IP通信拡散法を用いたVPN装置の実装と性能評価

東京電機大学 理工学部 情報システム工学科
有泉 徹也 寺西 貴 横山 雄一 桧垣 博章

前橋工科大学 工学部 情報工学科
遠山 宏明

TCP/IP インターネットを介して、物理的に離れた複数のLANを論理的に接続するVPN(Virtual Private Network)への要求が高まっている。ここでは、LANを相互接続するインターネットにおいて、IPデータグラムを盗聴されることなく、安全に配送することが求められる。本論文では、暗号技術と組み合わせてより安全な通信路を実現するためのIP 通信拡散手法と、それを利用したVPN装置の実装について述べる。IP 通信拡散手法は、IP の機能であるソースルーティングを用いずに、動的に決定する複数の経路を用いてIPデータグラム群を配送する手法である。本手法をVPN装置として実装するためには、異なるLAN間を配送されるIPデータグラムについては、暗号通信と拡散通信を組み合わせることが必要である。一方、LAN内のコンピュータとインターネット上のコンピュータとの間を配送されるIPデータグラムについては、アドレス変換が必要とされる。これらの機能をLinux上に実装した。また、IP 通信拡散手法の経路制御において中継点として使用されるルータの処理オーバヘッドを評価した。

Performance Evaluation of Dynamic Multiple-Route IP Transmissions

Tetsuya Ariizumi Takashi Teranishi Yuichi Yokoyama Hiroaki Higaki
Department of Computers and Systems Engineering
Tokyo Denki University

Hiroaki Toyama
Department of Information Engineering
Maebashi Institute of Technology

For achieving secure communication against snooping, encryption is applied in the TCP/IP Internet. It is based on that too much computation is required for snooper to get an original data from an encrypted data. Hence, the higher performing computers are developed, the more complex encryption algorithms have to be designed and implemented. This paper proposes a novel methodology that IP datagrams are transmitted through multiple routes determined dynamically. Since no additional function is introduced in routers, it is highly applicable. Finally, we evaluate overhead on the router in which ICMP encapsulated packets are processed and the result shows the proposed method is reasonably implemented in the Internet.

1 背景と目的

電子メールやWWW (World Wide Web) サービスの普及により、企業や個人のインターネット利用環境が広く普及している。また、ISDN、ADSL、CATV、光ファイバーの普及により、アクセスネットワークの高速広帯域化が著しく、これにともなってISP (Internet Service Provider) のバックボーンネットワークの高速広帯域化も進み、マルチメディアデータの実時間配送など、サービスの高度化、多様化が可能となっている。このように、企業活動、社会活動のインフラストラクチャとしてのインターネットの地位が高まるなかで、第三者に情報が遺漏することなく、安全にコンピュータ間で情報を交換するためのネットワークセキュリティ技術への要求が高まっている。インターネットを用いて交換しているデータを悪意のある第三者が不当に入手することを避けるために、暗号通信が広く利用されている。TCP/IP インターネットにおけるネットワーク層プロトコルであるIP (Internet Protocol) [8] には、暗号通信の機能は含まれていない。そこで、アプリケーションにおいてデータの暗号化を行なったり、IPsec [5,14] を用いてIPデータグラムのデータ部を暗号化するなどの手法が採られている。暗号通信を実現するために、様々な暗号アルゴリズム [4,6,15,16] が提案されているが、いずれの方法も、復号鍵を持たないで暗号文から平文を入手する (あるいは復号鍵を推定する) ために必要とされる計算量の大きさのみに、その安全性の根拠を置いている。コンピュータの計算能力の向上は著しく、悪意のある第三者が、ある暗号アルゴリズムを用いて作られた暗号文から平文を入手するために十分な計算能力を持つコンピュータを入手することは可能となり得る。通信の安全性を保つためには、新しい暗号アルゴリズムを導入しなければならない。例えば、DES [16] は現在では十分に安全な暗号アルゴリズムであると言うことはできず [15]、3-DES [4] やIDEA [6] といった新しいアルゴリズムへの移行が必要となっている。しかし、多数のコンピュータが相互接続されているインターネット環境においては、新しい機能をすべてのコンピュータに頻繁に導入することは困難である。したがって、コンピュータの計算能力の向上とは無関係に、暗号通信をより頑強にする手法の導入が求められている。

