

MS-DOS への移動ホスト用プロトコルVIPの実装

渡辺 恭人¹ 慶應義塾大学環境情報学部
 植原 啓介² 電気通信大学情報工学専攻
 富永 明宏³ 慶應義塾大学環境情報学部
 寺岡 文男⁴ (株) ソニーコンピュータサイエンス研究所
 村井 純⁵ 慶應義塾大学環境情報学部

概要

インターネット上でホスト移動透過性を実現するプロトコルとして我々はVIP (Virtual Internet Protocol)[1][2]を提案している。現在VIPはUNIX上で稼働している。しかし携帯型計算機用のオペレーティングシステムとしては、より小さなハードウェアでも動作するMS-DOSが有利である。本稿ではMS-DOS上のネットワークソフトウェアであるKA9QにVIPを実装し、MS-DOSによる移動計算機環境を実現した。MS-DOSはネットワークプロトコルスタックのプリミティブな部分を本来持っておらず、またマルチタスクシステムではない。VIPはVIPアドレスとIPアドレスの二重のアドレスを持ち、入力出力の際にアドレスの検査と入れ換えを行なう。これらはKA9Qを利用することで実装できた。VIPを組み込むことによりKA9Qの実行ファイル(net.exe)の大きさは20% (約220KByteから約260KByte)増加した。本ソフトウェアは稼働しているが、現在、移動時のネットワークパラメタの自動設定などが未実装であり、将来拡張予定である。本ソフトウェアは配布する予定である。

Implementation of VIP, a Protocol Supporting Host Mobility, on MS-DOS

Yasuhito Watanabe Keio University
 Keisuke Uehara University of Electro-Communications
 Akihiro Tominaga Keio University
 Fumio Teraoka Sony Computer Science Laboratory
 Jun Murai Keio University

Abstract

We are proposing VIP (Virtual Internet Protocol) as a protocol providing host migration transparency in the Internet. Currently, VIP is running on UNIX. However, MS-DOS is preferable as an operating system for mobile computers because it can run on poorer hardware. In this paper, we built a mobile computing environment on MS-DOS by implementing VIP in KA9Q, a network software for MS-DOS. MS-DOS has no primitive function of protocol stack of network and no multitask system. VIP has two addresses, which are VIP address and IP address, and check and exchange their addresses when packets are input and output. These functions can be implemented as a result of using KA9Q. The size of executable file of KA9Q (net.exe) increased only 20% (from 220KBytes to 260KBytes). Since automatic configuration of network parameters upon moving has not been implemented, it shall be included in our software. We plan to distribute our software.

¹riho-m@sfc.wide.ad.jp

²kei@cs.uec.ac.jp

³tomy@sfc.wide.ad.jp

⁴tera@csl.sony.co.jp

⁵jun@sfc.wide.ad.jp

1 はじめに

近年、計算機は高性能化 小型軽量化が進行し、ノートブック型コンピュータと呼ばれる計算機が登場している。これにより、人々は高性能でかつ小型な計算機を持ち運んで利用するようになった。また一方で、広域コンピュータネットワークが普及している。将来の計算機環境では、これら二つの利用形態が融合した形態、つまり、ネットワークに接続しながら持ち運ぶという移動利用形態が必要となる。しかし、一般的に利用されているインターネットではこのような利用形態は考慮されていない。インターネット上で移動計算機をサポートするため、我々は Virtual Internet Protocol(VIP) [1][2] を提案している。これは、仮想ネットワーク [3] の概念を IP(Internet Protocol) に適用することによりホストの移動透過性を実現したプロトコルである。

また、当初は UNIX のワークステーションなど一部の計算機でのみ利用可能だったネットワーク機能も、現在ではパーソナルコンピュータ(PC)上で利用可能である。MS-DOS 及びその互換 DOS は全世界の数十万台の PC 上で使用されており、最も普及しているオペレーティングシステムである。その提供する機能は、ファイルシステム、最小限度のメモリ管理のみであり、システムは基本的に、シングルタスク、シングルユーザである。また MS-DOS は UNIX 等のマルチタスク処理を行なうオペレーティングシステムと比較して、負荷の軽い実行及び処理環境が得られる。したがって、ハードウェアの制約が大きい携帯型の PC のオペレーティングシステムとしては MS-DOS が適しているといえる。ある。本研究では、VIP を実装することにより MS-DOS 上に移動計算機環境を実現することを目標とする。

