システムソフトウェアと
オペレーティング・システム 64－3
マルチメディア通信と分散処理 65－3
（１９９４．５．１９）

トンネリング技術の考察

出水法俊[1]　山口英　山本平一

奈良先端科学技術大学院大学

トンネリングの利用はインターネット技術の随所で見られる。しかし一方ではトンネリングには様々な問題点があることも指摘されている。そこで本稿はトンネリングを安全に利用するための方法を提案する。まず、トンネリングを議論する上でベースと成り得る階層モデルとして「自由順序階層モデル」を提案し、そのモデルにおいてトンネリングが自然に記述できることを示す。次に、トンネリングの経路上での問題点及び経路制御の問題点を自由順序階層モデルにおける問題としてとらえ、その解決方法を提案する。最後に、自由順序階層モデルとは独立した問題についての解決方法を提案する。

For a Safety Tunneling Technique

Noritoshi Demizu[2]　Suguru Yamaguchi　Heiichi Yamamoto

Nara Institute of Science and Technology, Japan

In recent years, the tunneling technique has become popular in some internet protocol technologies. However, tunneling has several unresolved problems. In this paper, we give solutions for the problems of tunneling to make tunneling safe. First of all, we propose "free-ordered layering model" which can naturally describe tunneling. We then discuss "arrive or die" problem and routing problem in the model and give solutions for them. Finally, we give solutions for the problems independent on our "free-ordered layering model".

[1] オムロン株式会社より留学中
[2] He also works for OMRON Corporation.

1 はじめに

トンネリングはインターネットにおいて、主に経路制御を簡単にするための手段として多く利用されている。

ひとつ目の例として、新しいプロトコルや異なるプロトコルを通す場面が挙げられる。例えば現在のマルチキャストでは、マルチキャストを扱えないルータをスキップするためにトンネリングを利用している。あるいは次世代 IP (IPng) への移行期にトンネリングを利用することが考えられている [3][13]。また、他のプロトコルを運ぶために "OSI CLNP over IP"[4][9]、"IPX over IP"[14] などが定義されている。GRE (Generic Routing Encapsulation)[5][6] はこのような場面での利用を想定している。

2 番目の例として、移動ホストの支援が挙げられる。例えばコロンビア大学方式 [8] では、移動ホストはキャンパス全域にわたる仮想ネットワークに無線で接続することでインターネットと通信が可能になる。その仮想ネットワークはキャンパスの各所にある MSR (Mobile Support Router) の間をトンネリングで結ぶことで実現している。

3 番目の例として、経路制御を簡単にするために、ネットワーク・トポロジーの補助線としての仮想ネットワークの提供が挙げられる。例えば IDRP は、AS を通過するデータグラムを border gateway 間で運ぶ方法のひとつにトンネリングを考えている [7]。これにより AS 内の経路制御と AS 間の経路制御を分けて考えることができるようになる。別の例として、ルータの forwarding 技術の制限 [2] や経路制御プロトコルの制限 [15] を避ける手段としての利用も考えられている。さらに、企業が事業所間の通信をインターネット経由で行う場面でセキュリティ確保の手段としても考えられている [16]。Internet Encapsulation Protocol[18] は、以上のような場面での利用を想定している。"PPP over TCP" なども利用可能である。以上を実装する際のトンネリング利用モデルとして DDT が提案されている [1]。

4 番目の例として、IP option の拡張があげられる。現在の IP では IP option 長は最大 40 オクテットしかなく、暗号情報等を載せるには小さすぎる。そこでトンネリングの形を借りて、IP option に入れるべき情報と送るべきデータグラムを上位プロトコルのデータとして扱うことにより目的を達成することが考えられている。

以上のようにトンネリングは広く用いられているにもかかわらず、未解決な問題点も多く残っている。例えば、パケットが爆発的に増える可能性が指摘されている [17]。また、パケットを識別できなくなるという問題もある。さらに、現在の経路制御アルゴリズムはトンネリングを含んだネットワークを扱うことができない。そのため、トンネリングは危険で避けるべきと考えているインターネット研究者も多い。

