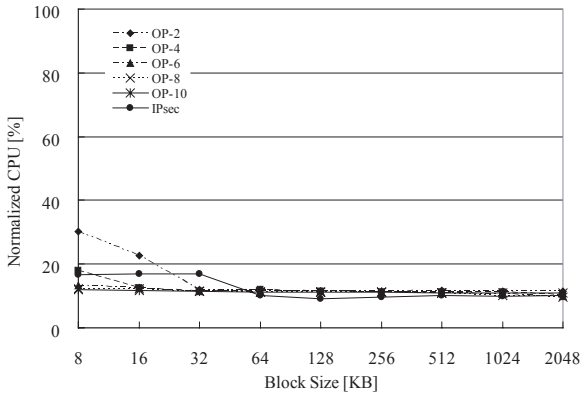


図 14　片道遅延時間 64ms におけるスループットで正規化した CPU

IPsec はゲートウェイ間通信を暗号化するトンネルモードを用いた場合には TCP 層のデータと TCP ヘッダを暗号化する．本稿の提案システムでは，上位層で暗号化を行っているため，TCP ヘッダは暗号化されないが，本実験は IPsec のトランスポートモードと比較を行っており，どちらの方式においても IP ヘッダは暗号化されない．よって，セキュリティの面においてはどちらの方式も同等のレベルであると考えられる．

暗号化アルゴリズムについては，次世代暗号標準である AES (Advanced Encryption Standard) が，2000 年に NIST (National Institute of Standards and Technology) によって選定された．これに基づき，ソフトウェアやハードウェアによる実装が可能になり，AES 暗号化が増えてきつつある．しかし，現状では，まだ多くの機器において，3DES がデフォルトとして設定され広く使用されている．本稿の実装では実用的な面を考慮するために

3DES を採用したが，暗号処理最適化ミドルウェアは AES においても適用可能であるため，AES を用いた場合でも性能向上が期待できる．

## 5. 性能評価実験の解析

提案手法の性能評価を行う際の指標として，シーケンシャルライトアクセスにおけるスループットのモデル化を行った．まず，暗号処理最適化を適用せず，上位層で暗号化を行う場合の提案システムのスループットを考える．

イニシエータ側でデータが暗号化され，ターゲット側でデータの復号を行った後に Response コマンドが転送されるまでを 1 サイクルとする．図 1 より，1 サイクルに要する時間 (1 サイクル時間) は RTT（Round Trip Time) などを用いて以下の式で表される．

1 サイクル時間

$$= 2 \times RTT + \text{データ転送時間} + \text{暗号化時間} + \text{復号時間} \quad (1)$$

また，データ転送時間は以下のように表される．

$$\text{データ転送時間} = \frac{\text{転送データサイズ}}{\text{下位層のスループット}} \quad (2)$$

ここで，下位層のスループットとは，iSCSI 層以下のスループットであり，iSCSI を用いない場合の単純なソケット通信における値である．下位層のスループットを測定したところ，約 11.584MB/sec であった．

また，暗号化時間は以下の式で表される．

$$\text{暗号化時間} = \frac{\text{転送データサイズ}}{\text{暗号化速度}} \quad (3)$$

暗号化と復号の 3DES アルゴリズムはほぼ同じであるため，復号時間も同様に表すことができ，計測により暗号化時間と復号時間はほぼ変わらない値であった．本稿の実装では，3DES で暗号化した場合の暗号化速度は 10.822MB/sec であることが確認されている．

以上により，暗号処理最適化を適用しない場合における提案システムの予測されるスループットは，ブロックサイズを用いて次の式でモデル化することが可能である．

スループット

$$= \frac{\text{ブロックサイズ}}{2 \times RTT + \frac{\text{ブロックサイズ}}{\text{下位層のスループット}} + \frac{\text{ブロックサイズ}}{\text{暗号化速度}} + \frac{\text{ブロックサイズ}}{\text{復号速度}}} \quad (4)$$

一方，通信の待ち時間に次のデータの暗号化処理をする最適化を行った後のスループットモデルを考える．ここでは，提案システムの処理数を 2 としてモデル化した．図 2 より，iSCSI の 1 サイクルがオーバーラップし，暗号化処理に復号処理が隠蔽されるため，

スループット

$$= \frac{\text{ブロックサイズ}}{RTT + \frac{\text{ブロックサイズ}}{\text{下位層のスループット}} + \frac{\text{ブロックサイズ}}{\text{暗号化速度}}} \quad (5)$$

