VIPv4の設計と実装

The Design and Implementation of VIPv4

舌間 一宏†　　　寺岡 文男††
Kazuhiro Shitama　　　Fumio Teraoka

†慶應義塾大学大学院 計算機科学専攻
Graduate School of Computer Science, Keio University
††(株)ソニーコンピュータサイエンス研究所
Sony Computer Science Laboratory Inc.

概 要

我々はこれまで、インターネットにおいて移動透過性を提供するプロトコルとしてVIP(Virtual Internet Protocol)を提案してきた。しかし既存のVIPは、セキュリティ強化のためにパケットのフィルタリングを行う最近のルータでVIPパケットが廃棄されてしまうという実用面での問題を有している。そこで本稿ではVIPv4を提案し、“独立レイヤ方式による実装”および“IP-in-IPによるカプセル化”によってこの問題を解決した。

1　はじめに

近年計算機の小型化・軽量化により、ユーザは計算機をどこへでも持ち運ぶことが可能となった。また単に持ち運ぶだけでなく、計算機を移動先のネットワークに接続し様々なサービスを受けるという形態も一般化しつつあり、いわゆるモーバイルコンピューティング環境が現実的なものとなってきている。

このような状況の中でネットワーク層で計算機の移動を隠蔽し、上位層に対して移動透過性を提供するプロトコルとしてVIP(Virtual Internet Protocol)を提案してきた[1]。VIPの基本概念はノードの位置指示子(IPアドレス)とホスト識別子(VIPアドレス)の明確な分離である。このVIPアドレスとIPアドレスの対応付けは各ホストがもつAMT(Address Mapping Table)で行われる。移動ホストのホームネットワークに接続するルータ(ホームルータ)が必ずその移動ホストに関する最新のAMTエントリをもち、通信相手のAMTエントリをもたないVIPホストやIPホストが移動ホストへ送信したパケットは、その移動ホストのホームルータによって転送される。既存のVIPには3つのバージョンがある。VIPv1は移動透過性を実現し、VIPv2ではノードの認証機能を加え[3]、VIPv3では安全なファイアウォール通過機構[4]および移動サブネットのサポート[5]を加えた。

一方最近ではセキュリティへの関心の高まりとともにIPオプション付きパケットやソースアドレス偽造パケットを廃棄するルータが多くなっている[7]。このような機能をもつルータはセキュリティを強化する上で非常に有用である。しかし、既存のVIPではその設計および実装方法のため、このようなルータをVIPパケットが通過する際に重大な問題が生じてしまう。

本稿では既存のVIPの問題点を明らかにし、その解決策を示す。さらにこの解決策をもとにVIPv4を提案し、その設計および実装について述べる。

2　問題点と解決策

2.1　IPオプション付きパケットの廃棄

プロトコル階層としてVIP層はIP層とTCP/UDP層の間に位置する。プロトコル階層

にしたがってVIPヘッダを独立したプロトコル階層として実装するとVIPホストと既存のIPホストの通信ができなくなる。そこで既存のVIPではIPとの互換性を考慮し、VIPヘッダをIPオプションとして実装している[2]。しかしIPオプションの1つを利用したセキュリティ攻撃が指摘されたため、種類にかかわらずIPオプションがついているパケットを廃棄するようにルータを設定することが多くなった。そのためVIPパケットはこのようなルータで廃棄されてしまう。

2.2 ソースアドレス偽造

既存のVIPでは、VIPホストがIPホストへVIPパケットを送信する場合、IPホストのTCP/UDP層にVIPホストのVIPアドレスが正しく伝わるように、VIPパケットの送信元IPアドレスとしてVIPホストのVIPアドレスを設定する[1]。つまり、VIPホストがホームネットワーク以外のネットワークに接続している場合、VIPホストのIPアドレスとVIPアドレスが異なっているため、そのVIPパケットは見かけ上ソースアドレス偽造パケットと同じになってしまう。そのため、セキュリティ強化のためにソースアドレス偽造パケットを廃棄するように設定されたルータでVIPパケットが廃棄されてしまう。この問題はIETFで標準化が進められているMobile-IP[6]でも同様に生じている。