本稿は、以下の構成でトンネリングを安全に利用するための方法を提案する。まず 2 節でトンネリングを定義し、その問題点を明らかにする。次に 3 節でトンネリングを記述することが可能な階層モデルである自由順序階層モデルを提案する。4 節では、自由順序階層モデルをトンネリングにおける諸問題に適用し、その解決方法を提案する。さらに提案された方法が既存のネットワーク技術の拡張により実現可能であることを示す。5 節ではその他の問題点の解決方法を示す。

2 トンネリングの定義と問題点

本稿におけるトンネリングとは、広義にはネットワーク層のプロトコル x をネットワーク層以上のプロトコル y で運ぶことである。この状態を x *over* y と呼ぶ。狭義には x と y とが等しい場合を指す。本稿では、トンネルに入れるための操作を encapsulation、それを行なうルータをトンネルの入口、そのとき付加されるヘッダを encapsulation header と言う。トンネルから出すための操作を decapsulation、それを行なうルータをトンネルの出口と言う。

トンネリングには以下に示す問題点があることが知られている。以下では *TP-num* の形で参照する。

1. モデルとの相性
   広義のトンネリングは複数のプロトコル・ファミリにまたがった現象であり、狭義のトンネリングは Layer Violation を含んだ現象であるが、どちらも現在のプロトコル階層モデルでは記述できない現象である。

2. 迷子パケット
   目的地に到着できないパケットは必ず消滅しなければならない。ところがトンネリングにおける TTL の機構が確立されていないため、迷子

のパケットが必要以上に長い寿命を持ったり消滅しなかったりする可能性がある。そのようなパケットがトンネルを抜け出ることなく再び同一トンネルに突入するような無限ループに出会うと、encapsulation header によりパケットが無限に大きくなりうる [17]。

3. QoS
   QoS (Quality of Service) はパケットが経路に対して持つ要求を表すものである。encapsulation において QoS が無視あるいは十分に反映されない場合が多いため、パケットの経路制御が不適当になる可能性がある。

4. パケットの識別
   encapsulation は、元々のパケットの識別を難しくする。その結果 packet filter がうまく機能しなくなり、セキュリティ上問題となる。またネットワークの利用統計を正しく取ることも難しくなるため、ネットワークを評価する上で支障となる。

5. 経路制御
   トンネリングによる仮想ネットワークを含むネットワークにおける一般的な経路制御方法は知られていない。

6. エラー報告
   トンネル内のエラーの報告がトンネルの入口のルータに送られる可能性がある。その場合、その報告を無視すべきか真の送信者に転送すべきかの判断基準 [18]、及びエラー報告パケットの組み立て方が問題になる。

7. MTU
   トンネリングによる仮想ネットワークを構築したとき、実際には存在しない MTU が実装上必要になる場合がある。

8. オーバーヘッド
   トンネリングを利用すると、encapsulation ヘッダのためのバンド幅が余分に必要になる。また encapsulation/decapsulation 操作のために CPU が消費される。

9. 制御不能なトポロジー
   トンネリングを用いるとユーザは自由に仮想リンクを作ることができる。その結果、ネットワーク管理者はトポロジーを把握できなくなり管理に支障がでる可能性がある。

トンネリングの問題点はトンネリングを包含したモデルの上で議論すべきことから、以上の問題点のうち最も基本となる問題はモデルとの相性 (TP-1) であると言える。そこで 3 節ではトンネリングを包含するモデルを提案する。

3 自由順序階層モデル

本節ではトンネリングを包含する階層モデルである **自由順序階層モデル**を提案する。

狭義のトンネリングが Layer Violation を引き起こす原因は、従来の階層モデルでは階層間の序列、各層の役割分担、全体の深さが厳格に決められていたことにある。これらは同時に複数のプロトコル・ファミリにまたがる広義のトンネリングを扱えない原因でもある。トンネリングを議論する上でベースとなりうる階層モデルでは、それらの制約はなくさねばならない。そこで本稿では次のような自由順序階層モデルを提案する。

