The system evaluation of IP Security processing with PC platform

Seiji Ariga (Faculty of Environmental Information, Keio Univ.)
Masaki Minami (Research and Development Center, Toshiba Corp.)
Hiroshi Esaki (Information Technology Center, Tokyo Univ.)

IP Security ソフトウェア処理の性能評価

有賀 征爾 (慶應義塾大学 環境情報学部)
南 政樹 (株式会社 東芝 研究開発センター)
江崎 浩 (東京大学 情報基盤センター)

Abstract

IP Security(IPsec) は，インターネットを利用するユーザに対して，安全な通信手段を提供する．しかし，ここで用いられている認証や暗号化の処理は，他の処理と比較して計算機資源を多く必要とするため，IPsec を利用することによる通信性能の低下が予測される．そこで，実験室環境と実トラフィック環境 (インターネット) において，IPsec を使用した通信の性能評価を行い，その利用可能性について考察した．

1 はじめに

インターネットのユーザが安全な通信を利用したい場合，既存のインターネット環境では，安全性を提供するアプリケーションやプロトコルを使用しなければならない．これは，その基盤技術である Internet Protocol(IP)[1] が安全な通信路を提供する仕組みを持たないことに起因している．従って，安全なネットワークの構築の構築の際に，様々なアプリケーションやプロトコルを組み合わせて利用することになり，安全にネットワークを使うためのコストが高くなっている．

その一方で，インターネットを利用した社会活動が増えており，インターネットは重要な社会基盤として急速に浸透しつつある．つまり，安全な情報社会を築くためには，その基盤であるインターネットを安全なネットワークにし，またこの安全なネットワークが簡単に構築できなければならない．

1.1 既存のセキュリティ機能

既存のインターネットでは，いくつかの方法で安全な通信路を提供している．その代表的な手法が，アプリケーション層におけるデータの暗号化である．利用するアプリケーションやプロトコル毎に通信内容を暗号化するため，送信ノードと受信ノード間に安全な通信路を構築できる．

この方法の代表的な例には，Secure Shell(ssh)[2] や，Secure Socket Layer(SSL)[3] を用いたアプリケーションが挙げられる．

しかし，この方法では，そのアプリケーションやプロトコルが暗号化の機能に対応していなければならず，必ずしも全てのユーザにとって有効とは言えない．というのも，ユーザが利用するアプリケーションやプロトコルは多岐に渡り，そのすべてが暗号化の機能に対応しているわけではなく，ユーザが意図的に暗号化の機能を持ったアプリケーションやプロトコルを，選択しなければならないからである．

1.2 IPsec

これまで，アプリケーションやプロトコル毎に個別に対応していたセキュリティ機能を，より統一的に扱えるようにするために，IP レベルでセキュリティ機能を実現する IP Security(IPsec) [4] が提案されている．IPsec には大きく分けて次の二つ機能がある．

† **通信相手の認証**
受信ノードがそのパケットが正しく送られてきたかどうかを認証できるように，送信ノードは送信するパケットに Authentication Header(AH) [5] と呼ばれるヘッダを付加し署名する．パケットを受けとった受信ノードは，その署名を検査し，そのパケットの正当性を判断する．もし署名が正当であればそのパケットを次の処理に渡し，もし署名が不当であればそのパケットを破棄する．これにより，正しい通信相手からのパケットのみ受け取ることができる．

† **通信内容の暗号化**
送信ノードと受信ノードの間の第三者によって通信内容を傍受，あるいは改竄されないように，送信ノードは，あらかじめ交換しておいた鍵を使ってそのパケットの内容を暗号化し，Encapsulating Security Payload(ESP) [6] と呼ばれるパケットにカプセル化する．これにより，鍵を知らない第三者の不正な操作を防ぐことができる．

アプリケーションやプロトコルで通信内容の暗号化を行なう方法と比較して，IPsec は，通信が暗号化されていることを，アプリケーションやユーザに意識させずに提供できる点，また，アプリケーションやプロトコル自体が暗号化の機能を持っていなくても，暗号化通信の機能を提供できる点で優れている．

ところが，IPsec は IP の機能として提供されるため，もともと IPsec を想定していなかった既存のインターネット環境との親和性には乏しい．しかし，次世代インターネット環境の中心となる Internet Protocol Version 6(IPv6)[7] では，IPsec 機能が必須となるため，近い将来にはインターネット上の全てのノードで IPsec が利用できる．つまり，IPsec は，次世代インターネットにおける安全なネットワーク環境を提供する中心的な技術基盤であると言える．

ところで，IPsec を利用するためには，認証データの計算や通信内容の暗号化が必要となり，ネットワーク通信性能の低下が予測される．

そこで本稿では，IPsec を使用した通信性能を計測し，使用しない場合との比較評価を行なう．また，IPsec がその拡張機能の一つでしかない IPv4 と，IPsec が必須となる IPv6 のそれぞれについて性能評価し，結果を比較する．そして，これらを元に考察を行った．