以下、第 2 章で VIP の概要を述べる。第 3 章では MS-DOS 上のネットワークソフトウェアである KA9Q の概要について述べる。第 4 章で KA9Q への VIP の実装について述べ、第 5 章で実装したものを評価する。第 6 章で考察を行ない、第 7 章で本論文を締めくくる。

2 VIP の概要

ネットワーク層プロトコルである IP では、ホストをインターネットアドレスまたは IP アドレスと呼ばれる 32 ビットのアドレスで特定する。IP アドレスは、ネットワークを識別する netid とネットワーク上のホストを識別する hostid の対で構成される。netid はネットワーク全体を通じて一意であり、hostid は netid が示すサブネットワークの中で一意である。インターネット全体では IP アドレスはホストの識別子ではなく位置情報を示すものであると考えられる。しかし、トランスポート層以上では、IP アドレスが相手を特定するための識別子として使用されている。

このようなアドレス形式を規定するインターネットアーキテクチャでは、相手ホストが他のネットワークに移動すると相手ホストの IP アドレスが変化する。したがって、相手ホストを識別することができないために通信できない。そのため、移動透過性は実現されていない。ホストの位置を意識することなくホストの識別を行なうためには、ホストアドレスが位置情報を含まない識別子のみで構成される必要がある。

そのため、VIP では位置情報を含むアドレス(IP アドレス)に加え、ホストを特定する識別子(VIP アドレス)を導入した。VIP アドレスはホストを特定する完全な識別子であり、移動した場合も変化しない。IP アドレスはネットワーク上のホストの位置を表し、IP パケットの経路制御に用いられる。この場合、ホストがどこに移動しても、TCP/UDP 層(トランスポート層)からは VIP アドレスで相手を特定する。これにより、移動による IP アドレスの変化を TCP/UDP 層が意識することがなくなり、移動透過性が保証される。したがって、VIP 層は TCP/UDP 層から相手ホストの識別子である VIP アドレスを受けとり、それを実際のホストの位置を表す IP アドレスに変換して IP 層に渡すということが必要になる。アドレス変換を行なうために、アドレス変換テーブル(Address Mapping Table(AMT))という、ホストの識別子(VIP アドレス)を鍵として位置情報(IP アドレス)を得るための表を導入する。この AMT は各ホストやルータで保持される。

3 KA9Q の概要

3.1 特徴

MS-DOS はシングルタスクシステムである。また、MS-DOS にはネットワーク関係の処理を行なう機構は含まれておらず、物理層のデバイスドライバ、データリンク層のネットワークインタフェースソフトウェア(例えば FTP software 社のパケットドライバ)が必要である。アプリケーションはその上で稼働する。VIP も含めてネットワーク関係のプログラムは、UNIX で設計実装されたものが多く、マルチタスクシステムに依存した部分が存在する。例えば、UNIX で利用されているデーモン等をそのまま実装することはできない。

MS-DOS 上でネットワークのシステムを構築する場合、このような制約を考慮すると以下のような解決法がある。

1. サーバ専用、クライアント専用のホストに分けてネットワークシステムを構築する。
2. (疑似) マルチタスク システムになるようなプログラムを走らせ、その上ですべてを実行する。

この二つの解決法のうち、後者を選択する場合を考える。この場合は、KA9Q[4] というパッケージを使用

することで解決される。KA9QはMS-DOSを使用するIBM-PCとその互換機上などで動作するTCP/IPのパッケージであり、バイナリとソースを含んでいる。これはPhil Karnが中心となって骨格部分が作成され、アマチュア無線でのパケット通信ネットワークで盛んに使用されている。さまざまな人々によって核となるプロトコルは精力的に試験され、実際のネットワークを構築するための頑丈なプラットフォームとなっている。また、プログラミングのインタフェースがより簡潔な形で実装されており、新たなプロトコルやアプリケーションの実装や実験を行なうことができる。