本論文では、ひとつのデータを配送するための複数のIPデータグラムを複数の経路を用いて配送することによって、盗聴者がデータの全体を得ることを困難にする手法を提案し、その実現プロトコルを設計する。ここで、複数の経路を固定的に定めるのではなく、通信要求が発生するごとに動的に決定することによって、盗聴者がIPデータグラムの通過するルータを特定することを困難にしている。また、提案手法を実現するためには、送信元コンピュータと送信先コンピュータに本論文で提案する機能が導入されることのみが必要であり、インターネットのルータには、特殊な機能を導入する必要がない点で適用性に優れている。提案手法をLinuxオペレーティングシステムがインストールされたパーソナルコンピュータに実装する方法について論じる。特に、VPN装置に実装する場合、暗号通信を実現するIPsec、アドレス変換を実現するIPマスカレード(iptable)と共存する必要がある。本論文では、拡散通信と暗号通信(VPN装置間通信)の機能をアプリケーションプロセスとして実装し、OSカーネルに実装されたアドレス変換機能との複合実装方法について述べる。さらに、IP通信拡散手法では、経路を拡散させるための中継点として使用するルータの処理オーバヘッドが、通常のIPデータグラム配送に比べて大きくなることが考えられる。そこで、このオーバヘッドを評価し、インターネット環境での適用可能性について議論する。

2 従来手法

TCP/IPインターネットにおけるセキュリティへの脅威には、組織LANへの攻撃と組織LAN間の通信への攻撃がある。前者の解決策としてファイアウォールがある。これは、インターネットと組織LANとの境界にファイアウォールの機能を持つルータ装置を配置することによって実現される。一方、後者の解決策としてVPN (Virtual Private Network) がある。VPNは、インターネットに接続されている複数の組織LANをインターネットを介して論理的に接続する。アプリケーションに対しては、異なる組織LANに属するコンピュータ間の通信を同一LAN内の通信と同等に見せることができる。このとき、各組織LAN間は専用線ではなく、インターネットを用いることから、組織LAN間の通信の安全性を確保することが必要である。これは、IPsec [5,14] などを利用した暗号通信を用いることで実現される。暗号通信は、送信元と送信先で共通の秘密情報(鍵)を持つことを前提とする秘密鍵暗号と秘密情報を持つことを前提としない公開鍵暗号とがある。前者には、DES [16]、IDEA [6] 等がある。また、後者には、RSA [11,12]、Diffie-Hellman [1]、Merkle-Hellman [18] 等がある。暗号通信は、暗号文を入手した盗聴者であっても、そこから平文を入手するために必要な計算を、現在のコンピュータ技術では十分短時間には実行できないことに安全性の根拠を置いている。したがって、コンピュータの計算能力の向上によって、使用されている暗号通信技術は陳腐化することになる。本論文で提案するIP通信拡散手法は、この暗号通信の安全性を、コンピュータの計算能力の向上に無関係に補完する技術である。ここでは、あるデータを配送するIPデータグラム群を複数の経路を用いて配送することによって、データを盗聴するのに十分なIPデータグラムの入手を困難にする。

3 提案手法

3.1 IP通信拡散手法

TCP/IPインターネットに接続された2台のコンピュータ c_s と c_d との間で、悪意のある第三者 M (以下では盗聴者とよぶ) にデータを盗聴されることなく安全に通信する方法として、本論文では、IP通信拡散手法を提案する。IP通信拡散手法では、c_s から c_d へデータ D を配送するためのIPデータグラム群 $G_D = \{IP_0, \ldots, IP_{n-1}\}$ を、N 個のサブグループ $SG^i_D \subset G_D (i = 0, \ldots, N-1)$ (ただし、$\cup_i SG^i_D = G_D$ かつ $\forall i \neq \forall i', SG^i_D \cap SG^{i'}_D = \emptyset$) に分割する。また、$c_s$ は、c_s から c_d への N 個の経路 $r^i_{\langle s,d \rangle} = \langle c^i_0 = c_s, c^i_1, \ldots, c^i_{l(i)-1}, c^i_{l(i)} = c_d \rangle$ $(i = 0, \ldots, N-1)$ を決定する。そして、SG^i_D に属するIPデータグラムを $r^i_{\langle s,d \rangle}$ を用いて配送する。これによって、M が D を配送するための G_D のすべてを入手するためには、N 個の経路すべてを監視しなければならない。すなわち、ルータ $\forall i, 0 < \exists k(i) < l(i), c^i_{k(i)}$ もしくは通信路 $\forall i, 0 \leq \exists k(i) < l(i), \langle c^i_{k(i)}, c^i_{k(i)+1} \rangle$ において、SG^i_D に属するIPデータグラムをすべて入手しなければならない。特に、c_s に存在するアプリケーションプロセス AP_s から c_d に存在するアプリケーションプロセス AP_d へ渡されるデータ D_{orig} の暗号化データ $D = encrypt(D_{orig})$ が配送される場合には、SG^i_D の分割方法によって、D の獲得をより困難にすることも可能である。