2 実験内容

今回の実験では，まず基本的な通信性能を調べるために，あらかじめ用意した実験ネットワークで測定した．その後，実際の環境に近い状態での通信性能を調べるために，実際のインターネット環境で測定した．

前者は，無償で公開されている通信性能評価ツールであるDBS[8] とnetperf[9] を利用し，エンド—エンド間での性能測定を行なった．

後者は，UDP を用いた Digital Video ストリーム転送アプリケーションを使い，エンドノードが受信したパケット数とバイト数から性能測定を行なった．

IPsec の鍵は，すべての実験において，自動鍵交換デーモンなどは使わず，手動で設定した．

2.1 DBS による測定

DBS version 1.1.5 を用いて，IPv4 の TCP 転送実験を行なった．

2.1.1 使用した計算機環境

使用した計算機の仕様は送信側，受信側ともに，表 1 のとおりである．

表 1: 計算機の仕様

CPU:	PentiumII 450MHz
Memory:	128MB
NIC:	Intel EtherExpress Pro 100
OS:	FreeBSD 2.2.8 KAME 19990809-stable

2.1.2 性能評価の方針

まず初めに，ユーザ空間においてできるだけ資源を割り当てた場合の，IPsec を使用しない場合とした場合の，TCP のスループット，ジッタ，遅延などを計測した．これにより，実験環境で IPsec を使用しない場合とした場合の，限界速度を推定することができる．

次に，他のプロセスが CPU パワーを消費している場合の，スループットを，同じように，IPsec を使用しない場合とした場合とについて計測した．

2.1.3 実験方法

最初の実験は，計測プログラムにできるだけ資源を割り当て，他の機器の影響がでないように，

- 通信性能評価プログラムの優先度を nice コマンドを使い，最も高くする
- 余計なデーモンプロセスを実行しない
- CPU 時間，メモリ使用量などを無制限にする
- 2 台の計算機をクロスケーブルでつなぐ

という環境で行なった．

次の実験では他のプロセスに CPU パワーを消費させるために，円周率を計算するプログラムを 1 つ実行し，平均負荷が 1.00 に近づいてから，計測を開始した．

どちらの実験でも，送受信バッファは 32768 バイト，メッセージサイズは 8192 バイト，送受信の間隔，待ち時間は 0 秒で，メッセージを 8192 回送信した．

IPsec はトランスポートモードで実行し，AH では HMAC-SHA1，ESP では 3DES-CBC を使用した．

2.1.4 実験結果

スループット

平均スループット (図 1)

IPsec を使用しない	65.18Mbps
AH を使用	26.01Mbps
ESP を使用	10.94Mbps

負荷がかかってる時の平均スループット (図 2)

IPsec を使用しない	66.12Mbps
AH を使用	13.67Mbps
ESP を使用	5.70Mbps

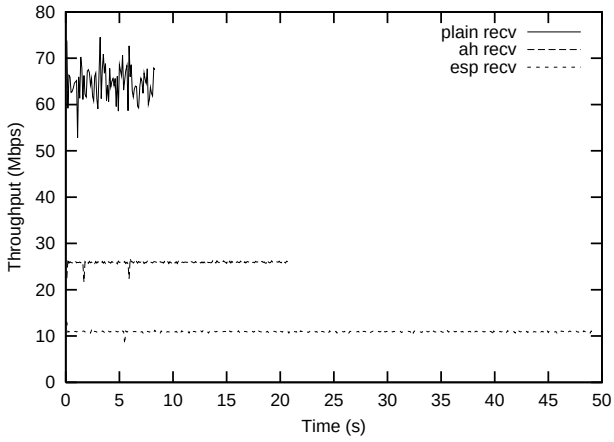

図 1: スループット

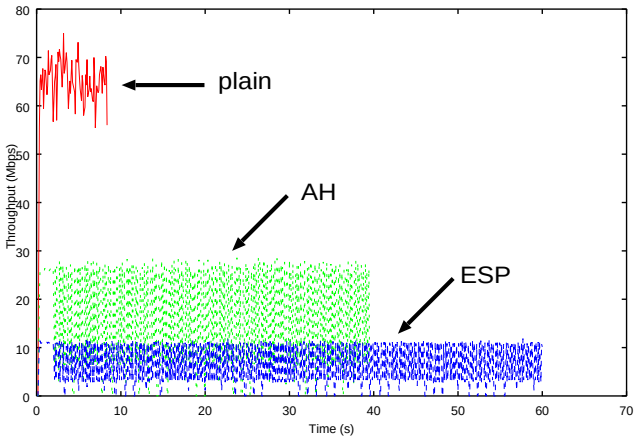

図 2: 負荷がかかっている時のスループット

ジッタ

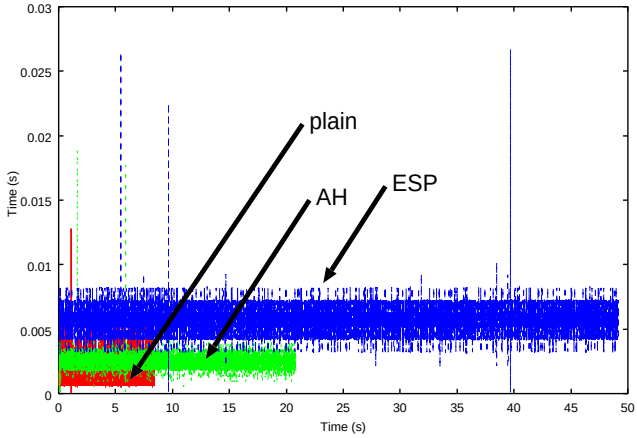

図 3: ジッタ

遅延

平均遅延 (図 4)