またIPホストからVIPホストへのパケットをホームルータが転送する場合も、そのVIPパケットは見かけ上ソースアドレス偽造パケットと同じになってしまい、ソースアドレス偽造パケットを廃棄するように設定されたルータで廃棄されてしまう。

2.3 解決策

VIPパケットがセキュリティを強化したルータで廃棄されてしまうという問題点を、既存のVIPを次のように変更することにより解決した。

- 独立レイヤ方式によって実装する。
- VIPパケットがソースアドレス偽造パケットと見かけ上同じになる場合は、IP-in-IPのカプセル化を用いてそれを回避する。

独立レイヤ方式で実装するとVIPヘッダはIPのデータ部分の一部となるため、IPオプション付きパケットを廃棄するルータでVIPパケットが廃棄されなくなる。また、独立レイヤ方式の実装により拡散キャッシュ法が効率的でなくなってしまうが[1]、通信中のホスト同士がお互いのAMTエントリをもつだけで十分であると考える。

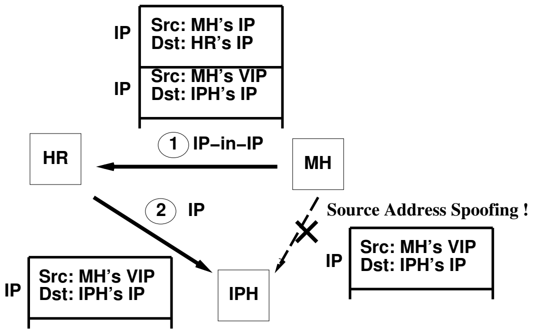


図1　IP-in-IPでのカプセル化(1)

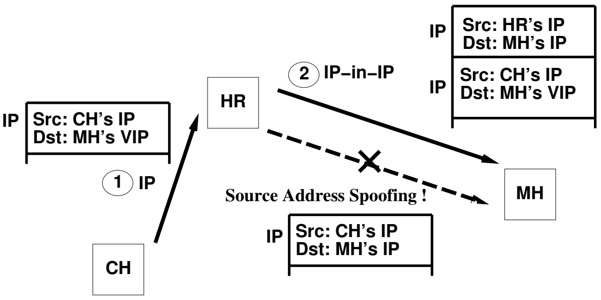


図2　IP-in-IPでのカプセル化(2)

またソースアドレス偽造パケットと見かけ上同じになるVIPパケットを別のIPヘッダでカプセル化することにより、ソースアドレス偽造パケットを廃棄するルータでVIPパケットが廃棄されることを回避することができる。IP-in-IPのカプセル化は次のように行う。なお、図1、図2は、通信時に送受信されるIPパケットのIPヘッダ中での送信先アドレスSrcと送信元アドレスDstの状態を示している。まず図1のように移動ホストMHがIPホストIPHと通信する場合、直接IPHにIPパケットを送信すると見かけ上ソースアドレス偽造パケットと同じになってしまう。そのため、そのIPパケットをIP-in-IPでカプセル化してまずMHのホームルータHRに送信する。それを受信したHRはカプセル化を解除し、IPHへ転送する。

また図2のようにホームルータHRが自分をホームルータとする移動ホストMH宛てのIPパケット(送信元のホストをCHとする)を受信した場合、IPヘッダ中の送信先IPアドレスを直接書き換えてそのIPパケットを送信してしまうと見かけ上ソースアドレス偽造パケットと同じになってしまう。そ

のため、受信したIPパケットをIP-in-IPでカプセル化してMHに送信する。

ただし、IP-in-IPのカプセル化を用いるのは上で示すような場合のみであり、通信中のホスト同士がお互いのAMTエントリをもっている場合は既存のVIPと同様に最適経路で通信が行われる。

3 VIPv4の設計および実装

本章では、2.3節で述べた解決策をもとにVIPv4の設計および実装を行う。

3.1 設計

VIPv4は独立レイヤ方式を採用し、またソースアドレス偽造パケットと見かけ上同じになるようなVIPパケットの送信をIP-in-IPのカプセル化で回避することによって、セキュリティを強化したルータでVIPパケットが廃棄されるという既存のVIPの問題点を解決している。