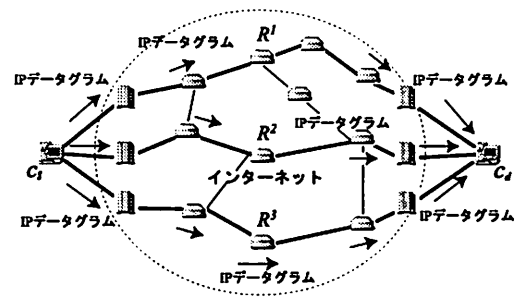

図 1: IP通信拡散における中継ルータ

ここで、M によるデータ入手を困難にするためには、以下の条件を満たすことが求められる。

[要求条件]

- $r^i_{\langle s,d \rangle}$ を M が事前に入手することが不可能 (困難) である。
- $r^i_{\langle s,d \rangle}, r^{i'}_{\langle s,d \rangle}$ に共通に含まれるルータが存在しない (少ない)。□

事前に M が $r^i_{\langle s,d \rangle}$ $(i = 0, \ldots, N-1)$ を知ることが可能であるならば、それぞれの経路上のルータ $c^i_{m(i)} \in r^i_{\langle s,d \rangle}$ あるいは通信路 $\langle c^i_{m(i)}, c^i_{m(i)+1} \rangle$ (ただし、$0 \leq m(i) < l(i)$) に盗聴者 M_i を配置することによって、D を入手することが可能である。これを回避するために、IP通信拡散手法においては、データの配送要求が発生するごとにオンデマンドで経路を探索し、決定する。このとき、経路探索手続きにランダムネスを入れることによって、M が M_i の配置位置を決定すること

を困難にしている。

また、2つの経路 $r^i_{\langle s,d\rangle}$, $r^{i'}_{\langle s,d\rangle}$ $(0 \le i < i' < N)$ に共通のルータ $\exists cc_{\{i,i'\}} \in r^i_{\langle s,d\rangle} \cap r^{i'}_{\langle s,d\rangle}$ (ただし $cc_{\{i,i'\}} \neq c_s$ かつ $cc_{\{i,i'\}} \neq c_d$) が存在するならば、$cc_{\{i,i'\}}$ あるいはこれに直接接続する共通の通信路に M_i を配置することによって D を入手する可能性が高くなる。最も極端な場合として、ルータ $\exists cc \in \cap_i r^i_{\langle s,d\rangle}$ (ただし、$cc \neq c_s$ かつ $cc \neq c_d$) が存在するならば、盗聴者 M_i が cc もしくはこれに接続する共通の通信路に配置された場合、D を入手することが可能となる。この問題を回避するためには、経路拡散の度合 (以下では、拡散度とよぶ) を大きくすればよいと考えられる。すなわち、c_s から c_d への最短経路 (一般的にルーティングテーブルに従って配送される経路はこの経路である) からより離れた複数の経路を選択することで、選択の自由度が大きくなり、共通のルータを含む可能性が低下する。しかし、拡散度を大きくすると、経路長 $l(i)$ が大きくなることが一般的に成立する。各IPデータグラムは独立に配送されるが、上位層のプロトコルを介してアプリケーションプロセス AP_s と AP_d との間でデータ D_{orig} を配送するために要する時間は、最も遅延したIPデータグラムの配送時間によって支配される。すなわち、$\max_i l(i)$ をより小さく抑えることが要求される。

3.2 中継ルータ決定方法

IPデータグラム群 $G_D = \{IP_0, \ldots, IP_{n-1}\}$ を、送信元コンピュータ c_s から送信先コンピュータ c_d まで複数の経路 $r^i_{\langle s,d\rangle}$ を用いて配送するための方法を考える。IPv4には、ソースルーティングの機能がある [8]。これは、オプション機能として定義されており、c_s において、送信するIPデータグラムのヘッダのオプション部に中継点のルータのIPアドレスを列挙することによって、このIPデータグラムをそれらのルータを順番に通過して、c_d へと配送する。しかし、IPヘッダの最大サイズは60バイト (ヘッダ長を格納するフィールドのサイズは4ビットであり、ここには4バイトを単位とした値を格納する) に制限されており、必修フィールドのサイズが20バイトであることから、オプションの最大サイズは40バイトであり、ソースルーティングのための中継ルータを最大9個 (ソースルーティングオプションの固定フィールドに3バイトを要し、IPアドレスは4バイト長である) しか格納することができない。IPデータグラムを中継するルータのアドレスのすべてをヘッダ部に列挙する専用プロトコルを導入し、ソースルーティングを実現する方法が考えられる。しかし、IP通信拡散手法を実現するためにこのような方法を導入するには、インターネットに存在するすべてのルータが新しいプロトコルに従ってIPデータグラムを処理することが必要となる。このように、インターネットに対して変更を加えることなく、既存のルータが持つ機能の範囲内で提案手法を実現することが必要である。