KA9Qは、それ自身で簡易的なオペレーティングシステムとして機能する。そして、最大の特徴はカーネル内部でシングルプロセスマルチスレッド型のプロセス制御を行なっていることである。これにより、メモリが許す限り複数のプログラムを並行して実行することができるため、MS-DOSでは本来不可能なデーモンプロセスを利用したプログラミングが容易になる。また、複数のネットワークセッションを並行して実行可能なので、ユーザは効率的な利用環境を得ることができる。

3.2 構造

KA9Qパッケージでは図1のような構造でネットワークが実装されている。

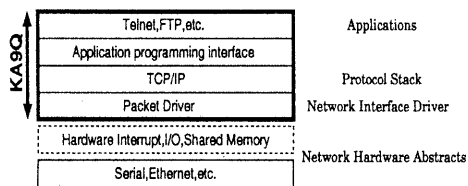


図1: KA9Qのネットワーク構造

MS-DOSのカーネルにはネットワーク関係のコードが含まれていない。そのため、各々のハードウェアに対応したデバイスドライバとその上層にネットワークのプログラムを実装するためのプログラムインタフェースを提供するドライバが必要となる。KA9Qパッケージには、FTP Software社の“PC/TCP Version1.08パケットドライバの仕様”と互換性のあるパケットドライバが提供されている。これは、データリンクレベルでネットワークインタフェースが使用できるプログラム用インタフェースである。

3.3 KA9Qでのネットワーク実装

KA9Qにおけるネットワーク処理の流れは、図2のようになっている。KA9Qでは、二つのデーモンプロ

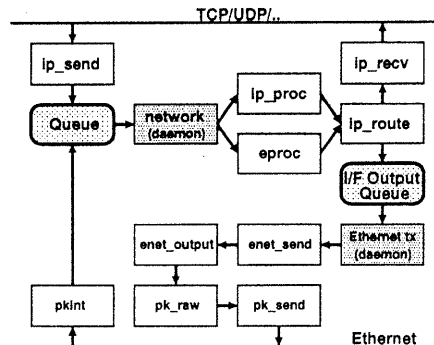


図2: KA9QのIP関係ブロックダイアグラム (Ethernetの場合)

セスと二つのキューによって処理が成立している。まず、ネットワークインタフェースがパケットを受信すると割り込みによってpkintが呼び出され、受信パケットはキューに挿入される。また、上層から送信されるべきパケットは、ip_sendにてIPヘッダを付加されてIPパケットとなり、同じくキューに挿入される。ここでnetworkという名のデーモンプロセスはキューを監視しており、パケットを取り出す。パケットがキューに入力されたことがわかると、networkはそれがどのようなパケットか調べ、IPパケットであれば、ip_procを通じてip_routeに渡される。Ethernetのパケットであれば、Ethernetヘッダを取り除きIPパケット化してip_routeへ渡す。ip_routeでは、渡されたIPパケットの終点をチェックし、自分宛ならばIPヘッダを取り除いて上層へ渡し、そうでなければ、出力インタフェース用のキューにパケットを渡す。そこでもう一つのデーモンプロセス、ethernet_txがキューを監視して、パケットを取り出す。この場合はEthernetに送信するために、enet_sendでEthernetアドレスの対応づけを行なう。enet_outputでEthernetヘッダを付加し、それをpk_rawでパケットドライバに渡す。最後にパケットドライバの送信ルーチンpk_sendより送信される。

4 実装

4.1 問題点と解決法

ここではVIPをKA9Qに実装する場合の問題点と解決法について述べる。これらは、主にKA9QでのIPの処理のUNIXとの相違、VIPの問題点などから次のように絞られた。

1. IPパケットからVIPパケットへの変換

VIPにおけるルータの機能は、中継すべきパケッ

4.4 移動方法

現在の実装では、有線のネットワークにおいて次のような手順でホスト移動が行なわれる。

1. ケーブルを外して、別のサブネットワークにケーブルを接続する。
2. 接続先でBOOTPクライアントを起動。
3. BOOTPにより、IPアドレス、ネットマスク、ブロードキャストアドレス、デフォルトゲートウェイアドレスを取得。
4. VIPのコントロールパケットを送信。
5. 通信再開。