IPsec を使用しない	0.015 秒
AH を使用	0.007 秒
ESP を使用	0.003 秒

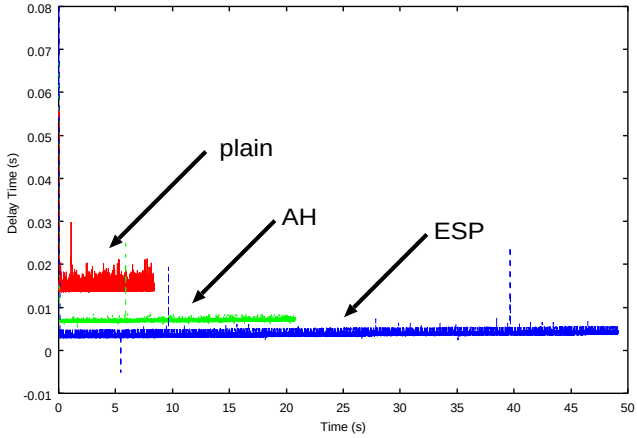

図 4: 遅延

2.1.5 考察

まず初めに，ユーザ空間においてできるだけ資源を割り当てた場合の計測，について考察する．

平均スループットに注目すると，AH を使用した場合，使用しない場合の約 0.4 倍の約 26Mbps であり，ESP を使用した場合は，使用しない場合の約 0.2 倍の約 11Mbps である．これが，使用した計算機の CPU を利用した場合の，IPsec の限界であると推定できる．

図 1 に注目すると，IPsec を使用しない場合は，IPsec を使用した場合と比べて，スループットにばらつきが見られる．これは，IPsec を使用した場合は，暗号化

の計算がボトルネックとなり，スループットが一定してしまうためと考える．

ジッタ (図 3)，遅延 (図 4) も同じように，IPsec を使用した場合の方が，値は小さくなっている．これも IPsec の処理がボトルネックとなり，他の影響が出にくいためであると考える．

次に，高負荷環境での計測について考察する．

初めの測定環境とは反対に，他のプロセスが CPU を大量に使用している環境で測定を行なった．

その結果，IPsec を使用しない場合のスループットは初めの測定と変わらなかったが，IPsec を使用した場合のスループットは，どちらも約半分にまで落ち込んだ．また，図 2 から分かるように，単位時間あたりのスループットにも非常に幅が出ている．

これは，IPsec を使用した場合は，暗号化という大量の計算処理が必要であるが，他のプロセスと CPU 資源を分け合う形になり，処理が遅れたためと推定できる．

2.2 netperf による測定

netperf 2.1pl3 に，IPv6 を利用できるように変更したパッチ [10] を適用し，これを用いて，基本的な通信の性能評価を行なった．

2.2.1 使用した計算機環境

この実験で用いたネットワークの構成を，図 5 に示す．尚，このネットワークは他のトラフィックによる測定への影響を無くすために，完全に独立したネットワークとして構成した．

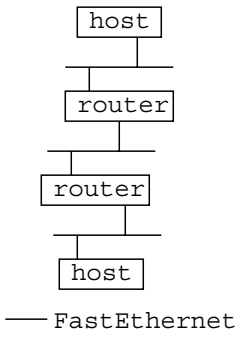

図 5: netperf による性能評価実験ネットワークの構成

ホストの仕様は表 1 と同じであり，ルータの仕様は表 2 のとおりである．

表 2: ルータの仕様

CPU:	PentiumIII 500MHz
Memory:	256MB
NIC:	Intel EtherExpress Pro 100 DEC 21040 PCI Ethernet
OS:	FreeBSD 2.2.8 KAME 19990809-stable

2.2.2 性能評価の方針

通信性能の測定は，次に挙げる 4 つのパターンについて行なった．

- IPsec を使用しない場合
- IPsec の AH のみを使用した場合
- IPsec の ESP のみを使用した場合
- AH と ESP を同時に使用した場合

それぞれのテストは，利用するプロトコルの組合せとして，IPv4 と IPv6，および TCP と UDP について行なった．

TCP と UDP の性能評価は，Netperf が提供している二つの方式に基づいて行なった．一つ目は，二点間のスループットを計測する STREAM テストである．このテストによって，それぞれのパターンで利用可能な帯域幅を測定することができる．二つ目は，送信ホストが要求メッセージを送ってから，受信ホストがそれに対する応答メッセージを送るまでを 1 処理単位とした時の，処理数 (トランザクション数) を計測する REQUEST/RESPONSE テストである．