[要求条件]

- IP通信拡散手法を適用するための特殊な機能を中継ルータに導入することを前提としない。□

本論文では、送信元コンピュータが中継ルータを1つだけ指定することとする。すなわち、c_s は、D を配送するために $G_D = \{IP_0, \ldots, IP_{n-1}\}$ を SG^i_D $(i = 0, \ldots, N-1)$ に分割するとともに、N 個の中継ルータ R^i を決定する。そして、各 SG^i_D に属するIPデータグラムを R_i を含む経路 $r^i_{\langle s,d\rangle} = \langle c^i_0 = c_s, c^i_1, \ldots, c^i_{r(i)} = R_i, \ldots, c^i_{l(i)-1}, c^i_{l(i)} = c_d\rangle$ を用いて配送する。ここで、c_s から R_i までの $r^i_{\langle s,d\rangle}$ の部分経路 $\langle c^i_0 = c_s, c^i_1, \ldots, c^i_{r(i)-1}, c^i_{r(i)} = R_i\rangle$ および R_i から c_d までの $r^i_{\langle s,d\rangle}$ の部分経路 $\langle c^i_{r(i)} = R_i, c^i_{r(i)+1}, \ldots, c^i_{l(i)-1}, c^i_{l(i)} = c_d\rangle$ に含まれるルータ $c^i_1, \ldots, c^i_{r(i)-1}$ および $c^i_{r(i)+1}, \ldots, c^i_{l(i)-1}$ は、それぞれ R_i および c_d を送信先とするルーティングテーブルのエントリを参照することによって、各ルータが決定する。

[中継ルータの決定]

1. c_s は、c_d までのホップ数 $hop_{\langle s,d\rangle}$ を以下の手順を用いて測定する。なお、このホップ数が c_s のキャッシュに保存されている場合には、その値を用いる。
   - 1-1. c_s は、送信元と送信先をそれぞれ c_s と c_d、TTLの初期値を T_{init} としたホップ数測定要求メッセージ *hreq* を送信する。
   - 1-2. c_d は、受信した *hreq* のTTL値 T_{obsv} を得る。
   - 1-3. c_d は、送信元と送信先をそれぞれ c_d と c_s とし、T_{obsv} をデータ部に含むホップ数測定応答メッセージ *hrep* を送信する。
   - 1-4. c_s は、*hrep* を受信すると、c_s から c_d までのホップ数 $hop_{\langle s,d\rangle} = T_{init} - T_{obsv}$ を得る。
2. c_s は、ルータのルーティングテーブルに従って配送された場合の c_d までの経路のホップ数 $hop_{\langle s,d\rangle}$ に対して、最適な拡散ホップ数 $hop_{\langle s,m\rangle} = dhop(hop_{\langle s,d\rangle})$ を求める。
3. c_s は、32ビットの乱数値を N 個生成することにより、仮想目標アドレス $vadd^i$ $(i = 0, \ldots, N-1)$ を得る。
4. c_s は、送信元を c_s、送信先を $vadd^i$、TTLを $hop_{\langle s,m\rangle}$、上位プロトコルを未定義のプロトコルとする中継ルータ検出のためのIPデータグラム *mreq* を送信する。
5. c_s が *mreq* に対応するICMPメッセージを受信する。
   - 5-1. これがICMP時間切れメッセージであるならば、このICMPメッセージの送信元を中継ルータ R^i とする。
   - 5-2. これがICMP到達不可能メッセージであるならば、$vadd^i$ を再生成し、4.へ戻る。
6. c_s は、自身から $hop_{\langle s,m\rangle}$ ホップだけ離れた N 個の中継ルータ R^i $(i = 0, \ldots, N-1)$ を得る。□