そのためVIPv4は以下のような点において既存のVIPと異なる特徴をもつ。

- 各ホストは自分のもつAMTエントリにないホストをIPホストであると判断する。
- パケットを送信する際、送信先ホストに関するAMTエントリがある場合に限り、VIPパケットを使用する。
- パケットを受信した際、そのパケットにVIPヘッダがない場合に限り、自ホストのAMTエントリを送信元ホスト上に作成するため、そのパケットの送信元ホストにAMTアップデートパケットを送信する。
- 移動ホストが異なるサブネットに移動した場合は、その移動ホストのホームルータおよび通信中の相手ホストにのみAMTアップデートパケットを送信する。
- AMTアップデートパケットにはUDPを用いる。

なおAMTアップデートパケットを受信した場合は認証を行い、確かに本当のホストからのパケットであることを確認した上でAMTエントリを操作する。実際の認証方法は既存のVIPとほぼ同じ手法[3]を用いて行う。

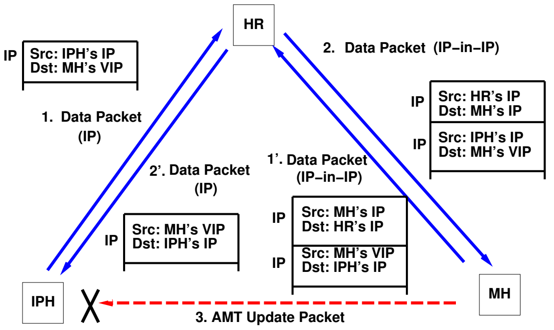


図3　MHとIPH間の通信

3.2 動作例

ここではVIPv4を実装した移動ホストMH(MHのホールルータをHRとする)が、次のような2つホストと通信する際の実際の動作例を示す。

- VIPv4を実装していないホストIPH
- VIPv4を実装している移動ホストMH2(MH2のホームルータをHR2とする)

MHとIPH間の通信

IPHがMHにパケットを送信する場合、IPHはMHの現時点でのIPアドレスを知らないため、必ずHRを中継することになる(図3の1.と2.)。その際、ソースアドレス偽造パケットと見かけ上同じにならないように、図2にならってHR→MHの経路はIP-in-IPのカプセル化を用いて送信する。

このパケットを受信したMHは、受信パケットにVIPヘッダがないので、IPHへAMTアップデートパケットを送信する。しかしIPHはIPホストであるため、そのAMTアップデートパケットは無視されてしまう(図3の3.)。なおAMTアップデートパケットはデータパケットと異なるため、IPヘッダの送信元IPアドレスはMHのIPアドレスのままでよい。したがってHRを中継する必要はない。

またMHがIPHにパケットを送信する場合も、MHはIPHに関するAMTエントリをもつことはないため、IPHをIPホストであると判断し、IPパケットを用いて送信する。その際もソースアドレス偽造パケットと見かけ上同じにならないように、図1にならってまずHRへIP-in-IPのカプセル化を用いて送信し、それを受信したHRはカプセル化を解除して、IPHへそれを転送する(図3の1'.と2'.)。

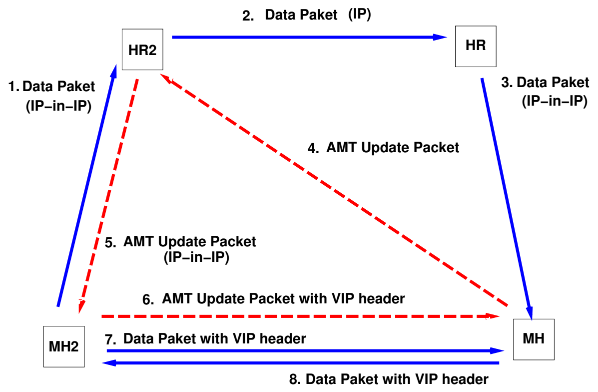


図4　MH と MH2 間の通信

MH と MH2 間の通信

MH と MH2 はお互いの AMT エントリを通信開始時点にもっていないものとする。MH2 から MH への通信では、MH2 は MH の AMT エントリをもっていないため MH を IP ホストであると判断し、IP パケットを MH へ送信する (図 4の 1. と 2. と 3.)。