- プロトコルの関係は階層的である。
- 各プロトコルが下の層に対して要求する制約条件を満たしさえすれば目的に応じて自由に階層を組み立てることができる。

自由順序階層モデルはトンネリングを議論できるだけでなく、暗号・認証・圧縮等の機能を持つ層を任意の場所に自然に挿入を許す枠組みでもある。様々な階層を任意の順序で組み立てるためには、各層の SAP (Service Access Point) のインタフェースと制約条件記法が共通化されている必要がある。その意味では各ノードにおけるプロトコル選択機構は System V の STREAMS に似ている。

通信経路を通しての階層の関係は、横軸に始点から終点までの経路、縦軸に各点における階層構成を取ることにより表すことができる。このとき、縦軸方向の再上層には伝えるべき情報が位置し、再下層には物理媒体が位置する。その間は、ネットワーク機能やトランスポート機能等を持つ層が任意の順番で任意の数並んでいる。階層の深さは経路上で一定

ではなく、ところによって異なっている可能性がある。横軸の通信経路は一般にトポロジーを持つので2次元平面と考えることができることから、縦軸の階層を含めた全体では3次元空間と考えることができる。

ここで、自由順序階層モデルにおいてトンネリングを再定義する。広義には、ネットワーク機能を持つ層が複数存在する階層構造を作ることである。狭義には、アドレス空間を共有する場合を指す。このように自由順序階層モデルはトンネリングを自然に記述できる (TP-1 参照)。従来のネットワーク技術が2次元平面のトポロジーを対象としたものであることを考えると、それをそのまま3次元空間の現象であるトンネリングに適用するのは危険であることが容易に想像がつく。

自由順序階層モデルでは、従来の階層モデルは階層の構造を理解するための "view" とみなされる。従来の階層モデルによる view がひとつだけ定義できるとき、そのモデルが保証する安全性を得ることができる。逆に複数の view が存在するとき、個々の view の前提が矛盾して問題が発生する可能性がある。トンネリングが行なわれるときは、まさに view が複数存在している。

トンネリングの問題点を解決するためには、自由順序階層モデルにおける一般的な安全性保証手法を構築する必要がある。その手法をトンネリングを利用する場面の状況に合わせて適用すれば、トンネリングの問題点の多くは解決可能である。

4 モデルに基づく解決方法案

自由階層モデルはトンネリングを包含していることから、トンネリングの経路上の問題点 (TP-2, TP-3, TP-4) 及び経路制御の問題点 (TP-5) は自由順序階層モデルにおいても同様に問題となる。逆に、自由順序階層モデルにおけるこれらの問題に対する解決方法は、トンネリングにおける上記問題点に対する解決方法にもなる。本節では2次元平面のネットワーク技術を3次元空間のネットワーク技術に拡張する形で、自由順序階層モデルにおけるこれらの問題の解決方法を提案する。なお、本節では特に断らない限り、自由順序階層モデルの中でもネットワーク機能を持つ層だけに注目する。

4.1 経路上の問題点

経路上における問題点 (TP-2, TP-3, TP-4) を解決するには以下を実現せねばならない。

- 目的地に到着できないパケットは消滅させる。
- 目的地までできるだけ望み通りの経路を通す。
- 流れているパケットの正体がわかるようにする。

これらの目標の達成方法は、階層を縦方向に移動する時の encapsulation の方法、つまりトンネリングの種類によって異なる。ここでは "Connectionless" トンネルと "Connection-Oriented" トンネルについて考える。

Connectionless トンネリングとは、パケットをひとつひとつ encapsulation して別々に送るものである。この場合、送られるデータに注目すると、移動範囲が2次元平面のネットワークに閉じず、3次元空間のネットワークに広がったと考えることができる。このとき、TTL, QoS, ID などの属性が送られるデータ自身に付随するものであると考えて、それらを一番外側のヘッダに書くことにすれば、3次元空間のネットワークにおいても2次元平面のネットワークと同様の手法で上記の目標が達成できる。つまり、横軸方向の移動では TTL については移動のたびに減少させ、QoS, ID については変更しないようにすれば良い。縦軸方向に降りるときは、無限に降り続けるのを防ぐために TTL を減少させた上で、TTL, QoS, ID を次の層に合わせて正しくマップしなければならない。これらを正しくマップする方法、及び ID 情報の偽造防止方法は今後の研究課題である。