3.3 データ配送方法

送信元コンピュータ c_s から送信されるデータ D のためのIPパケット群 G_D を分割した N 個のサブグループ SG^i_D $(i = 0, \ldots, N-1)$ のそれぞれを、中継ルータ R^i を経由して送信先コンピュータ c_d に配送する方法について論じる。IPv4には、オプションとしてソースルーティングが定められている。経路上のすべてのルータを指定するストリクトソースルーティングは、3.2節で述べたように、IPヘッダの最大サイズの制約のために使用することは難しい。しかし、中継ルータを1つだけ指定する方法であれば、ルースソースルーティングを用いる

ことにより実現が可能である。ところが、配送経路を指定したIPデータグラムは、DoS(Denial of Service)攻撃のためのIPデータグラムの配送や、悪意のあるデータを含んだIPデータグラムの送信元を偽るための踏台攻撃に利用される[7]。そのため、現在利用されている多くのルータでは、ソースルーティングされたIPデータグラムを受信してもそれを転送することはなく、ただちに破棄するように設定されている。

ルータが持つ機能のうち、受信したデータをそのまま送信するものとして、ICMPエコー[9]がある。ICMPエコー要求メッセージを受信したコンピュータ(ルータを含む)は、エコー要求メッセージの送信元コンピュータに対して、ICMPエコー応答メッセージを送信する。このとき、エコー要求メッセージに含まれるデータは、エコー応答メッセージにコピーされる。本論文では、このICMPエコーを用いて、c_sからルータR^iを経由してc_dへとIPデータグラムを配送することを実現する。

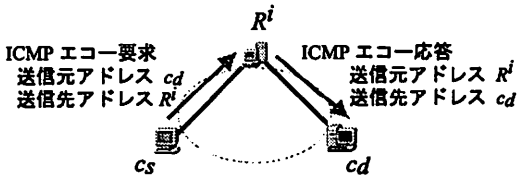

図2: ICMPエコーを用いた経路制御

[IPデータグラム配送]

1. c_sは、送信元をc_s、送信先をc_dとするIPデータグラムIP_{real}を作成する。
2. c_sは、IP_{real}をデータ部に含むICMPエコー要求メッセージ$ereq$を作成する。
3. c_sは、$ereq$をデータ部に含むIPデータグラム$IP_{caps}(ereq)$を作成し、送信する。このとき、送信元をc_d、送信先をR^iとする。
4. $IP_{caps}(ereq)$を受信した各ルータは、ルーティングテーブルを参照し、このIPデータグラムをR^iへと配送する。
5. R^iは、$IP_{caps}(ereq)$を受信すると、対応するICMPエコー応答メッセージ$erep$を作成する。ここで、$erep$のデータ部には、$ereq$に含まれるデータ、すなわちIP_{real}がコピーされる。
6. R^iは、$erep$をデータ部に含むIPデータグラム$IP_{caps}(erep)$を作成し、送信する。このとき、送信元はR^i、送信先はc_dとなる。
7. $IP_{caps}(erep)$を受信した各ルータは、ルーティングテーブルを参照し、このIPデータグラムをc_dへと配送する。
8. c_dは、$IP_{caps}(erep)$を受信すると、そこからIP_{real}を取り出す。□

4 VPN装置の実装

4.1 拡散通信の実装

3章で述べたIP通信拡散手法により、TCP/IPインターネットに接続された複数のLAN L^i $(i = 0, \ldots, P-1)$を論理的に接続し、ひとつのLANとしてアプリケーションに提供するVPNを実現することができる。それぞれのLAN L^iとインターネットとは、VPN装置v^iによって接続されている。v^iには、少なくとも1つのL^iへのインタフェースと少なくとも1つのインターネットへのインタフェースが存在する。インターネットへの接続が複数存在する場合もある。例えば、複数のISP(Internet Service Provider)と契約する場合がこれにあたる。2つのLAN L^iとL^jとの間は、一般的には、v^iとv^jとの間のトンネリングを利用することによって接続する。本章では、IP通信拡散手法を用い、v^iとv^jとの間に複数の経路を動的に決定し、それを用いてIPデータグラム群を配送することによって、盗聴が困難な通信環境を実現するプロトタイプシステムの構築について述べる。VPN間の拡散通信機能は、Linuxオペレーティングシステムがインストールされた PC上のアプリケーションプログラムとして実装される。VPN装置におけるTCP/IP通信には、UDPソケット、Rawソケット、Packetソケットを用いる。なお、以下では、LAN L^iに接続されたコンピュータC_sからLAN L^jに接続されたコンピュータC_dへIPデータグラム群G_Dを配送する場合を例として説明する。