このように移動しても、TCPのコネクションは保存される。また、位置に関わらず、相手のVIPアドレスを指定することにより通信することができる。

5 デモンストレーション

VIPは、インターネットでの移動サポートプロトコルの標準案候補として提案されている。昨年、第28回IETF (Internet Engineering Task Force) ミーティングという技術者会議でデモンストレーションを行ない、紹介した。また、国内でのネットワーク管理者の会議であるIPmeeting'93においても同様のデモンストレーションを行なった。デモンストレーション環境の概略図を図4に示す。実験環境は図のように3つのサ

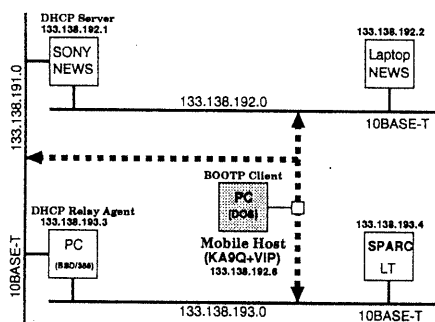


図4: IETF'93及びIPmeeting'93でのデモ環境

ブネットワークから構成されている有線(10BASE-T)のネットワークである。その3つのサブネットワーク間をホストが移動する。

133.138.192.6がPCのVIPアドレスであり、移動しても変化しない。133.138.191.0のサブネットワークでは、133.138.191.6、133.138.192.0のサブネットワークでは133.138.192.6、133.138.193.0のサブネットワークでは133.138.193.6、のIPアドレスがそれぞれDHCP

によりPCに割り当てられる。それぞれのサブネットワークでIPアドレスが割り当てられた時に、VIPアドレスとIPアドレスとの対で構成されるAMTが作成され各ホストやルータに伝播する。したがって相手ホストが移動した場合にはAMTによりVIPアドレスとIPアドレスの変換が行なわれ、相手の位置に関係なく通信を継続することが可能になる。

このような環境において、移動ホストは3つのサブネットワークを移動しても、遠隔ログインコマンドtelnetのセッションが保存され、ユーザにとっての移動透過性も実現されることを示した。また、PCがどのサブネットワークに接続されている場合でも、VIPアドレスを指定することにより通信できることを示した。

6 考察

6.1 VIPについて

実験環境においては、異なるサブネットワーク間を移動してIPアドレスが変化しても、セッションが保存され、通信が継続できることが明らかになった。これは、VIPがホストの識別子としてVIPアドレスを使用することで移動透過性を実現していることを実証している。

さらに、VIPの機構は比較的単純であるために新たに追加する部分 変更を要する既存の部分は非常に少ない。これは、VIP自体のポータビリティの高さを示すものである。

また、VIPではルートエントリに関する処理は行なわない。よって、移動時におけるホストの再設定が行なわれたときには移動前のIPアドレスでのルートエントリを消去し、その他のルートエントリを初期化する必要がある。これにより移動後のパケットが正しく経路制御される。

6.2 今回の実装について

VIPの課題として移動先でのホストの設定問題が存在していた。この度の実装では移動先でのホストの設定情報を取得するために、移動時にBOOTPのクライアントを使用する。これにより、移動時のホスト設定の手続きが半自動化され簡略化された。但し、移動時の回線の切断 接続は自動的に認識できない。したがって、BOOTPはユーザによって明示的に起動されなければならない。現在は特定のキーを押すことでBOOTPを起動するように実装しているが、回線状態の認識とホストの設定の自動化を検討中である。その際BOOTPは取得できる情報の種類が限定されるため、より多数の情報を取得できるDHCPの実装が望ましい。

また、VIPでのルータの役割はIPパケットをVIPパケットへ変換することであるが、MS-DOSの移動ホストはルータの機能を持たせるほど重要度が高いと考えられないため、実装していない。将来、無線ネットワー