前者のテストではパラメータとして，ソケットバッファサイズと送信メッセージサイズを変更して測定した．

2.2.3 実験方法

受信ホストで netserver を起動し，送信ホストから以下のそれぞれのパラメータで netperf を起動し，60 秒間計測を行なった．

IPsec はトランスポートモードで実行した．設定は以下の通りである．

NONE	-	IP Security を使用しない
AH	-	AH を使用する (HMAC-SHA1 160bit)
ESP	-	ESP を使用する (3DES-CBC 192bit)
AH/ESP	-	AH と ESP を使用する

2.2.4 実験結果

以下では，TCP を用いた STREAM テスト，UDP を用いた STREAM テスト，そして RE-

QUEST/RESPONSE テストの順で実験結果を示す.

TCP STREAM

この測定で用いたパラメータを表 3 に示す．なお，ソケットサイズの1 と2 は，図 6 と図 7 の凡例の1 と2 に対応する．

表 3: TCP STREAM テスト

パラメータ	
メッセージサイズ	4,096 バイト
ソケットサイズ	**1.** 57,344 バイト
	2. 32,768 バイト
結果 IPv4	
NONE.1	60.86Mbps
NONE.2	60.57Mbps
AH.1	23.10Mbps
AH.2	23.01Mbps
ESP.1	10.87Mbps
ESP.2	10.90Mbps
AH/ESP.1	7.74Mbps
AH/ESP.2	7.75Mbps
結果 IPv6	
NONE.1	58.21Mbps
NONE.2	58.21Mbps
AH.1	24.20Mbps
AH.2	24.43Mbps
ESP.1	10.88Mbps
ESP.2	10.90Mbps
AH/ESP.1	7.65Mbps
AH/ESP.2	7.68Mbps

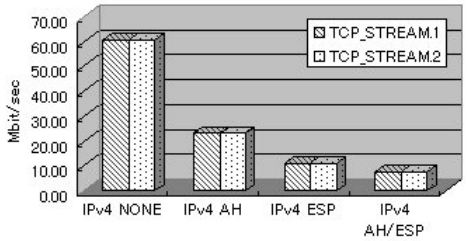

図 6: IPv4 TCP STREAM

図 6 と図 7 は，それぞれ IPv4 と IPv6 上の TCP を用いた STREAM テストの結果を示している．この結果によれば，IPsec を使用していない場合と比較して，IPsec を使用した場合では，AH で約 1/2 程度，ESP で約 1/4 程度，スループットが低下している．しかし，AH と ESP を同時に使用した場合は，ESP を単独で使用した場合と比較してもあまり差は見られなかった．また，IPv4 と IPv6 の違いという点についても，ほとんど性能に差が見られなかった．

UDP STREAM

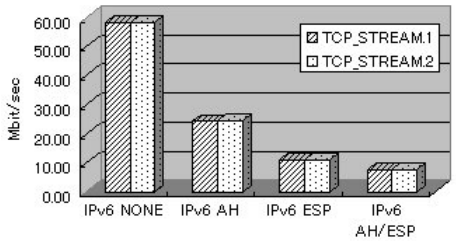

図 7: IPv6 TCP STREAM

この測定で用いたパラメータを表 4 に示す．なお，メッセージサイズの1 と2 は，図 8 と図 9 の凡例の1 と2 に対応する．

表 4: UDP STREAM

パラメータ	
メッセージサイズ	**1.** 4,096 バイト
	2. 1,024 バイト
ソケットサイズ	32,768 バイト
結果 IPv4	
NONE.1	93.98Mbps
NONE.2	93.74Mbps
AH.1	30.73.24Mbps
AH.2	23.83Mbps
ESP.1	11.96Mbps
ESP.2	11.32Mbps
AH/ESP.1	8.98Mbps
AH/ESP.2	8.09Mbps
結果 IPv6	
NONE.1	93.16Mbps
NONE.2	92.37Mbps
AH.1	34.03Mbps
AH.2	25.83Mbps
ESP.1	11.94Mbps
ESP.2	11.32Mbps
AH/ESP.1	9.01Mbps
AH/ESP.2	8.12Mbps

図 8 と図 9 は，それぞれ IPv4 と IPv6 上の UDP を用いた STREAM テストの結果を示している．この結果によれば，IPsec を使用していない場合と比較して，IPsec を使用した場合では，AH で約 1/3 程度，ESP で約 1/9 程度，スループットが低下している．AH を使用している場合，メッセージサイズを増やす (1,024 バイトから 4,096 バイトへ) ことにより，若干のスループットの向上が見られる．また，TCP の STREAM テストと同じように，ESP を単独で使用した場合と，AH と ESP を同時に使用した場合を比較しても，スループットの差はほとんどなかった．また，IPv4 と IPv6 の違いという点についても，TCP STREAM の時と同じように性能の差もほとんどなかった．