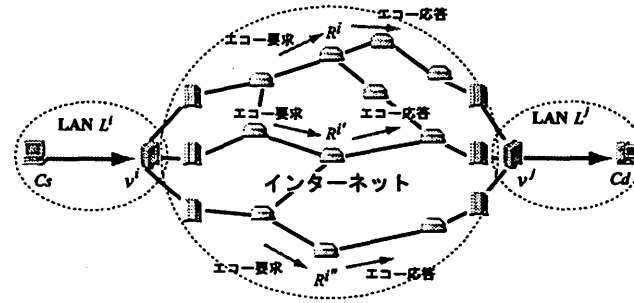

図3: IP通信拡散手法を用いたVPN間通信

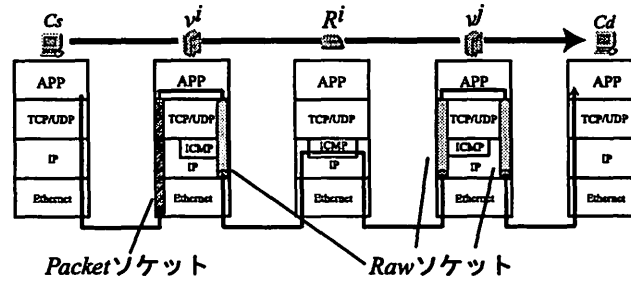

図4: ソケットインタフェースを用いた実装

[C_s–v^i間(L^i内)の配送]

1. G_Dに含まれるIPデータグラムは、送信元をC_s、送信先をC_dとして、C_sから送信される。
2. L^i内のルータは、1.で送信されたIPデータグラムを受信すると、ルーティングテーブルを参照し、このIPデータグラムをv^iへと配送する。
3. v^iは、このIPデータグラムをPacketソケットを通して受信する。Packetソケットを用いることによって、v^iを送信先としないIPデータグラムの全体をアプリケーションで処理することが可能となる。□

[v^i–v^j間(インターネット内)の配送]

1. C_sからC_dへと配送されるIPデータグラム群を受信したVPN装置v^iは、3.2節で述べた方法によりN個の中継ルータを決定する。

1-1. v^i と v^j との間のホップ数の測定には、UDP を用いる。各 VPN 装置では、定められたポート番号でホップ数測定要求のための UDP メッセージを受信する。このとき、送信側では TTL の初期値をアプリケーションで設定し、受信側では TTL の現在値をアプリケーションで利用することから、Raw ソケットを用いる。なお、ホップ数測定応答のための UDP メッセージは、UDP ソケットを用いて送受信する。

1-2. N 個の中継ルータを決定するための *mreq* メッセージは、TTL の初期値をアプリケーションで設定するため、Raw ソケットを用いて送信する。一方、*mrep* メッセージは、ICMP のヘッダを参照する必要があること、送信元の IP アドレスを取得する必要があることから、Raw ソケットを用いて受信する。

2. 1. で受信した IP データグラムを ICMP エコー要求メッセージにカプセル化して R^i へ送信する v_i は、IP データグラムの送信元を v^j とすることから、Raw ソケットを使用する。

3. 2. で送信された ICMP エコー要求メッセージに対応して、R^i が送信した ICMP エコー応答メッセージを受信する v^j は、これを Raw ソケットで受信する。v^j は、ICMP メッセージのデータ部に格納された IP データグラムを取り出す。□

[v^j–C_d 間 (L^j 内) の配送]

1. 受信した ICMP エコー応答メッセージのデータ部から IP データグラムを取り出した v^j は、この IP データグラムを Raw ソケットを用いて送信する。□

4.2 暗号通信/アドレス変換との共存

本論文で提案する IP 通信拡散手法は、暗号通信と対立、競合するものではなく、暗号通信の欠点のひとつを補完するものである。したがって、VPN 装置間の通信においては、拡散通信と暗号通信を組み合わせて使用することが望ましい。VPN 装置間の暗号通信を実現するものとして、IPsec [5,14] がある。IPsec には、公開鍵暗号を用いたデータ配送のための秘密鍵の配送機能 (IKE) と、この鍵を用いた秘密鍵通信によるデータ配送機能とが含まれている。データ配送には、3-DES が用いられている。IPsec の Linux への実装として FreeS/WAN プロジェクト [19] によるものがある。ここでは、IPsec の機能は、Linux カーネルモジュールとして実現されている。一方、拡散通信機能はアプリケーションプログラムとして実装されている。ここで、送信側の VPN 装置における処理を考えると、LAN に接続するインタフェースから受信した IP データグラムの IP ヘッダを拡散通信プログラムが読み取る必要があることから、拡散通信プログラムによる処理の前に IPsec による処理を行なうことはできない。また、拡散通信プログラムが作成した ICMP エコー要求の送信先は VPN 装置ではなく中継ルータであることから、拡散通信プログラムによる処理の後に IPsec による処理を行うこともできない。受信側の VPN 装置においても同様のことが成り立つことから、暗号通信はカーネルの機能を流用するのではなく、拡散通信プログラムの一部として実現する必要があるといえる。