ク用経路制御プロトコルは全ての移動ホストがルータになることも予想されるため、ルータの機能は必要となる。

6.3 トランスポート層の問題

VIPを使用することで他層の問題点も明らかになった。トランスポート層、特にTCPを使用するアプリケーションでは、回線を切断して移動して再接続するまでに一定時間以上経過すると、タイムアウトによりそのセッションは終了してしまう。これはTCPの仕様上の問題である。このような状況は、有線ほどの安定度を持たない無線ネットワークで生じる可能性がある。したがって、切断 接続を検知し、オペレーティングシステムレベル、アプリケーションレベルでこのような状況を支援する機能が必要である。

6.4 MS-DOS 上での移動計算機環境

現在のKA9Q版VIPの実験では、telnetを使用している。しかし、telnetでは計算機環境を遠隔ホストに依存している。つまり、プロセス自身と自分の情報、(ファイルなど)両方の実体は移動ホストには存在しないのである。自分の情報は仮想的に移動ホスト上において移動ホスト側の計算機資源を利用して作業する形態、さらに固定された遠隔ホスト上の自分の情報を移動ホストのファイルシステムと共有して作業する形態、の二つの形態が必要である。MS-DOSではこのような環境を支援する機能が必要がある。

7 まとめ

本研究では、VIPのUNIXへの依存性とMS-DOS上でのネットワーク機能実装の際の問題点を明らかにした。それらは、VIPがVIPアドレスとIPアドレスという二重のアドレスを持ち入出力時にアドレスの検査と入れ換えを行なうこと、MS-DOSがネットワークプロトコルスタックのプリミティブを持っていないこと、マルチタスクシステムではないことである。その問題点をKA9QというTCP/IPパッケージを利用することで解決することにより、VIPをMS-DOSへ実装した。これにより、MS-DOS利用による移動計算機環境の基盤を実現するとともに、VIPのオペレーティングシステムに対する汎用性を高めることができた。また、移動時のアドレス及び設定情報の取得に関しては、BOOTPのクライアントを使用することで手続きを半自動化した。

しかし、VIPのみでは、オペレーティングシステムレベル、アプリケーションレベルでの移動計算機環境に対する支援は十分ではない。それらは、各々のアプリケーションに依存した処理に関係する部分の対応や、ネットワークとの切断 接続が発生するレベルや処理の方法といった別の面からの研究が必要である。

また、MS-DOS上のVIPに関する課題は、回線の切断 接続を検知して自動的にネットワークの設定を行なう機構を実装することである。また、移動時のホストの設定に関してBOOTPより設定情報が多くアドレスに関して明確な管理手法を持つDHCPクライアントを実装する必要がある。その他、SLIP PPPなどシリアル回線用のデータリンク層プロトコルでの利用実験、KA9Qの日本語化、統合移動計算機環境の構築などの課題がある。これら課題の解決と安定化の後、本ソフトウェアの配布を予定している。

謝辞

本研究を進めるにあたって有意義な議論をして頂いたVIP Working groupをはじめとするWIDE Projectのメンバーの方々に感謝致します。また、本研究を進めるにあたっていろいろと力になって頂いた、慶應義塾大学村井徳田研究室の方々、電気通信大学並列分散処理研究室の方々に感謝致します。

参考文献

- [1] Fumio Teraoka, Kim Claffly, and Mario Tokoro: Design, Implementation, and Evaluation of Virtual Internet Protocol In *Proceedings of the 12th International Conference on Distributed Computing Systems*, June 1992
- [2] Uehara, K., Teraoka, F., Sunahara, H., and Murai, J.: Enhancement of VIP and Its Evaluation, In *Proceedings of INET'93*, August 1993.
- [3] Fumio Teraoka, Yasuhiko Yokote, and Mario Tokoro: A Network Architecture Providing Host Migration Transparency. In *Proceedings of ACM SIGCOMM*, September 1991.
- [4] Phil Karn, et al: KA9Q Network Operating System(NOS) package, 1989.
- [5] W. Croft and J.Gilmore: *Bootstrap Protocol*, RFC951, September 1985
- [6] R. Droms: *Dynamic Host Configuration Protocol*, RFC1541, October 1993
- [7] 富永明宏, 寺岡文男, 村井純: 動的ホスト設定プロトコル(DHCP)の実装の評価情報処理学会 マルチメディア通信と分散処理研究会 予稿集, 1993年12月