REQUEST/RESPONSE

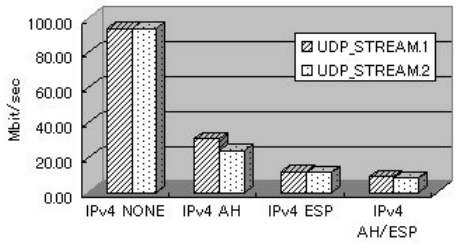

図 8: IPv4 UDP STREAM

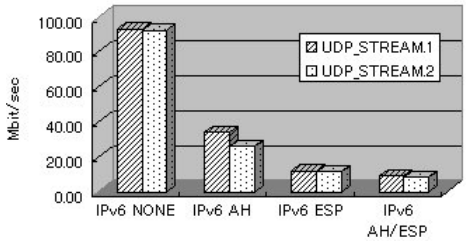

図 9: IPv6 UDP STREAM

この測定で用いたパラメータを表 5 に示す.

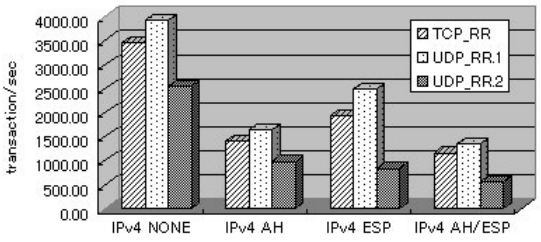

図 10: IPv4 RR

図 10 と図 11 は，それぞれ IPv4 と IPv6 上を用いた REQUEST/RESPONSE テストの結果を示している．この結果によれば，IPsec を使用していない場合と比較して，IPsec を使用した場合は，全体的にトランザクション数が低下している．傾向として，IPsec を使用していない場合も IPsec を使用した場合も，TCP_RR, UDP_RR.1，UDP_RR.2 の間での性能差の比率は同じであった．また，TCP_RR と UDP_RR.1 では，AH を使用した場合よりも ESP を使用した場合の方がトランザクション性能が高い．

AH と ESP を同時に使用した場合のトランザクション性能は，AH を単独で使用した場合とほぼ同じである．ここでも IPv4 と IPv6 による差はほとんど見られない

表 5: REQUEST/RESPONCE テスト

パラメータ Request/Response サイズ	
TCP	1 バイト/1 バイト
UDP 1	1 バイト/1 バイト
UDP 2	516 バイト/4 バイト
結果 IPv4 TCP RR (Transaction/sec)	
NONE	3454.84
AH	1404.26
ESP	1940.77
AH/ESP	1164.21
結果 IPv4 UDP RR	
NONE.1	3929.57
NONE.2	2559.89
AH.1	1650.64
AH.2	988.27
ESP.1	2498.62
ESP.2	839.26
AH/ESP.1	1358.40
AH/ESP.2	575.49
結果 IPv6 TCP RR	
NONE	3586.17
AH	1550.57
ESP	2048.07
AH/ESP	1144.48
結果 IPv6 UDP RR	
NONE.1	3955.94
NONE.2	2551.06
AH.1	1657.83
AH.2	1047.74
ESP.1	2446.02
ESP.2	832.67
AH/ESP.1	1339.44
AH/ESP.2	576.00

2.2.5 考察

STREAM テスト

ここでは netperf による STREAM テストの結果の考察を行なう．以下では，比較のために着目した点を列挙し，それぞれについての考察を述べる．

- IPsec **を使用した場合としない場合**
  IPsec を使用した場合，ペイロードに対して AH のハッシュ化や ESP の暗号化などの処理が必要になるため，IPsec を使用する場合のスループットは，IPsec を使用しない場合に比べて低下する．

- TCP **と** UDP
  TCP の STREAM テストと UDP の STREAM テストの結果を比較すると，UDP の場合の方が，IPsec を使用した場合と使用しない場合のスループットの差が大きい．これは，IPsec を使った場合でも TCP と UDP でのスループットがほぼ同じであることから推測して，IPsec の処理限界は，上位プロトコルに関係なく一定であることが分かる．つまり，STREAM テストは，IPsec を使った場合のスループットの限界に達しているので，

TCP と UDP のどちらの場合もほぼ同じように結果として表れたと考えられる．

しかし，AH を使用した UDP STREAM の場合のみで，メッセージサイズを大きくした場合に，若干スループットが向上している．

このことから，次の二つの点が推測できる．

- UDP STREAM テストでは，AH の性能限界には達していない
- メッセージサイズに比例して IPsec の処理の負荷が高くなるわけではない

- AH **と** ESP
  AH のみを使った場合と ESP のみを使った場合を比較すると，AH のみを使った場合の方が ESP のみを使った場合の約二倍のスループット性能であることが分かった．これは，AH はペイロードをハッシュするだけだが，ESP はペイロード全体を暗号化しなければならないため，ESP の方が負荷が高くなりスループットが低下するからである．また AH と ESP を同時に使用した場合は，ESP によるオーバーヘッドが大きいため，AH を使用することによる性能低下はあまり見られない．