VPN を利用する LAN では、プライベートアドレスが広く利用されている [10]。LAN に接続されたクライアントコンピュータがインターネット上のサーバにアクセスし、WWW や FTP などのサービスを利用する場合には、プライベートアドレスをグローバルアドレスに変換しなければならない。このアドレス変換は、LAN とインターネットとを接続するルータで行われるのが一般的であり、VPN を使用する環境では VPN 装置において実現される。実現には、IP アドレス変換のみを行う NAT(Network Address Traslator) [2] と IP アドレスとポート番号の変換を行う IP マスカレード [13] があり、Linux では iptable というカーネルモジュールとして実装されている。本実装では、iptable を使用する。

以上をまとめると、VPN に属する LAN 間の暗号通信/拡散通信はアプリケーションによって実現し、LAN とインターネットとの間の通信は、カーネルの機能によって実現する。したがって、LAN に接続するインタフェースから受信した IP データグラムについては、その送信先がインターネット上のサーバコンピュータであるか VPN に含まれる他の LAN 上のコンピュータであるかによって、その処理をカーネルで行うかアプリケーションで行うかが異なる。同様に、インターネットに接続するインタフェースから受信した IP データグラムについても、その送信元がインターネット上のサーバコンピュータであるか、VPN に含まれる他の LAN 上のコンピュータであるかによって、その処理をカーネルで行うかアプリケーションで行うかが異なる。4.1 節で示したように、拡散通信プログラムはこれらの IP データグラムを Packet ソケットもしくは Raw ソケットで受信する。この場合、受信した IP データグラムのコピーはアプリケーションによって受理、処理されるとともに、カーネルの TCP/IP モジュールでも受理、処理される。そこで、拡散通信プログラムと iptable の処理のはじめに送信先あるいは送信元のアドレスを確認し、処理が不要であるモジュールではこの IP データグラムの処理を中止することとする。

[LAN から受信したパケットの処理]
LAN から受信した IP データグラムは、Raw ソケットを通した受信によってコピーされる。この IP データグラムが VPN に含まれる LAN を送信先とする場合、アプリケーションプログラムによって ICMP エコーカプセル化とデータ部の暗号化が行われ、送信される。OS カーネルでは、iptable によって IP データグラムが破棄される。受信した IP データグラムがインターネット上のサーバを送信先とする場合、アプリケーションプログラムでは IP データグラムは破棄される。カーネルでは、iptable によって IP マスカレードの処理が施された IP データグラムが送信される。

[インターネットから受信したパケットの処理]
インターネットから受信した IP データグラムは、Raw ソケットを通した受信によってコピーされる。この IP データグラムが VPN に含まれる LAN を送信元とする場合、アプリケーションプログラムによって暗号化されたデータ部の復号化が行われ、送信される。OS カーネルでは、iptable によって IP データグラムが破棄される。受信した IP データグラムがインターネット上のサーバを送信元とする場合、アプリケーションプログラムでは IP データグラムは破棄される。カーネルでは、iptable

のアドレス変換テーブルに従ってIPマスカレードの処理が施されたIPデータグラムが送信される。

5 評価

IP通信拡散手法とそれを利用したVPN装置では、経路の拡散に用いる中継点のルータにおいて、ICMPエコーの処理が必要となる。具体的には、受信したIPデータグラムのデータ部に格納されたICMPエコー要求メッセージに対応するICMPエコー応答メッセージが作成され、これをデータ部に格納したIPデータグラムが送出される。このとき、ICMPエコー要求メッセージのデータ部にカプセル化されたIPデータグラムがICMPエコー応答メッセージのデータ部にコピーされることで、中継点を経由したIPデータグラムの配送およびこれを利用したLAN間のVPN通信を実現している。したがって、通常のIPデータグラムを処理する場合と比較して、ICMPエコーの処理を必要とする分だけ、中継ルータの負荷が増えることになる。そこで、図4のネットワーク構成において、通常のIPデータグラムを配送する場合とIP通信拡散法によって配送する場合とのスループットの違いを測定した。測定に用いたルータ装置は、以下の通りである。