表 5　スループット計算値とに最適化を行わない提案システムの実測値（片道遅延時間 16ms）

| Block Sizse | Calculated Value (MB/sec) | Actual Measurement (MB/sec) |
|---|---|---|
| 8KB | 0.118 | 0.239 |
| 16KB | 0.229 | 0.235 |
| 32KB | 0.431 | 0.421 |
| 64KB | 0.771 | 0.794 |
| 128KB | 1.276 | 1.342 |
| 256KB | 1.896 | 2.069 |
| 512KB | 2.505 | 2.831 |
| 1024KB | 2.984 | 2.810 |
| 2048KB | 3.300 | 2.397 |

表 6　スループット計算値とに最適化を行った提案システムの実測値（片道遅延時間 16ms）

| Block Sizse | Calculated Value (MB/sec) | Actual Measurement (MB/sec) |
|---|---|---|
| 8KB | 0.234 | 0.410 |
| 16KB | 0.448 | 0.444 |
| 32KB | 0.830 | 0.864 |
| 64KB | 1.446 | 1.561 |
| 128KB | 2.299 | 2.581 |
| 256KB | 3.259 | 3.594 |
| 512KB | 4.120 | 4.911 |
| 1024KB | 4.744 | 4.751 |
| 2048KB | 5.138 | 4.324 |

とモデル化することができる．

(4) 式により計算した計算値と，片道遅延時間 16ms において，最適化を行わずに上位層で暗号化する提案システムのスループット実測値を表 5 に示す．また，最適化後のスループットのモデル式 (5) より計算した計算値と処理数を 2 として最適化を行った場合の実測値を表 6 に示す．実測値はブロックサイズによってばらつきがあるが，スループットモデリングの計算値と実測値が近い値になっているため，このスループットのモデル式 (4)，(5) はほぼ正しいものであるといえる．従って，最適化処理を行い，暗号化処理をオーバーラップさせると性能が向上することがモデル式を使用して実証でき，本稿の暗号化の先処理による最適化の提案手法が性能に関して非常に有効であることがわかった．

## 6. 関 連 研 究

iSCSI の関連研究としては，まず文献 [10] において，Ng らは独自の SCSI over IP 実装を用いて IP ストレージの性能に関する詳細な測定と解析を行っている．同文献では 8KB のブロックサイズにおけるシーケンシャルアクセスの性能を測定し，人工的な遅延が存在する環境下で異なる OS とファイルシステムにおける性能比較を行っている．また同文献では，独自の実装においてホスト側におけるキャッシュの適用やアプリケーションによるプリフェッチが有効であることを指摘している．

文献 [11] においては，Sarkar らによる iSCSI のソフトウェア実装と TOE (TCP Offload Engine) や HBA (Host Bus Adapter) を

用いた iSCSI ハードウェア実装の比較に関する研究が行われており，ハードウェア実装は，CPU の負荷を軽減させることはできるが，総合的にはソフトウェア実装の方が性能が高くなることが実証されている．本稿では CPU 負荷を軽減するだけでなく，スループットなどの総合的な評価を対象としているため，iSCSI のソフトウェア実装を用いている．

文献 [12] において，Aiken らは iSCSI ソフトウェア実装における遅延を考慮した性能の比較と詳細な分析を行い，ネットワーク遅延が増加するにつれ，iSCSI プロトコルの通信性能が急激に低下するという結果を得た．

文献 [13] において，藤田らは iSCSI ターゲットに着目して実装手法を考慮し，2 種類の iSCSI のオープンソースソフトウェアを用いて，iSCSI の性能評価を行っている．同文献では，OS のカーネルが提供する標準機能を利用する手法よりも，変更を加えたカーネルが提供する iSCSI ターゲットに最適化された機能を利用する手法の方が性能が優れていることを確認している．本稿の手法はカーネルや iSCSI ソフトウェアに変更を加えずに，最適化された機能をミドルウェアで実装するものであり，同文献のアプローチとは異なる．

また文献 [14] では，FC-SAN と比較して性能面でハイエンドな環境には適さないと指摘されるソフトウェアベースの iSCSI イニシエータを用いて性能評価を行っている．その結果，10 Gigabit Ethernet 環境において，I/O 命令の多重数が多い条件では，ハイエンド向けとされるインターコネクト技術を使ったシステムよりも性能が高く，Read で 800MB/sec 以上，Write で 700MB/sec 以上のスループットが得られたことを報告している．

セキュリティを考慮した研究では，Tang らによってソフトウェアベースの iSCSI 実装における IPsec と SSL の性能が比較されており [15]，小さなデータサイズの場合は IPsec の方がスループットが高いが，データサイズが大きくなると SSL の方が早く改善されるという結果が得られている．また IPsec は SSL よりも CPU の負荷が高いことが実証されている．