- IPv4 **と** IPv6
  IPv4 と IPv6 の結果を比較すると，IPv4 と IPv6 で同条件の実験を行なった場合，ほぼ同じ程度の性能が出ていることがわかる．これは，IPsec が主に IP ペイロードに対する処理であるため，IPsec を使用しない場合で性能の差がなければ，IPsec を使用した場合にプロトコルによる差は無視して良いと考えられる．

REQUEST/RESPONSE テスト

ここでは netperf による REQUEST/RESPONSE テストの結果の考察を行なう．以下では，比較のために着目した点を列挙し，それぞれについての考察を述べる．

- IPsec **を使用した場合としない場合**
  IPsec を使用した場合，STREAM テストと同じように，IPsec を使用しない場合に比べて性能低下はある程度見られるが，STREAM テストの場合ほと大きな低下ではない．これは，REQUEST/RESPONSE テストでは，STREAM テストにおける一様な連続したトラフィックと異なり，データを受信しそのデータに対して応答するという一連の処理が，トランザクション数を低下させているため，相対的に IPsec による性能低下が 小さくなっているためだと考えられる．

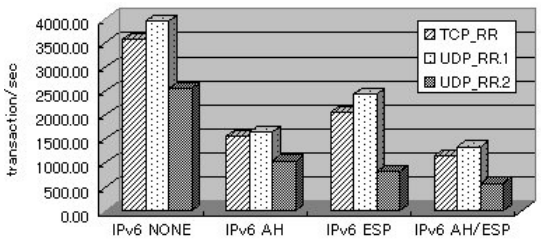

図 11: IPv6 RR

- TCP **と** UDP

  TCP の REQUEST/RESPONSE テストと UDP の REQUEST/RESPONSE テストの結果を比較すると，同じメッセージサイズの場合，UDP の場合の方がトランザクション性能が上がっていることから，REQUEST/RESPONSE テストでは，IPsec による性能限界には達していないと考えられる．

- AH **と** ESP

  STREAM テストの場合と異なり，メッセージサイズが小さい場合は，AH を使用した場合よりも ESP を使用した場合の方が性能が高いという結果が出ている．このことについて，いくつかの理由が考えられるが，現時点では完全に絞り切れていないので，今後の課題としたい．ただ，実際にネットワークを利用する際には，小さなサイズのメッセージをやりとりすることは多く，必ずしも ESP を使った場合の方が AH を使った場合より性能の点で, 劣るわけではないことがわかる．

  AH と ESP を同時に使用した場合の性能が，AH を単独で使用した場合とほぼ同じであることから，REQUEST/RESPONSE テストでは，ESP ではなく AH の処理がボトルネックとなっていることが分かる．

- IPv4 **と** IPv6

  IPv4 と IPv6 のトランザクション性能を比較すると，STREAM テストと同じように，IPsec の使用の有無にかかわらず，同程度の性能が得られることが分かった．

2.3 IPsec 上での DV 送信実験

実ネットワーク環境で，IPsec を使用した場合の性能評価を行なうため，IPsec を使用した IPv6 上での Digital Video(DV) の送受信実験を行なった．なお，今回はエンド―エンド間での性能評価に着目し，IPsec のトランスポートモードのみ実験した．

2.3.1 使用した計算機環境

今回の実験で用いたネットワーク構成を，図 12 に示す．なお，本実験は，東京大学情報基盤センターと慶應義塾大学湘南藤沢キャンパスの間で，実際に運用されているネットワークを経由して行なわれた．

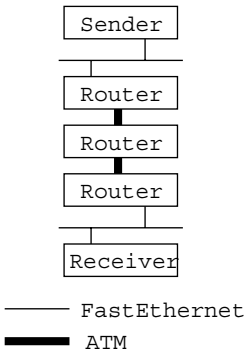

図 12: DV を使った実験ネットワークの構成

またこの実験で使用した計算機の仕様を表 6 と表 7 に示す．

表 6: 送信側ホストの仕様

CPU:	PentiumII 400MHz
Memory:	64MB
NIC:	Intel EtherExpress Pro 100
OS:	FreeBSD 3.2
	KAME 19990809-stable
Software:	DVTS-0.0.9

表 7: 受信側ホストの仕様

CPU:	PentiumII 450MHz
Memory:	128MB
NIC:	DEC 21040 PCI Ethernet
OS:	FreeBSD 3.2
	KAME 199900705-snap
Software:	DVTS-0.0.9

2.3.2 性能評価の方針

実トラフィックを想定した DV ストリームによって，IPsec を使用した場合にどの程度の性能が出るかを計測する．また二通りの暗号アルゴリズムを使用し，アルゴリズムによるスループットの差にも注目する．

2.3.3 実験方法

DV 機器を接続するため，送信ホスト，受信ホストの両方に IEEE1394 インターフェイスを取り付けた．そして，送信側ホストには DV カメラを，受信側ホストには DV デッキを接続し，送信ホストの DV カメラで撮影している映像を受信ホストに送信した．受信ホストでは，受信した映像を DV デッキを介しモニタに出力させる．