Connection-Oriented トンネルとは、パケットをストリーム内に直列に並べて、フロー制御をしながら送受信するものである。フローは同じ層内でしか作れないことから、2次元平面のネットワークにおけるフロー制御と同じ手法で Connection-Oriented トンネルのフロー制御を行なうことができる。目的地に到着できないパケットを消滅させることについては、フロー制御がその機能を持っている。フロー自身を構成するパケットが目的地に到着できない場合に消滅させるには Connection-less トンネリングの手法を使う。QoS, ID については、個々のフローを通るデータは一般に同質であると仮定されているため、フローを複数用意して QoS, ID などに合わせて

フローを選択しなければならない。これは 2 次元においても ftp と ftp-data のように目的別にフローを生成しているのと同じである。実際には ID ごとにフローを用意するのは難しいため、パケットの完全な識別は難しい。異なるアプローチとして、フローの QoS, ID を動的に変更する方法も考えられる。

4.2 経路制御

経路制御における問題点 (TP-5) は、3 次元空間のネットワークにおける経路の依存関係をループさせない仕組みを用意することで解決できると考える。ここでは依存関係を次のように定義する。ある層 x に存在する経路のトラヒックを、下の層 y が運んでいるとき、層 x は層 y に依存しているとする。また、同じアドレス空間を持つ層は互いに依存し合っているとみなす。この定義にしたがって依存する層から依存される層に矢を書けば、各層間の依存関係を表す有向グラフが作成できる。自由順序階層モデルにおける経路制御は、その結果でき上がった有向グラフの性質で分類して考える。

グラフが木である場合、つまりループがない場合は、各層ごとに閉じて経路制御を考えることができる。このとき、2 次元平面のネットワークにおける経路制御技術がそのまま利用できる。従来の階層モデルでは一般にこの性質が成り立つ。

グラフにループは存在するが、個々のループはすべて、同じアドレス空間を持つ層の間のものである場合は、ループに関係しない層では 2 次元平面のネットワークにおける経路制御技術がそのまま適用できる。ループに関係している層では、4.3 節で述べる経路制御アルゴリズムを利用できる。

それ以外の場合、依存する経路のプロトコル種、アドレス空間、アドレスの組を経路制御プロトコルで交換した上で、それに 4.3 節で述べる経路制御アルゴリズムを適用できる。各層のすべてのプロトコルでこの機構を支援する経路制御プロトコルを利用するのは難しいため、このような状態のネットワークを構築することは好ましくないと考える。

以上の方法の検証は今後の研究課題である。

4.3 経路制御アルゴリズム

3 次元空間における経路制御では、トンネルを抜け出ることなく再び同じトンネルに突入したり、利用するトンネルの入口への到達性が失われたりする問題を解決しなければならない。

Distance Vecotr 型の経路制御プロトコルでは、各経路情報に「destination に到達するまでに通過するすべてのトンネルについて、各トンネルの入口と出口が持つすべてのアドレスのリスト」を属性として追加できなければならない。これを用いて各ルータでは次の処理を行なう。

1. 上記リストに自分のアドレスが含まれている経路を捨てる (3 次元空間ループ対策)。
2. 利用できないトンネルを利用する経路を捨てる。
3. トンネル入口に確実に到達するための経路情報を追加 (トンネルの入口への到達性)。
4. 通常の経路制御アルゴリズムを適用する。