- Cisco 2621: MPC860 50MHz, 32MB Memory, IOS 12.2(2)T
- Cisco 2651XM: MPC860P 80MHz, 96MB Memory, IOS 12.2(12a)
- Cisco 7206VR: MPE300 262MHz, 128MB Memory, IOS 12.0(7)T
- PCルータ: Celeron 700MHz, 256MB Memory, LINUX(Kernel 2.4.17-14k)

netperf [3] を用いて測定した結果を表1に示す。

表1: netperfを用いた測定結果 (Mbps)

ルータ	提案手法	IP通信
PC(Celeron 700MHz)	94.10	95.83
Cisco 7296VR (MPE300 262MHz)	38.52	95.98
Cisco 2651MX (MPC860 80MHz)	12.66	57.95
Cisco 2621 (NPC860 50MHz)	9.10	67.24

ICMPエコーカプセル化したパケットのスループットは、IPデータグラムのスループットに比べて低くなっていることが測定結果から分かる。ただし、この差異は、ルータのCPU性能、メモリ量の増加等により、必ずしも問題となる差異であるとはいえない。一般に、中継ルータとして用いられるのは、ISP等の所有する高性能なルータである場合が多く、LANを対象としたCPU性能が低く、メモリも少量であるものとは異なる。以上の考察により、VPNとしての適用環境においては、実用上、性能の問題はないと結論付けられる。

6 まとめと今後の課題

本論文では、暗号通信を補完し、盗聴者の使用するコンピュータの計算能力の向上に依存せずに、安全な通信路を提供するIP通信拡散手法を提案した。ここでは、通信要求の発生に対して、動的に複数経路を探索、決定する。提案手法は、インターネットのルータに特別な機能を導入する必要がない点で適用性に優れている。また、本手法のVPN装置への実装方法について述べた。性能評価実験により、中継ルータには、通常のIPデータグラムに比べて大きなオーバヘッドを要すること、ただし、適切なCPU性能、メモリ量を持つものでは、実用上問題がないことを明らかにした。VPN装置そのものにおける処理オーバヘッドを測定すること、拡散度と安全性、伝達遅延との関係について実験的に明らかにすることが今後の課題である。

参考文献

[1] Diffie, W. and Hellman, M.E., "New Directions in Cryptography," Proceedings of AFIPS National Computer Conference, pp. 109–112 (1976).

[2] Egevang, K. and Francis, P., "The IP Network Address Translator (NAT)," RFC1631 (1994).

[3] Jones,R., "Netperf Homepage," http://www.netperf.org/netperf/NetperfPage.html

[4] Karn, P., "The ESP Triple DES Transform," RFC1851 (1995).

[5] Kent, S. and Atkinson, R., "Security Architecture for the Internet Protocol," RFC2401 (1998).

[6] Lai,X., "On the Design and Security of Block Ciphers," ETH Series in Information Processing (1992).

[7] Perkins, C., "IP Encapsulation within IP," RFC2003 (1996).

[8] Postel, J., "Internet Protocol," RFC791 (1981).

[9] Postel, J., "Internet Control Message Protocol," RFC792 (1981).

[10] Rekhter, Y., Moskowitz, B., Karrenberg, D., Groot, G.J. and Lear, E., "Address Allocation for Private Internets," RFC1918 (1996).

[11] Rivest, R.L., Shamir, A. and Adleman, L.M., "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," Communications of the ACM, vol. 21, no. 2, pp. 120–126 (1978).

[12] Rivest, R.L., Shamir, A. and Adleman, L.M., "On Digital Signatures and Public Key Cryptosystems," MIT Laboratory for Computer Science, Technical Report, MIT/LCS/TR212 (1979).

[13] Srisuresh, P. and Holdrege, M., "IP Network Address Translator (NAT) Terminology and Considerations," RFC2663 (1999).

[14] Thayer, R., Doraswamy, N. and Glenn, R., "IP Security Document Roadmap," RFC2411 (1998).

[15] Wiener,M,J., "Efficient DES Key Search," TR-244, School of Computer Science, Carleton University (1994).

[16] ANSI X3.92,"American National Standard for Data Encryption Algorithm(DEA)," American National Standards Institute (1981).

[17] ANSI X9.17(Revised),"American National Standard for Financial Institution Key Management(Wholesale)," American Bankers Association (1985).

[18] "Merkle-Hellman," http://www.wikipedia.org/wiki/Merkle-Hellman.

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