実験に利用した FreeBSD は，そのままでは IEEE1394 インターフェイスを使用できない．そこで，FreeBSD に組み込む専用のドライバと DV の送信・受信用のアプリケーションである DVTS[11] というパッケージを使用した．

DV の送信ホスト側のアプリケーションのパラメータとして，フレームレートを 1/10 に設定 (ただし，音声には変更を加えなかった) して行なった．この間，受信ホスト側で 1 秒毎の受信パケット数，受信バイト数を計測した．この実験に関する設定は表 8 の通りである．

表 8: DV 実験時のパラメータ

NONE	-	IP Security を使用しない
AH 1	-	AH を使用する (HMAC-SHA1 160bit)
AH 2	-	AH を使用する (KEYED-SHA1 160bit)
ESP 1	-	ESP を使用する (3DEC-CBC 192bit)
ESP 2	-	ESP を使用する (BLOWFISH-CBC 192bit)
AH/ESP 1	-	AH 1 と ESP 1 を使用する
AH/ESP 2	-	AH 2 と ESP 2 を使用する

2.3.4 実験結果

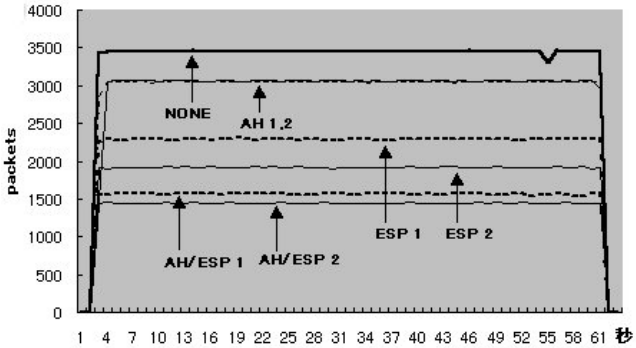

図 13: DV over IPsec (packets)

IPsec を使用しない場合で，約 13Mbps(約 3000 パケット/秒) ほどのスループットが得られた．これに対

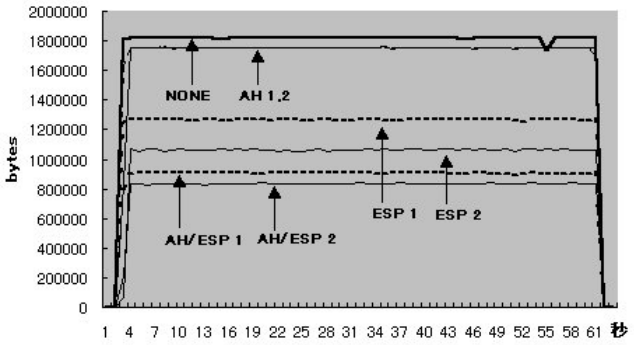

図 14: DV over IPsec (bytes)

して，AH を使用した場合は，どちらのアルゴリズムでも，使用しない場合とほとんど差がない程度である．一方，ESP を使用した場合は，明らかにスループットが落ちており，使用しない場合と比較して約 30%ほど低下していることが分かる．また，ESP.1 と ESP.2 に注目すると，BLOWFISH-CBC を使用した場合は，3DES-CBC を使用した場合に比べて 20%ほどスループットが低下していることがわかる．AH/ESP.1 と AH/ESP.2 のスループットには，ESP1 と ESP2 の差と同じ程度の差が見られる．

IPsec を使用しない場合，トラフィックはほぼ一様に流れているが，IPsec を使用した場合 (特に ESP を使用した場合) は，スループットに若干のばらつきが観測された．