この方法の詳細化、検証及び経路の metric の評価方法と、Link State 型経路制御プロトコルの場合のアルゴリズムは今後の研究課題である。

5 その他の解決方法案

本節では TP-6, TP-7, TP-8, TP-9 の解決方法を考える。

エラー報告方法 (TP-6) の問題はプロトコル依存度が大きいため一般的な解決方法を述べるのは難しい。IP の場合の提案は [1] にある。

実際には厳格な MTU は存在しない層であっても、実装時上 MTU が必要になる場合ある (TP-7)。その下の層が物理ネットワークである場合は、トンネルの経路における最小の MTU をその層の MTU とするのが良いと考える [12][11][10]。そうでないときは、その下の層の MTU からその層における encapsulation header の大きさを引いたものを MTU にするのが良いと考える。その結果、各ノード内において層間で依存関係のループができたとき MTU が 0 に収束するため、パケットがループに飛び込むのを防ぐことができる。

オーバーヘッド (TP-8) については、encapsulation protocol を工夫して bandwidth や CPU の消費を少なくするしかないであろう。

制御不能なトポロジー (TP-9) にどのように対処するかについては未解決である。

6 結論

本稿では、まずトンネリングを議論するためのベースとなる階層モデルとして自由順序階層モデルを提案し、このモデルでトンネルが自然に記述できることを示した。次に、トンネリングにおける問題点のうち、経路上の問題点及び経路制御における問題点を、自由順序階層モデルの中で解決する方法を提案した。さらに、個々の問題点とはその他の問題点について個々に解決方法を提案した。

未解決の問題点として、ネットワーク管理者がネットワークのトポロジーを把握できなくなる危険性が残っている。

今後の課題としては、本稿で提案した方法の IP への適用、その実装、そしてその実装を利用しての検証が残されている。

謝辞

本研究を進めるにあたり、意義ある議論と貴重なアドバイスをくださった WIDE Project のメンバに感謝します。

参考文献

[1] Noritoshi Demizu. DDT — A Viersatile Tunneling Technology. In *Proceedings of INET'94/JENC5*, 1994.

[2] K. Yamamoto et al. Autonomous System Partitioning and Policy Routing in the Japanese Internet. In *Proceedings of INET'93*, August 1993.

[3] Robert E. Gilligan, Erik Nordmark, and Bob Hinden. Internet Draft — IPAE: The SIPP Interoperability and Transition Mechanism. November 1993.

[4] R. Hagens, N. Hall, and M. Rose. Use of the Internet as a subnetwork for experimentation with the OSI network layer, RFC1070. February 1989.

[5] Stan Hanks, Tony Li, and Paul Traina. Internet Draft — Generic Routing Encapsulation (GRE). September 1993.

[6] Stan Hanks, Tony Li, and Paul Traina. Internet Draft — Generic Routing Encapsulation over IPv4 networks. September 1993.

[7] Susan Hares and John Scudder. Internet Draft — IDRP for IP. September 1993.

[8] John Ioannidis, Dan Duchamp, and Gerald Q. Maguire Jr. IP-based Protocols for Mobile Internetworking. In *Proceedings of ACM SIGCOMM'91*, pages 235–245, September 1991.

[9] Dave Katz. Internet Draft — Tunneling the OSI Network Layer over IP (EON). March 1994.

[10] C. Kent, K. McCloghrie, J. Mogul, and C. Partridge. IP MTU Discovery options, RFC1063. January 1988.

[11] S. Knowles. IESG Advice from Experience with Path MTU Discovery, RFC1435. March 1993.

[12] J Mogul and Steve Deering. Path MTU Discovery, RFC1191. November 1990.

[13] David M. Piscitello. Internet Draft — Transition Plan for TUBA/CLNP. March 1994.

[14] D Provan. Tunneling IPX traffic through IP networks, RFC1234. June 1991.

[15] Y. Rekhter. Internet Draft — Selecting an Indirect Provider. December 1993.

[16] Y. Rekhter, B. Moskowitz, D. Karrenberg, and G. de Groot. Address Allocation for Private Internets, RFC1597. March 1994.

[17] P. Tsuchiya. Mutual Encapsulation Considered Dangerous, RFC1326. May 1992.

[18] W. Woodburn and D. Mills. A Scheme for an Internet Encapsulation Protocol: Version 1, RFC 1241. July 1991.