現在まで iSCSI の性能に関する研究は多く発表されているが，暗号化などのセキュリティ実現方式が考慮された研究は十分になされていない．

## 7. まとめと今後の課題

本稿では，iSCSI ストレージアクセスを行う際，CPU の空き時間に次にデータの暗号化を先処理して性能を向上する暗号処理最適化ミドルウェアを構築した．また構築したシステムを用いて，片道遅延時間 64ms までの高遅延環境において iSCSI シーケンシャルライトアクセスを行った場合の性能を評価した．IPsec を使用した場合の性能と比較したところ，片道遅延時間が 1ms〜64ms の場合には，本稿のミドルウェアシステムが IPsec よりも最大で 1.4 倍〜6.6 倍スループットが向上した．従って本稿の提案システムは高遅延環境において IPsec より有効であることがわかった．また，スループットモデリングによる性能評価実験の解析を行い，測定結果と比較して評価を行った．その結果，実験により計測した値はモデル式にほぼ近い値を示しており，

今後の課題として，暗号処理最適化ミドルウェアを用いて低遅延環境において性能が向上する手法を提案し，データベースなどの実アプリケーションを使用した総合的な性能評価を行う．

## 謝　　辞

本研究は一部，文部科学省科学研究費特定領域研究課題番号 13224014 によるものである．

## 文　　献

[1] Storage Networking Industry Association, http://www.snia.org/.

[2] iSCSI Draft, http://www.ietf.org/rfc/rfc3720.txt.

[3] Kamisaka, K., Yamaguchi, S. and Oguchi, M.: Performance improvement of an iSCSI-based secure storage access, *the 16th IASTED International Conference on Pararellel and Distributed Computing and Systems (PDCS 2004)* (Gonzalez, T.(ed.)), IASTED, pp. 522–527 (2004).

[4] Kamisaka, K., Yamaguchi, S. and Oguchi, M.: Performance Evaluation of iSCSI System Optimized for Encryption Processing in the Upper Layer, *the International Special Workshop on Databases For Next Generation Researchers (SWOD2005) in conjunction with IEEE International Conference on Data Engineering (ICDE2005)*, pp. 204–207 (2005).

[5] 神坂紀久子, 山口実靖, 小口正人: iSCSI ストレージアクセスにおける暗号化処理の最適化を考慮したシステムの提案と性能評価, 先進的計算基盤システムシンポジウム (SACSIS 2005), pp. 435–442 (2005).

[6] 神坂紀久子, 山口実靖, 小口正人: iSCSI ストレージアクセス時の 3DES アルゴリズムを用いた上位層における暗号化適用方式の実装および IPsec との性能評価, 並列/分散/協調処理に関するサマー・ワークショップ　(SWoPP2005), 電子情報通信学会技術研究報告, CPSY2005-15〜26, pp. 13–18 (2005).

[7] 山口実靖, 小口正人, 喜連川優: iSCSI 解析システムの構築と高遅延環境におけるシーケンシャルアクセスの性能向上に関する考察, 電子情報通信学会和文論文誌 データ工学特集号, Vol. J87-D-I, No. 2, pp. 216–231 (2004).

[8] InterOperability Lab in the University of New Hampshire, http://www.iol.unh.edu/consortiums/iscsi/.

[9] FreeS/WAN Project, http://www.freeswan.org/.

[10] Ng, W. T., Hillyer, B., Shriver, E., Gabber, E. and Ozden, B.: Obtaining High Performance for Storage Outsourcing, *Proc. FAST 2002, USENIX Conference on File and Storage Technologies*, pp. 145–158 (2002).

[11] Sarkar, P., Uttamchandani, S. and Voruganti, K.: Storage over IP: When Does Hardware Support help?, *Proc. FAST 2003, USENIX Conference on File and Storage Technologies*, pp. 231–244 (2003).

[12] Aiken, S., Grunwald, D., Pleszkun, A. R. and Willeke, J.: A Perfomance Analysis of the iSCSI Protocol, *Proc. 20th IEEE Symposium on Mass Storage Systems and Technologies (MSS '03)*, pp. 123–135 (2003).

[13] 藤田智成, 小河原成哲: iSCSI ターゲットソフトウェアの解析, 情報処理学会誌コンピューティングシステム, Vol. 46, No. SIG3 (ACS 8).

[14] 藤田智成, 矢田浩二: 10GE 環境での iSCSI の性能評価, 並列/分散/協調処理に関するサマー・ワークショップ　(SWoPP2005), 電子情報通信学会技術研究報告, CPSY2005-15〜26, pp. 1–6 (2005).

[15] Tang, S.-Y., Lu, Y.-P. and Du, D. H. C.: Performance Study of Software-Based iSCSI Security, *Proc. First International IEEE Security in Storage Workshop*, pp. 70–79 (2002).