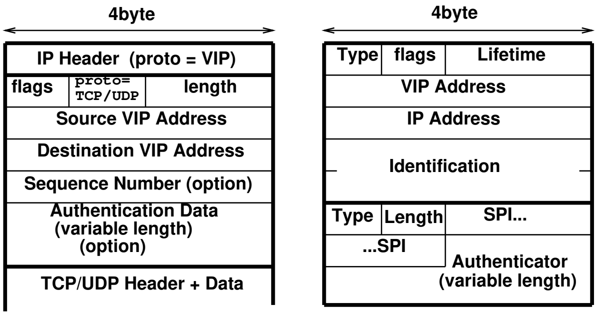
その IP パケットを受信した MH は、MH2 へ AMT アップデートパケットを送信する (図 4の 4. と 5.)。

MH からの AMT アップデートパケットを受信した MH2 は認証を行い、MH2 の AMT に MH のエントリを加える。さらに受信した AMT アップデートパケットには VIP ヘッダがないため、MH へ AMT アップデートパケットを送信する (図 4の 6.)。なお MH2 は MH の AMT エントリをもつので、これ以降 MH2 から MH への通信は VIP パケットを使用し最適経路で行われる (図 4の 7.)。

MH2 からの AMT アップデートパケットを受信した MH は認証を行い、MH の AMT に MH2 のエントリを加える。なお受信した AMT アップデートパケットには VIP ヘッダがあるため、MH は MH2 へ AMT アップデートパケットを送信しない。また、MH は MH2 の AMT エントリをもつので、これ以降 MH から MH2 への通信は VIP パケットを使用した最適経路で行われる (図 4の 8.)。

3.3　実装

現在 VIPv4 を FreeBSD-2.2.1 上で実装中である。なお、認証機構をユーザプロセスとして、残りの部分をカーネルの一部として実装を行っている。



(a) Data Packet　(b) AMT Update Packet (UDP)

図5　VIPv4 のパケットフォーマット

データパケットのフォーマットを図 5(a) に、AMT アップデートパケットのフォーマットを図 5(b) にそれぞれ示す。

4　おわりに

本稿ではセキュリティへの関心が高まったことによって生じる既存の VIP の問題点およびその解決策について説明した。さらにその解決策をもとにした VIPv4 の設計および実装について述べた。

今後は現在進行中の実装を早く完了させ、実際の広域網において動作の確認や十分な評価を行う予定である。また現段階では考慮していない安全なファイアウォール通過機構 [4] や移動サブネットのサポート [5] についても将来的には VIPv4 の機能として導入していくつもりである。

参考文献

[1] F.Teraoka, K.Uehara, H.Sunahara and J.Murai, *VIP: A Protocol Providing Host Mobility*, CACM Vol.37, No.8, pp.67-75, Aug. 1994

[2] F.Teraoka and M.Tokoro, *Host Migration Transparency in IP Networks: The VIP Approach*, ACM Computer Communication Review Vol.23, No.1, Jan. 1993

[3] 植原啓介, 寺岡文男, 砂原秀樹, 移動ノード用通信プロトコル VIP における認証機構, 日本ソフトウェア科学会 第 11 回大会論文集, 1994 年 10 月.

[4] 寺岡文男, VIPv3 における安全なファイアウォール通過機構, 情報処理学会 モーバイルコンピューティング研究グループ研究会 報告集, 1996 年 7 月.

[5] 石井公夫, 寺岡文男, 村井 純, 移動するネットワークのための透過的な通信機構の設計, 情報処理学会 マルチメディア通信と分散処理研究会 報告集, 1995 年 10 月.

[6] C.Perkins, *IP Mobility Support*, RFC 2002, October. 1996

[7] Computer Emergency Response Team (CERT), *IP Spoofing Attacks and Hijacked Terminal Connections*, CA-95:01, January. 1995