2.3.5 考察

ここでは，DV による実トラフィックの性能測定結果の考察を行なう．以下では，比較のために着目した点を列挙し，それぞれについての考察を述べる．

- netperf と DV 送信実験
  DV の通信は UDP で行なわれるため，まず初めに netperf の UDP IPv6 での STREAM テスト (メッセージサイズ 4,096 バイト) との比較をする．
  - AH
    netperf による性能測定では AH を使用した場合，約 65%程度のスループットの低下がみられたが，DV の送信実験では約 5%程度しか低下していない．
    これは，netperf で AH を使用した場合のスループットが 33.94Mbps，DV 送信実験で AH を使用しない場合のスループットが約 13.3Mbps であることから考えて，DV では AH によるオーバーヘッドではなく，ネットワークの帯域がボトルネックになっているためであると考えられる．つまり，ネットワークの帯域を広げることで，AH を使用した場合の性能も上がることが予測される．
  - ESP
    ESP を使用した場合は，netperf での測定結果と同じように使用しない場合に比べて，大きくスループットが低下している．netperf で ESP を使用した場合のスループットが約 11Mbps，DV 送信実験で ESP を使用した場合のスループットが約 9.6Mbps である．つまり，IPsec の処理性能として少なくとも 11Mbps 程度の性能が期待できるはずであるが，DV 送信実験ではそこまで達していないことから，この場合は ESP がボトルネックとなっていることがわかる．
    また，DV のペイロードサイズが 500 バイト，netperf による測定の際のメッセージサイズが 1,024 バイトであるが，スループットには大きな差がないことから，IPsec の処理にペイロードサイズは影響しないと考えられる．
- AH と ESP
  AH を使用した場合と ESP を使用した場合を比較すると，明らかに AH を使用した場合の方がスループットが性能が高い．
  これには，次の二つのような理由が考えられる．
  - DV が一様な連続した片方向のトラフィックであるため，REQUEST/RESPONSE テストのように他のパケット処理に時間がかからない．
  - netperf での測定結果と同様に，AH ではペイロードのハッシュ化だけでよいが，ESP ではペイロードの暗号化を行なわなければならないため，パケット処理の負荷が高い．
- 暗号化アルゴリズム
  ESP を使用した場合に，暗号アルゴリズムの変更によるスループットの変化について見てみる．同じ長さの鍵を使っている 3DES-CBC と BLOWFISH-CBC で比較すると，実際のスループットには 20%程度の差が見られる．アルゴリズム以外の条件は同じなので，このスループットの差は，アルゴリズムの性能の差によるものだと言える．
- AH/ESP の結果
  AH/ESP.1 と AH/ESP.2 を比較すると，次の二つのことが分かる．
  - AH.1 と AH.2 のスループットはほぼ同じである．

– AH/ESP.1 と AH/ESP.2 の差は ESP.1 と ESP.2 の差とほぼ同じである．

このことから，AH と ESP を同時に使用した場合のボトルネックは ESP であると言える．

- **スループットのばらつき**
  IPsec を使用しない場合，AH のみを使用した場合はスループットはほぼ一様であるが，ESP を使用した場合には若干のばらつきがみられる．これはホストにおける ESP の処理が，DV のトラヒックに追いついていないためだと思われる．

- **画質**

  受信した画質について観察すると，AH を使用した場合動きのある場面の時に若干引きずったような画面になった．しかし実用上ほとんど問題ない程度の画質であった．また，ESP を使用した場合は，この「引きずった」感じがひどくなり，動きの大きい場面では対象を認識することが困難であった．

  ただし，今回は帯域が十分でなかったため，IPsec を使用しない場合の画質もあまりよくなかった．

今回は送信者，受信者が最終的に FastEthernet につながっていたため，IPsec を使用しない場合でもスループットがあまりでなかった．そのため，IPsec によるオーバーヘッドなのか，ネットワークの帯域の問題なのかを完全に明確にすることができなかった．

3 まとめ

本実験では，初めにネットワークのベンチマークソフトにより，IPsec を使用した場合の性能測定を行なった．続いて，これから多く利用されるようになるであろう大容量トラフィックを発生するアプリケーションとして DV を選択し，大量のトラフィックが流れる実ネットワークで，IPsec を使用した場合の性能測定を行なった．

結果としては，次の 4 つが得られた．

- IPsec を利用した場合，スループットは低下した
- 必ずしも AH を使用した場合の方が，ESP を使用した場合よりもスループットが出るとは限らない
- IPv4 と IPv6 の IPsec の処理には，性能の差はみられない
- 暗号アルゴリズムによってパフォーマンスに差があらわれる

今後の課題として，次の 4 つを挙げる．

- アルゴリズムは同一のものを使用し，鍵長を変化させた時の IPsec の性能評価
- 同一の IPsec の設定のもとでペイロードサイズを連続的に変化させた場合のスループットの計測．
- 同一の IPsec の設定のもとで MTU を連続的に変化させた場合のスループットの計測．
- より広い帯域で IPsec を使用した場合の性能測定

参考文献

[1] RFC791 「Internet Protocol J. Postel ,1981/9

[2] Internet-Draft「SSH Protocol Architecture」T. Ylonen, T. Kivinen, M. Saarinen, T. Rinne, S. Lehtinen, 1999/7

[3] Internet-Draft 「The SSL Protocol Version 3.0」Alan O. Freier,Philip Karlton,Paul C. Kocher,1996/11

[4] RFC2401 「Security Architecture for the Internet Protocol」S. Kent,R. Atkinson,1998/11

[5] RFC2402 「IP Authentication Header」 S. Kent,R. Atkinson,1998/11

[6] RFC2406 「IP Encapsulating Security Payload (ESP)」S. Kent,R. Atkinson,1998/11

[7] RFC2460 「Internet Protocol version 6 Specification」S. Deering,R. Hinden,1998/12

[8] http://shika.aist-nara.ac.jp/member/yukio-m/dbs/index-j.html「DBS : A TCP Benchmark Tool」

[9] URL: http://www.netperf.org 「Netperf Homepage」

[10] KAME Project, ftp site, ftp://ftp.kame.net/pub/kame/misc/netperf-21pl3-19990824.diff.gz

[11] URL: http://www.sfc.wide.ad.jp/DVTS/